Same-Subject QC Protocol

Author: Lucía de Hoyos

Contact: luciadeh@gmail.com

29 May 2020

Description

The same subject quality control is a protocol used to check the data from a longitudinal sample by checking if the scans from different timepoints belong to the same subject or not (in case they have been mislabelled).

Availability

The scripts that are used in this protocol are available at github.com/ldehoyos/ssQC.

Process

This protocol has 4 steps that will be explained within this manual in more detail.

1. Extract Data
2. Create the Sagittal Images
3. Order the Sagittal Images
4. Visualize the Results

Step 1. Extract Data

This step is optional. It is useful in case you need to share or move the information of the subjects, as instead of moving the output from FreeSurfer, you have to move just 1 file per subject: the orig.mgz file. In case you do not want to do it, you can skip it, but if you do so, you will need to change the function you are using in step 2.

To run this step, you need to have one folder per subject, and within that folder the FreeSurfer output folders (Figure 1). The orig.mgz file is within the mri folder. That is where the script will look for the file.

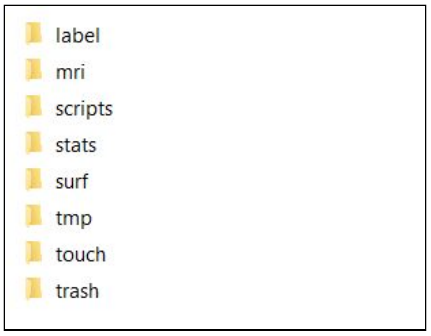

Figure 1. FreeSurfer output folders for each subject.

The script to run this step is **getorigFiles.R** and you need to do two things before running it:

1. Set the working directory to your subjects folder by changing the setwd() command on the script
2. Set the variable outDir to where you want the files to be outputted. This output directory will have the same number of folders as your working directory, but within each folder you will have the orig.mgz file instead of the 8 output folders (Figure 1).

Step 2. Create the Sagittal Images

In this step, we will obtain the sagittal images for each subject. There are 2 ways of selecting the output sagittal image:

1. Select a slice, e.g. slice 250. The problem with this approach is that not all the subjects will be centered, and the view you obtain is not the same.
2. Detect which are the side slices and get the slice which is in the middle of both (see Figure 2). The side slices are the first and last slice with a non-zero value (i.e. each side of our skull).

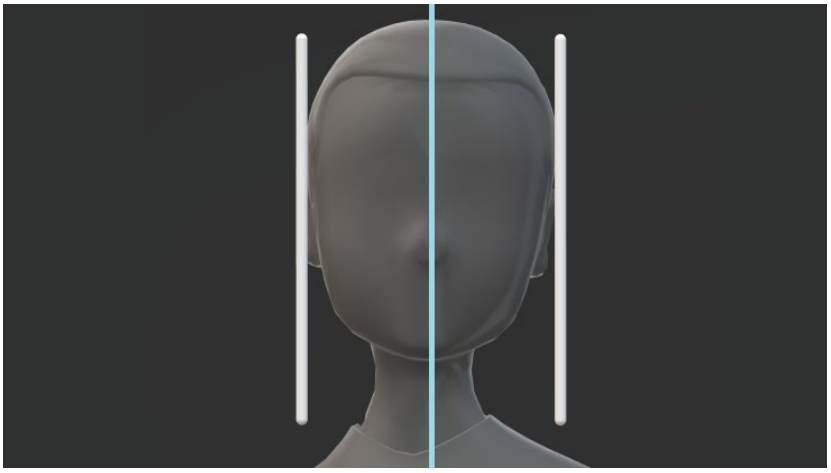

Figure 2. Side Slices (in white) that are used to select the mid slice (blue).

The most effective approach is the second one. To run this approach you need to run the script: SagittalQCing_MidSlice.m. Otherwise, if you want to run the first approach you should run the SagittalQCing_SelectSlice.m script. For both approaches you need to have the functions folder and the subjects folder (the output directory of Step 1) added to your MATLAB path. You also need to set the variable FS_directory to your subjects directory and, in case of the 1st approach, you need to set the slice in sct variable.

** In case you did not run Step 1, you need to change the functions used: getMidSagittalPNG by getMidSagittalPNG2 in SagittalQCing_MidSlice.m and getSagittalPNG by getSagittalPNG2 in SagittalQCing_SelectSlice.m **

Step 3. Order the Sagittal Images

Once the sagittal images have been created, they are outputted to the subjects directory. In order to organize them into folders for each subject, you have to run this step using the script orderSagImages.R.

Before running the script, you need a txt file with the names of the subjects (e.g. mygroup.txt). This is done in case you want to select certain subjects (e.g. those with 2 scans only). This script has the option of adding 3 different groups: subjects with 2, 3 and 4 timepoints (lines 6-8). In case you want to just use 1 group (because you only have 2 timepoints), you can comment those lines.

Then, in the script orderSagImages.R you need to set the variables inDir (where the sagittal images are) and outDir (the output folder to create the output for the same-subject QC). Also, an important part of this step is that your subjects are labeled as *subjid_ADNI1*, where subjid can be whatever you want, and ADNI1 means that is the first timepoint of the subject (the second will be ADNI2 and so on). In case there is another way of labelling the subjects, you may want to change the script.

Step 4. Visualize the Results

Once the images have been put into folders with the subject names, and within each folder there are the timepoints of the subject (ADNI1, ADNI2...), we can create an html file to visualize the results.

In this step, there are 3 scripts:

- make_webpage_2timepoints.sh
- make_webpage_3timepoints.sh
- make_webpage_4timepoints.sh

Each one is an example of how to create a webpage to display results of subjects with 2,3 and 4 timepoints, respectively.

To run this script, you need to use the linux terminal and run (example for make_webpage_2timepoints.sh):

```
# Change directory to where your script is
cd {scriptdir} (where scriptdir is the directory of this script)

# Change your script mode to execute it
chmod 777 make_webpage_2timepoints.sh

# Run the script. (dir0 is the output directory of the MATLAB script)
./make_webpage_2timepoints.sh ${dir0}
```

** Note that for this step it is also important to name the timepoints of your subjects as ADNIx, where x is the number indicating the timepoint **