Segmentation of MP2RAGE data

using BrainVoyager and manual

by Jorie van Haren

April 4, 2024

Contents

0.1 Short word

Below you will find instructions for my pipeline to automatically segment Ultra-High Field fMRI data, and tools to manually fine-tune the segmentation. The current procedure relies on a combination of BrainVoyager 22.2 (Version 22.2.1.4950, 64-bit), BrainVoyager's in house DNN 'Tiramisu' Segmentation (Python evnironment needed), custom Python BrainVoyager Tools and 3D-Slicer (or ITK Snap).

Lately, I have started incorporating BrainVoyager's advanced segmentation, specifically to obtain an adjusted graymatter segmentation. This segmentation is 'grown' from the DNN's white matter segmentation, and is quite good in preserving cortical thickness. However, I just use this additional graymatter segmentation as an alternative mask within 3D-Slicer.

- BrainVoyager download page:
- BrainVoyager DNN Segmentation installation instructions:
- BrainVoyager Python Plugins:

0.2 Automatic Segmentation

In order to automatically segment using BrainVoyager's DNN Segmentation tool, you will a MPRAGE or MP2RAGE image, preferably at 0.7mm isotropic.

1. Open UNI file: *e.g. 'UNI.vmr'*
2. Make sure the correct Python environment is running
    - Go to **Python/Select-Python**
    - Select your DNN segmentation environment (see BrainVoyager documentation for help): *e.g. DNNsegmentEnv*
3. Go to **Volumes/DNN-Segmentation**
    - Select MP2RAGE

- Select Slow

4. This process will result in a couple of files:

```
'UNI_reframed.vmr' :          reframed UNI to fit DNN bounding box
'UNI_reframed.v16' :          reframed UNI to fit DNN bounding box
'UNI_reframed_tissue-probs-slow.vmp' :   Tissue probability maps
```

5. Go to **Volumes/DNN-Postprocessing** (if not automatically opened)
   - Boost WM over GM: Play around with this a bit, and adjust based on results and personal preferences. For me *0.3* seems to work well with the hole filling procedure I later do within 3D slicer.
   - Press **Convert**, to create *'UNI_reframed_tissue-labels.voi'*
   - Lastly we **Clean** up the VOI by removing non-brain voxels.

0.3 Generating files for manual segmentation

Next we convert our created files to the correct resolution and convert them to NIFTI in order to manually finetune the segmentation files. In my pipeline I always adjust the VOI files directly instead of the WM/GM file, in a way this is personal preferences. In this step I will also generate a few additional files that help me Segment more efficiently, again, this is personal preferences with many of these files being optional.

1. Isovoxel *UNI_reframed* by going to **3D-Volume-Tools/Spatial-Transf /Standardize/Iso-Voxel...**
   - Set Target voxel size of X,Y,Z : **0.4**
   - Set Framing cube dimension : **512**
   - Set interpolation type : **3**
2. Restart BrainVoyager and select a Python environment with BVBabel enabled
   - Go to **Python/Select-Python**
   - Select your BrainVoyager Python environmen: *e.g. bv_preproc*
3. Isovoxel VOI (binary) files to Nearest Neighbour **Python/Python-development /Isovoxel_Nearest.py**
   - For ease of use, make sure you have UNI_**reframed.vmr** opened - This way the correct VOI bounds are automatically fetched
   - File Name: **'UNI_reframed_tissue-labels.voi'**
   - Framing Cube Dimension: **512**
   - VOI Bounds (probablly preloaded): **X=256, y=320, z=320**
   - Source Voxel Size: **X,Y,Z = 0.7**
   - Target Voxel Size: **X,Y,Z = 0.4**
4. Creating Gradient Magnitude file: **Volumes/Advanced-Segmentation-Tools**
   - Open *'UNI_reframed_ISO-0.4.vmr'*
   - Press **WM-GM Border** and select **Calculate**
   - This action results in a Gradient Magnitude file: *'UNI_reframed_ISO-0.4_GradsMag.vmr'*

5. Next we want to convert our newly created files into a format that can be read by *3D-Slicer* or *ITK-SNAP* (NIFTI)
   - Open **Python/Python-development/NIFTI_Tools**
   - Convert to NIFTI: *'UNI_reframed_iso-0.4.v16 '*
     Check **V16** and un-check any other File Type
   - Convert to NIFTI: *'UNI_reframed_ISO-0.4_GradsMag.vmr'*
     Check **vmr** and un-check any other File Type
   - Convert to NIFTI: *'UNI_reframed_tissue-labels_ISO04.voi'*
     Check **VOI** and uncheck any other File Type
     Set X,Y,Z to **512**
6. We also would like to have the labels and colours associated with our VOI file
   - Open **Python/Python-development/VOI_Tools**
   - VOI Document: *'UNI_reframed_tissue-labels_ISO04.voi'*
   - Press Create .ctbl for 3D-Slicer and .label for ITK-SNAP
7. Lastly, I always like to export the VTC outer box. This box helps me where my functional data is actually located, and where the most finetuning is actually needed
   - VTC Document: **any associated VTC file**
     e.g *S01_SES1_run1_FMR_SCSTBL_3DMCTS_THPGLMF7c_TOPUP.vtc*
   - Bounding Box: **512**
   - Press 'Create NifTi Bounding Box'

0.4 Moving Files

Next, I would advice that you move the created NIFTI and .ctbl/.label files to a 'segmentation' folder. These files include:

```
'UNI_reframed_ISO-0.4.nii.gz' :
            Main anatomical file
'UNI_reframed_tissue-labels_ISO04.nii.gz' :
            Main autmatical segmentation file
'UNI_reframed_ISO-0.4_GradsMag.nii.gz' :
            Gradient Magnitude file
'S02_SES2_run1_FMR_BoundingBox.nii.gz' :
            Bounding box of functional data
'UNI_reframed_ISO-0.4_tissue-labels.ctbl' :
            Tissue labels and colors
```

0.5 Manual segmentation in 3D-Slicer

I prefer the feel and touch support of 3D-Slicer, while the program is a bit on the heavy side it offers quite some nice tools in getting the segmentation at a good starting point before starting to manual segment.

1. Press **File/Add-Data**, Choose Directory to Add, Select only the .ctbl file (in order to load the labeling), press Ok
2. Add data once more, re-select the same folder, Uncheck only the .ctbl filePress **Show Options**

- Set *BoundingBox* to Segmentation
- Set *UNI_reframed* to volume
- Set *GradsMag* to volume (can later overlay this as segmentation map)
- Set *Tissue-labels* to Segmentation, with Color mode *UNI_reframed_tissue-labels*

3. One can edit the properties (visibility) of the segmentation. I normally put the slice fill alpha on 0.05 and the slice outline on 0.55
4. Go to Data, left-click on the GradsMag file and press Segment This, Press Add and Segment based on Threshold - Choose black as the segmentation colour
5. At the Data view select the main anatomical image by clicking the eye-symbol
6. Save the scene (.mrml) and gradiant magnitude segmentation (.seg.nrrd)
7. I start by growing the **CSF** into **Grey matter** (1x1x1), to reduce the bridging effect associated with the *Joint Smoothing s*
8. Next we apply **Join Smoothing** for everywhere: With smoothing factor set to **0.40**
9. We then grow the **Grey matter** into the **CSF** (1x1x1). Expanding it back to the previous state, with less likelihood for extensive bridging.
10. Normally I also do some extra growing iterations with a mask applied based on threshold, to selectively fill what is missing from tissue segmentations.
11. I then do some automatic segmentation steps, a few Smoothing/Opening Grey matter into White matter, Kernel size set to 3.0mm
12. And a few Median smoothing from White matter into Grey matter.

0.6 Segmentation Post-processing

When we are done segmenting our data and we are happy with the results, we return to BrainVoyager and do the last steps of DNN Post-Processing - removing cerebellum from subcortical VOI, Disconnecting left and right hemispheres, and creating standard segmentation VMR.

Similar to how we converted BrainVoyager files to NIFTI format, we can use **Python/Python-development/NIFTI_Tools** to convert NIFTI files back into BrainVoyager files. Most importantly, we can convert from NIFTI to a VOI, using our original VOI for the necessary header information.

0.7 Optional: Advanced Segmentation GM/CSF Boundary

An additional file to use within our manual segmentation program is the WM-GM file obtained from BrainVoyager's Advanced Segmentation pipeline. This approach uses the WM-GM boundary to grow GM until it hits a certain contrast difference. While this WM-GM file is worst in some ways (namely: is does not deal with blood vessels well), it is excellent in preserving cortical thickness, and assuming the existence of CSF.

The way I use this additional WM-GM file is as follows; I load the WM-GM file into 3D-Slicer as an additional segmentation. Since (in my case) the DNN segmentation seems to underestimate GM, I grow the DNN greymatter into the Advanced Segmentation GM (for a couple of iterations) - here I add an additional contrast mask to exclude blood vessels. Next, I take the CSF from the advanced segmentation and subtract it from my GM, this is

to deal with instances where GM was overestimated and CSF was underestimated. Lastly I sellect bloodvessels, and re-substract them from my adjusted GM.

Note that these steps should prefferably be done ad a first step after initial joint smoothing, and after manual WM adjustments.

1. Open *'uniden_IIHC_masked_ISO-0.4'*
2. Open **Options/Contrast and Brightness**
   - Adjust the brightness in a way that the GM/CSF boundary is clearly visible
   - Given that our attention is solely on the GM/CSF, it's acceptable if the white matter appears washed out.
3. Open **Volumes/Advanced Segmentation Tools/Preparation** and run **Tissue contrast enhancement** - *The following settings seems to work for me*
   - Cycles: *'7'*
   - Range: *'5'*
   - Run **Enhance**
4. Open **Analysis/Region of Interest Analyses** (or press Ctrl+R)
   - Press **Load**: *'UNI_reframed_ISO-0.4_tissue-labels_CL_RC.voi'*
   - Sellect *'White matter'*, *'Ventricles (Lateral, 5th'*, and *'Subcortical structures Edited'*
   - Go to **Color: 240**, and press **Draw in VMR** (*open >*)
5. Save *'uniden_IIHC_masked_ISO-0.4_ETC-7x-R5_WM.vmr'*
6. Open **Volumes/Advanced Segmentation Tools/GM-CSF Border** and run **Dilate to CSF**
   - No of dilate steps: **14**
   - Stop crit left from GM peak: **0,20** - *lower for underestimation*
   - Local stop criterion: **0,20**
7. Polish - (*This step might also be useful for the post-processing of a manual segmentation, to do so just export your NIFTI with wm: 240, and gm: 245*)
8. Save the WM-GM file as NIFTI
   - Open **Python/Python-development/NIFTI_Tools**
   - Convert to NIFTI: *'uniden_IIHC_masked_ISO-0.4_ETC-7x-R3_WM_GM.vmr'* Check **Vmr** and un-check any other File Type

Within 3D-Slicer you now load the saved NIFTI as Volume, and as Segmentation.

- **Volume:** allows you to WM, GM or CSF as an intensity threshold
- **Segmentation:** allows you to use *Logical operators* to *subtract*, *invert* and *add*
- **Segmentation:** additionally allows you to *Grow* **Grey matter** into **Segment_100**, while having thresholded bright areas (i.e. blood vessels)