fMRI Preprocessing Graphical User Interface (GUI) Manual

Version 1.0 (November 2016)
GUI Version: 3.0 (November 2016)

Katie Lavigne (lavigne.k@gmail.com)

Contents

Introduction

This document contains instructions for the Woodward Lab fMRI Preprocessing Graphical User Interface (GUI), which can be used for batch preprocessing of fMRI data following the Woodward Lab pipeline. The GUI and this manual can be found on the CFRI network drive under **cfrifs02/WoodwardLab/Scripts/fMRIpreprocessingGUI/**. The automated procedures of this GUI follow the preprocessing pipeline described in the original non-automated fMRI preprocessing manual **(cfrifs02/WoodwardLab/Manuals/PreprocessingManual/Woodward_fMRI_PreprocessingManual_Feb 2015**).

You should have the following software installed on a Linux-based machine (this program was developed on Ubuntu 16.04, but might work on other distributions and operating systems):

1. MATLAB (tested on R2014a)
2. SPM toolbox for MATLAB (tested on SPM8)

Other optional software used in this guide are **dcm2nii** (converting scans to NIFTI) and **MRIcron** (structural image/segmentation quality control). Dcm2nii comes packaged with MRIcron (http://people.cas.sc.edu/rorden/mricron/index.html) and is free to download. These programs are used because they can be run from the command line and are simple to script for batch processing.

Many of the additional data organization and quality control sections described below utilize Linux Bash Programming, which is why a Linux-based machine is strongly recommended. Included in this guide are example scripts and instructions for developing your own scripts for these processes (some minor details will change across different computers and studies). If you are unfamiliar with BASH programming, I would strongly suggest you learn it, as these small scripts will save countless hours of tedious work.

If you come across any errors during preprocessing, feel free to contact Katie Lavigne (lavigne.k@gmail.com), but see the **Common Errors** section below first.

1. Data Organization

When working with programs and scripts that loop through a large number of files and folders, it is imperative that your data is well organized, with a consistent directory structure and naming conventions. The underlying code of the fMRI Preprocessing GUI often looks for files and folders based on user-defined wildcards, and will miss data if they are not properly organized or named.

a. Directory Structure

The GUI's preferred directory structure is as shown in

Figure *1*. This has separate directories for functional and structural data, both with subject folders inside (below is an example with three subjects). For functional data, scans for each run are in separate, consistently-named run folders, while structural scans are in the main subject directory under the structural folder, without any additional subfolders.

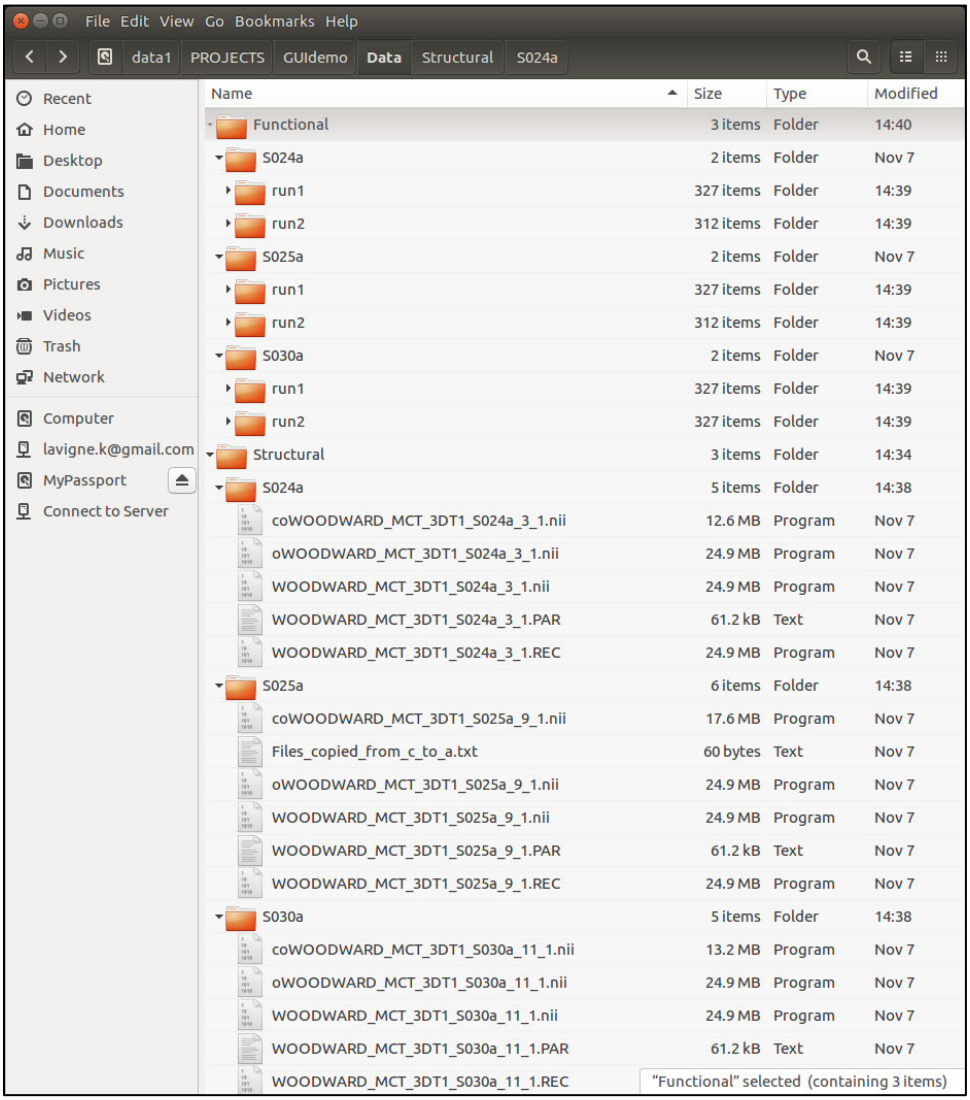


Figure 1. Preferred Directory Structure

b. Possible Alternative Directory Structures

While these have not been thoroughly tested, the GUI may still function with some modifications to the preferred directory structure:

1. For studies with a single run, you may put the functional scans directly in the subject folder without any run folders
2. Separate structural and functional folders may not be necessary, provided the structural scans are directly in the subject folder (see Figure 2). However, this might require you to have run folders for your functional data, especially if the functional/structural filenames overlap.

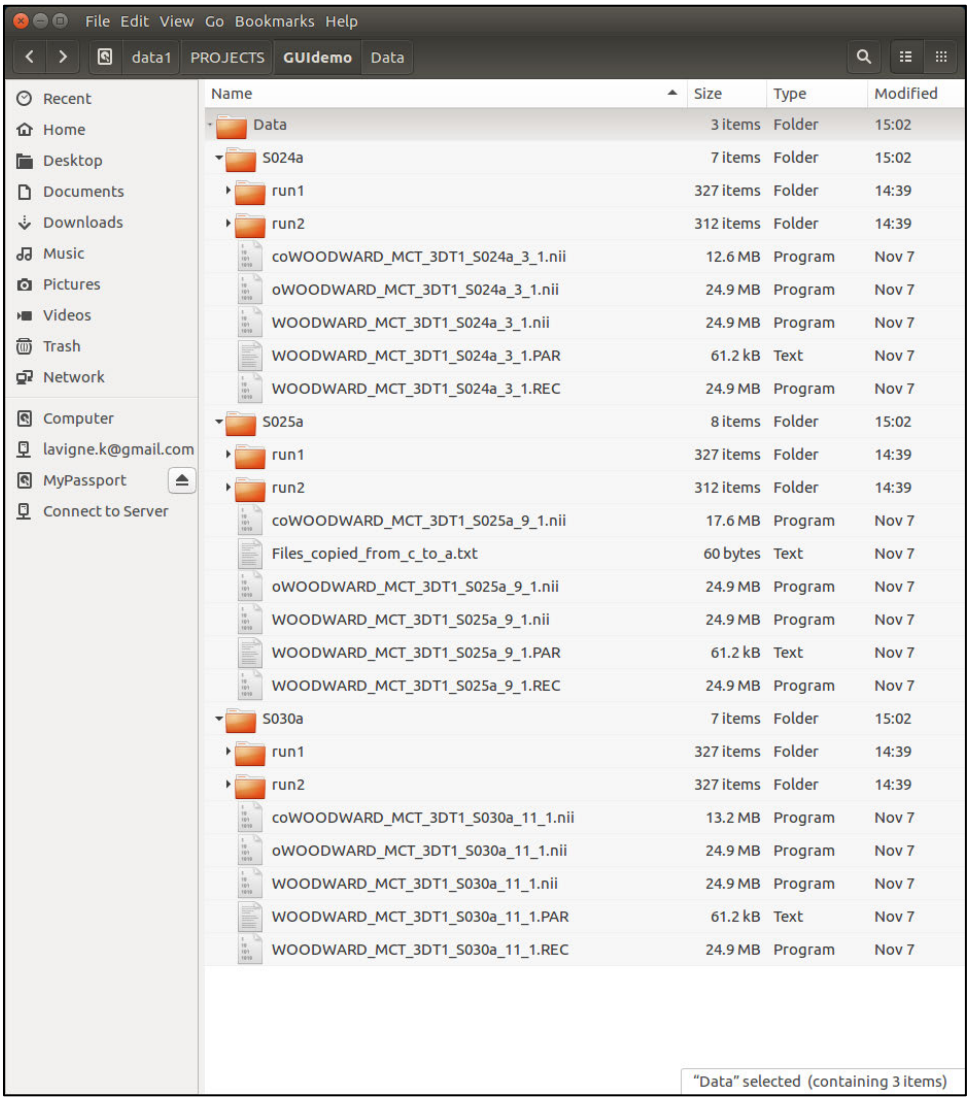


Figure 2. Alternative Data Organization

c. Quickly Reorganizing Data

Linux Bash Scripting is an extremely efficient way of reorganizing your files and folders if they do not fit the directory structure or naming conventions needed. Below is some example code for some common ways of reorganizing files and folders.

1. Replacing Spaces from Filenames

Spaces in filenames can cause a lot of trouble in Linux operating systems because it may treat the filename as two separate files. There are several different ways to remove spaces from folders and filenames (it is best to replace them with underscores so they remain legible).

Here is one example:

- Use the "find" command: find [path_to_parent_directory] -depth -name "* *" to find all the files and folders within the [path_to_parent_directory] that contain spaces
    - Add -type d or -type f to restrict this command to directories or files, respectively
    - Combine with the "rename" command: -exec rename 's/ /_/g' "{}" \; to replace spaces with underscores
    - Re-run the original find command to see if it worked (below you can tell it worked because there is no output from the second find command, meaning it could not find any more files with spaces)

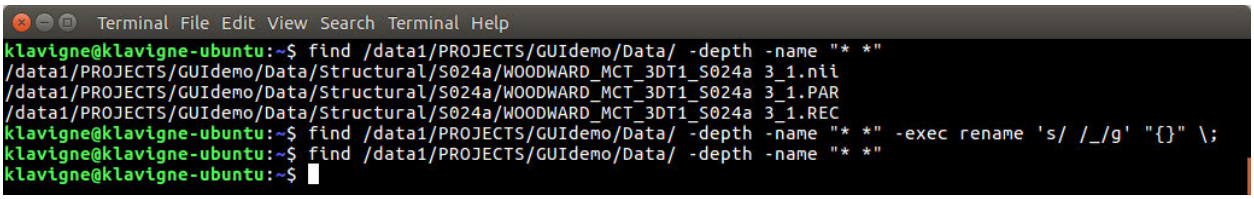


Figure 3. Replacing spaces in filenames

2. Renaming Files and Folders

The GUI will use the folder names as subject IDs and use these to look for the proper files. If your files don't include the subject names or the cases aren't consistent (upper vs lowercase), you can quickly fix this again using the find command: find [path_to_parent] -depth -name "*wildcard*" | while read file; do mv ${file} ${file//text_to_remove/text_to_add}; done. Below is an example for changing case of a particular subject, but you can make more general wildcards or use "mv" with a "for" loop instead to go through all subjects (see next section).

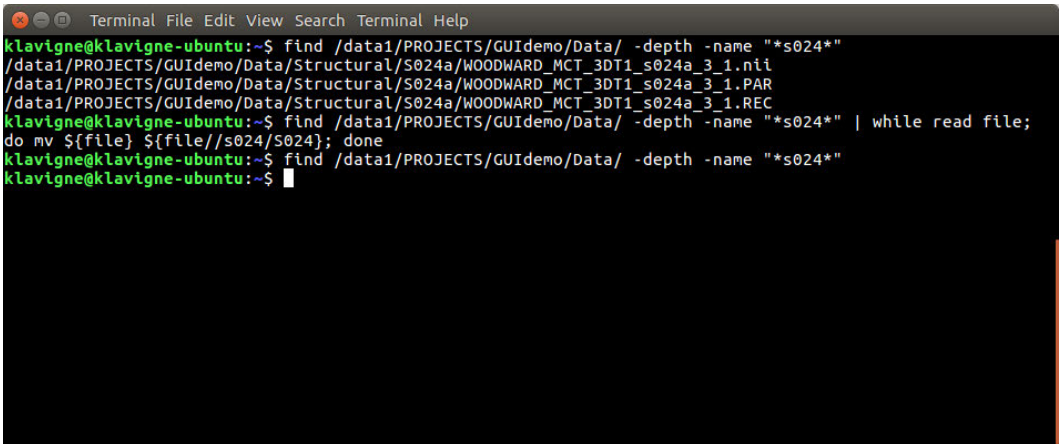


Figure 4. Renaming Files and Folders

3. Moving and Reorganizing Files and Folders

If your data is not in the proper directory structure, you will need to reorganize them in order for the GUI to work properly. Instead of doing it manually, you can write a small Bash script that will do everything for you. The layout of the script will heavily depend on your current directory structure. Below is an example:

Figure 5. Example script for moving/reorganizing files and folders.

2. Converting to NIFTI

The UBC Scanner outputs data in PAR/REC format, which must be converted to Analyze (.hdr/.img) or the newer NIFTI (.nii) format before preprocessing. The following instructions describe how to use dcm2nii to convert PARs/RECs to NIFTI format using Bash scripting, which is the preferred option when dealing with large datasets. If you would prefer to use the dcm2nii GUI, please see the alternative manual for non-automated preprocessing mentioned in the introduction of this guide.

There are three files you will need to convert your data, which are all located within the fMRIpreprocessingGUI folder. The first is **dcm2nii_convert** (see Figure 8), a Bash script that you will likely need to modify for your study. To do so, you can simply open it in a text editor, such as gedit (cd to directory where the file is located and type gedit dcm2nii_convert in a terminal). The other two files are **dcm2nii.ini** and **dcm2nii.txt**, which are parameter files that tell the program what options you want for conversion. The .ini file is updated regularly, so the .txt file is there to make sure the parameters stay consistent across subjects. If you want to make any changes to the conversion options, you should do it in the .txt file (see http://people.cas.sc.edu/rorden/mricron/dcm2nii.html).

The dcm2nii_convert script assumes your directory structure is organized as described above (with separate functional and structural directories, and run directories for functional data). If that is not the case, either update the directory structure or the script itself to make it consistent.

INSTRUCTIONS

- Open a Linux terminal (CTRL+ALT+t)
- Change to GUI directory: e.g., cd /data1/PROJECTS/GUIdemo/fMRIpreprocessingGUI
- Type ./dcm2nii_convert to run the script and then hit Enter

Figure 6. Running dcm2nii_convert

- Select the directory where your data is located
  - The directory you choose will depend on how you coded the script. With the example script, you should choose the directory above the Functional/Structural folders, because the script loops through those folders
- If your script is working properly, it should run through each subject, scan type (functional, structural), and run (functional only) and convert any PARs/RECs it finds into NIFTI format
  - This is an example of command line output for one subject's structural scan:

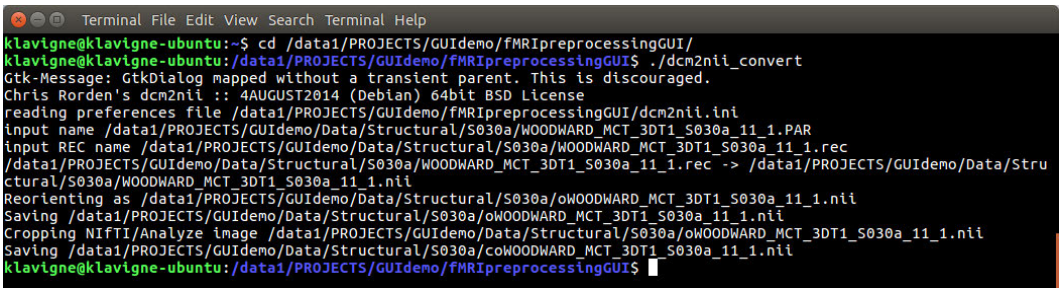


Figure 7. dcm2nii_convert output

IMPORTANT NOTE: The script example below includes a more complex code, which will prevent it from running subjects who have already been completed. This is very useful if data collection is ongoing, since the code can simply be re-run without any changes when new data are available and won't overwrite previously completed subjects. **If you are reorienting your structural images, this is especially important, because the structural .nii files are modified during reorientation and re-running this script without the flag for new subjects would overwrite the reorientation.**

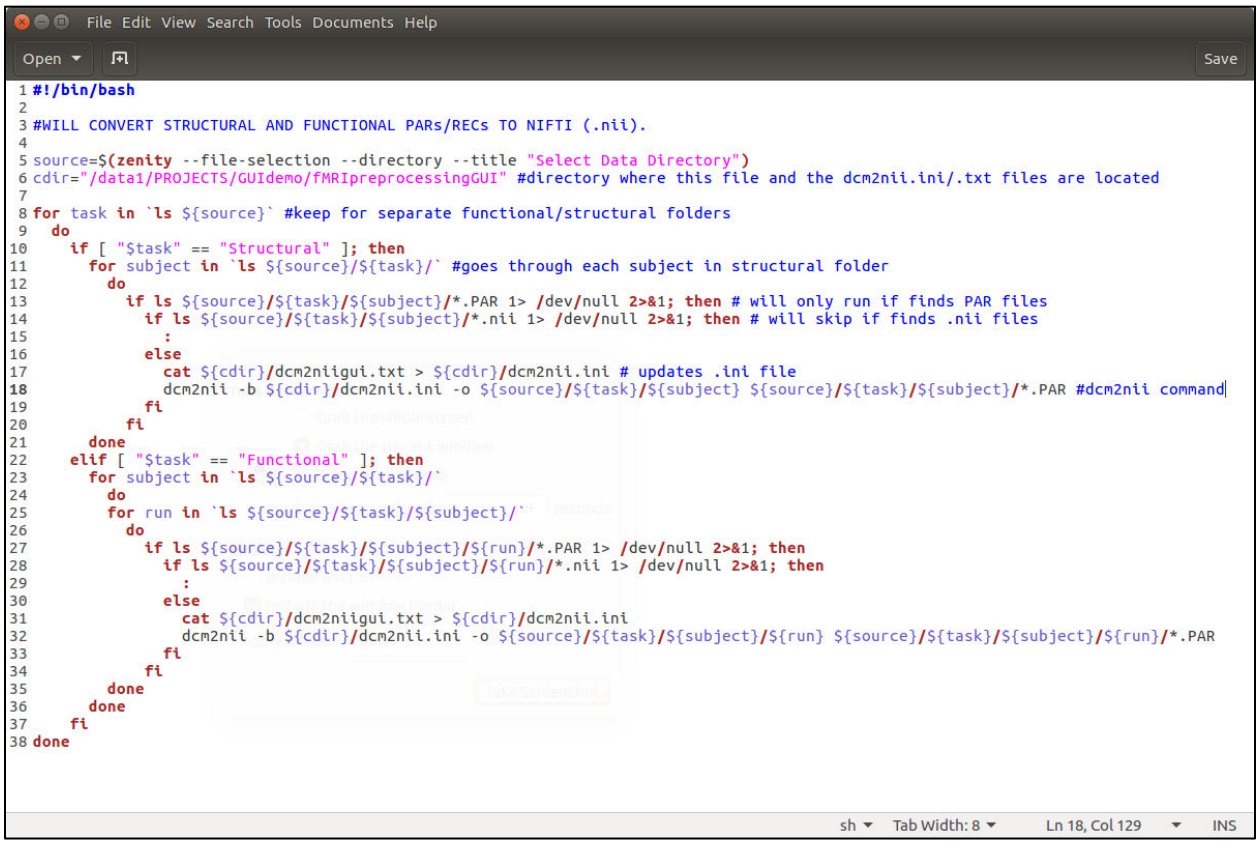


Figure 8. dcm2nii_convert script

3. Getting Started

- Open MATLAB by typing matlab in a terminal (do not use sudo)
- Change directory to where you want the preprocessing job files to be saved
  - It is best to create a new Preprocessing folder one or two directories up from where your data is. You can make a new folder in MATLAB using mkdir.

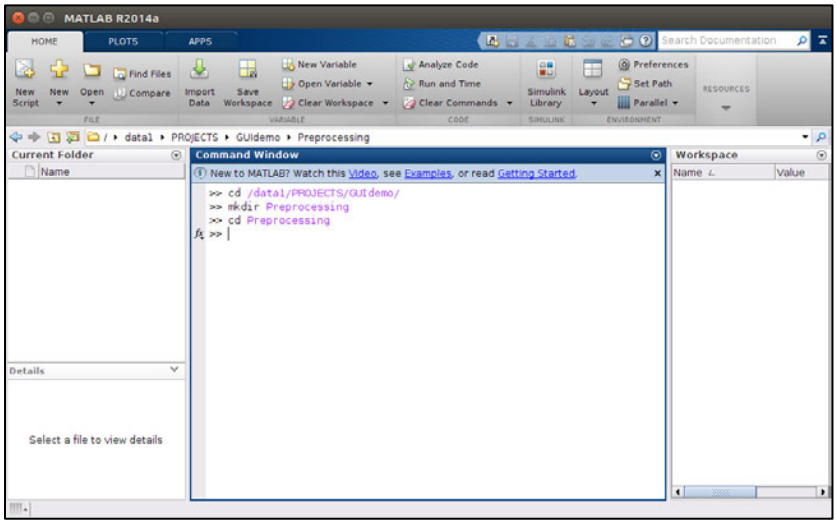


Figure 9. Make a separate preprocessing directory.

- Make sure the fMRI Preprocessing GUI is in your path
  - Try preprocessfMRI in the command window
  - If it doesn't work, add it using the addpath command or clicking on Set Path
    - addpath('/data1/Applications/fMRIpreprocessingGUI')
    - savepath /usr/local/MATLAB/R2014a/toolbox/local/pathdef.m

Note: if you do not have permissions to save the path, first type the following in a command window, being sure to modify the code so it is pointing to the right directory where your MATLAB program is installed: sudo chmod 777 /usr/local/MATLAB/R2014a/toolbox/local/pathdef.m

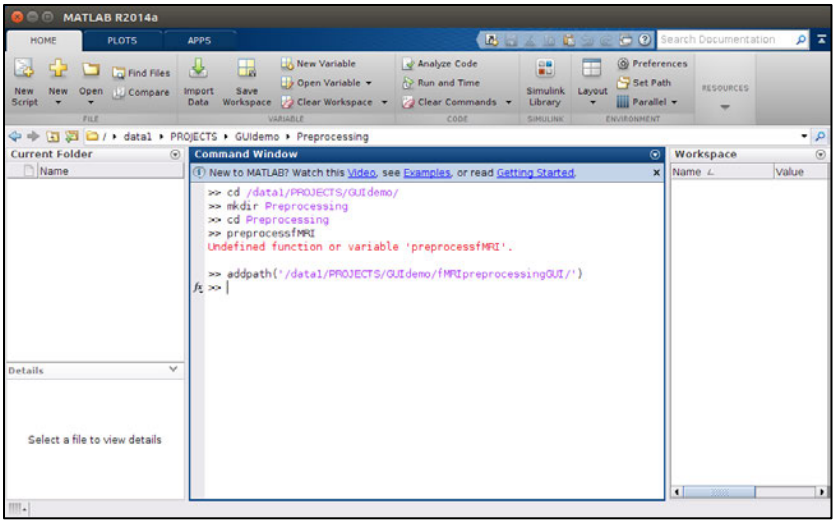


Figure 10. Add GUI folder to path

- Start the GUI by typing preprocessfMRI in the command window
  - If SPM is not in your path, it will ask you to add it:

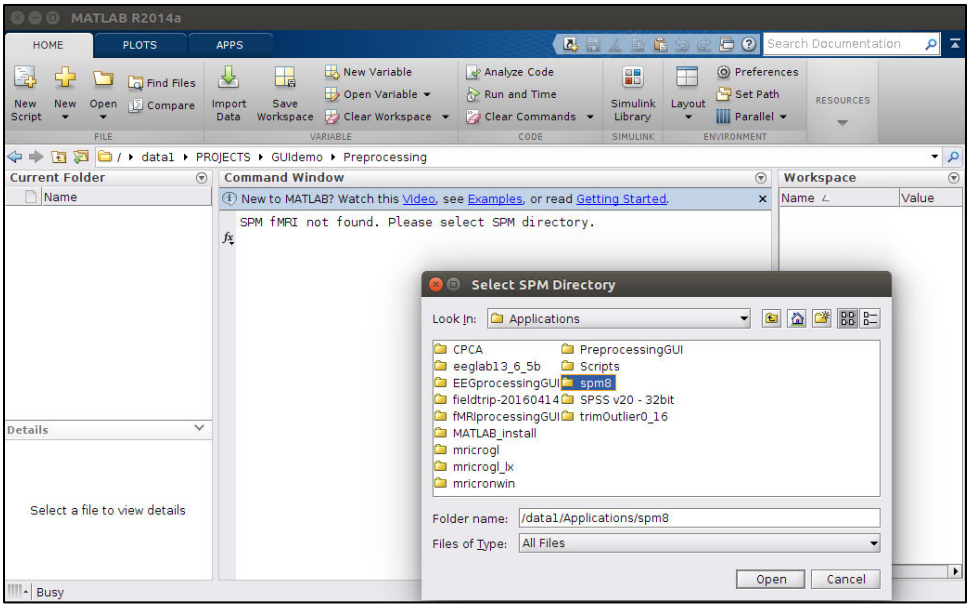


Figure 11. Select SPM folder.

- The GUI and SPM will open (**do not close any windows**):

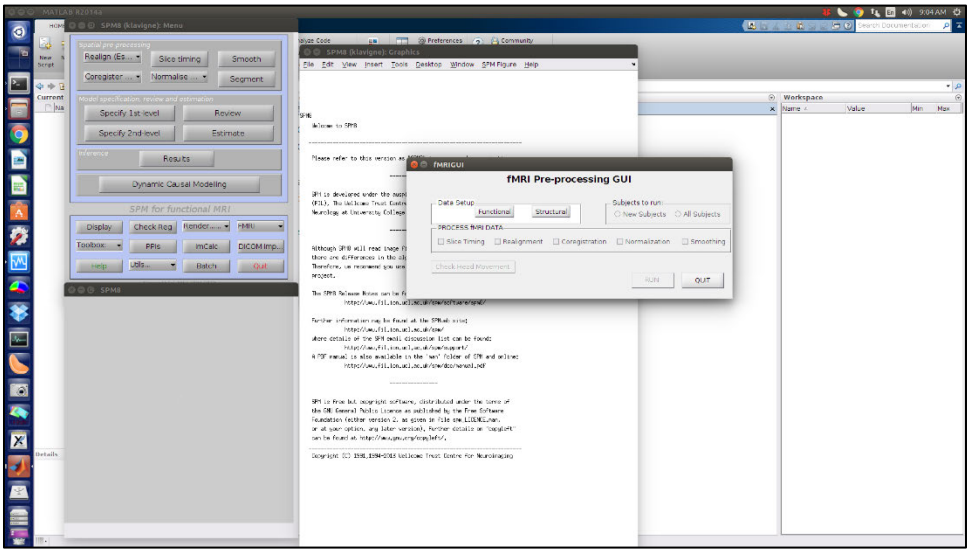


Figure 12. fMRI Preprocessing GUI.

4. Data Setup

In order for the GUI to find the right files and folders to work with, you need to tell it where your data are located as well as the wildcards it should use to find certain files and folders (this is why consistent naming is so important).

a. Functional Data

- Click on the "Functional" button in the Data Setup box
- Select your functional directory (the directory where all the subject folders are, under which are the run folders or functional scans)
- Enter your subject, run, and scan wildcards
    - These are the parts of the folder/filenames that are consistent across all subjects, so if you want to select all schizophrenia patients (whose subject IDs start with S), you could put "S*", or for all healthy controls "H*". This heavily depends on how your data are named and organized.
        - If you want it to go through all folders just put an asterisk "*"
    - Do the same for the run wildcard, such as "run*"
    - For the scan wildcard, you want to also include the extension (.nii or .img). If you converted the data with dcm2nii_convert, the scan wildcard should be f*.nii, but be sure to look at your filenames to confirm.

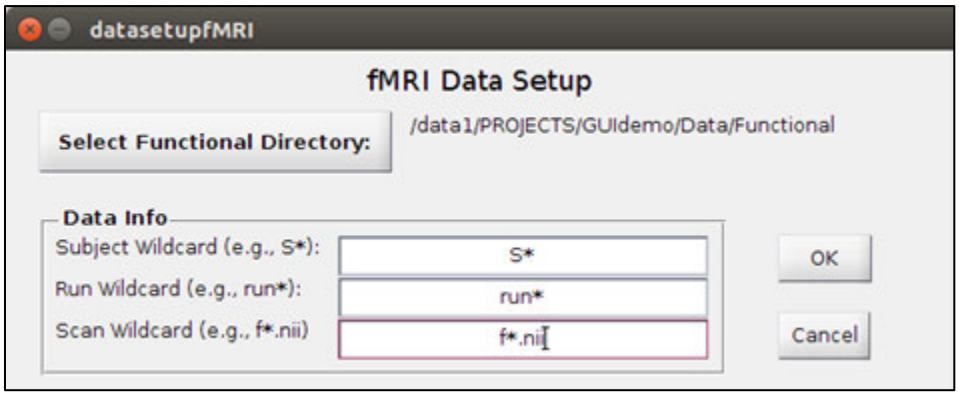


Figure 13. Data Setup

- Click OK.

NOTE: If you have entered in the information correctly and files were found, the functional preprocessing options in the main GUI (slice timing correction, realignment, etc) will become active. If these options are still inactive, it means the GUI didn't find and data and you likely did not perform the setup correctly (either in this section or during data organization).

b. Structural Data

If you have structural data for some or all participants and want to do coregistration and indirect normalization, then you need to prepare and setup your structural data, including (1) visually inspecting the images for movement, artifacts, and incidental findings, (2) reorienting each image, and (3) segmenting each image into gray matter, white matter and cerebrospinal fluid (CSF). Steps 1 and 2 are done outside of the GUI using MRIcron and SPM and are usually done before beginning preprocessing with the GUI. **Quality control is very important here because problems at any of these steps can lead to poor normalization and affect your functional data**. If you do not have structural scans or do not want to use them, skip this section.

1. Visual Inspection

Once data is converted to NIFTI format (after **Section 2. Converting to NIFTI**), you can examine each subject's structural scan visually in MRIcron. This can be done by simply opening MRIcron (click on icon or type mricron in a terminal) and dragging the structural .nii files one by one into the MRIcron window. A faster way to do this is to use a Bash script to automatically load each file one after the other. Once you are done visually inspecting a scan, closing the window will continue the script and lead to the next subject being loaded. You will likely need to change the filename (WOODWARD*.nii) below so it corresponds to your data.

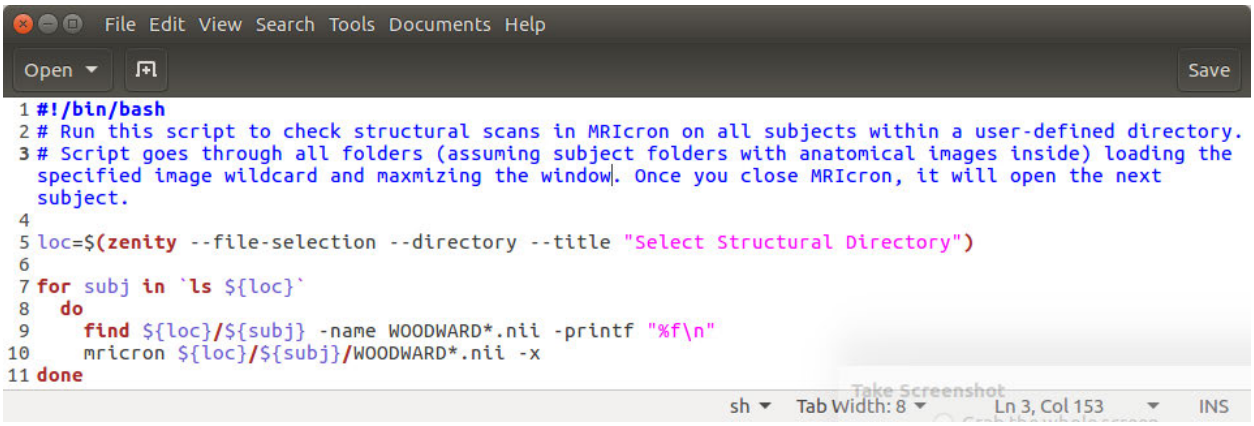


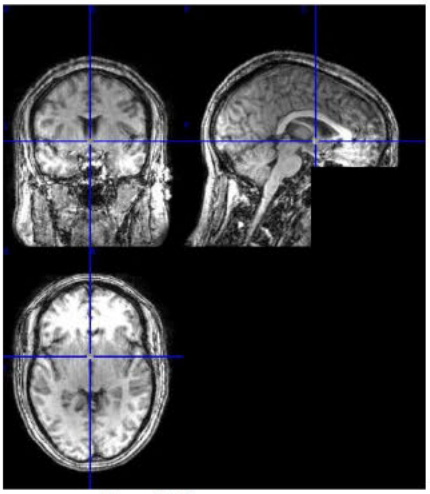
Figure 14. checkstruct script

INSTRUCTIONS

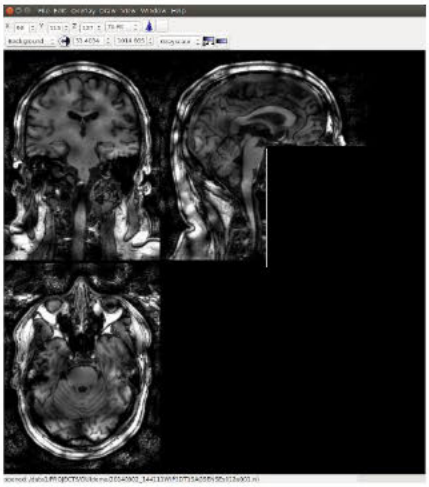
- Change directory to script location: e.g., cd /data1/PROJECTS/GUIdemo/fMRIpreprocessingGUI
- Run script: ./checkstruct
    - MRIcron will need to be downloaded and a link to the program will need to be in your /usr/bin folder (so that the mricron command works)
- Select the structural data folder containing subject folders
- Go through the subject's scan noting any issues with head movement, scanner artifacts or potential incidental findings
    - If the structural scan is not usable, you should move it to a separate folder (creating a new folder called BAD_SCANS or DO_NOT_USE within the subject folder is fine) so the GUI ignores it and runs direct normalization

*****NOTE: If you see a potential incidental finding, it is your responsibility to inform Dr. Woodward and the UBC MRI Technician, so the subject's scan can be forwarded to the radiologist.*****

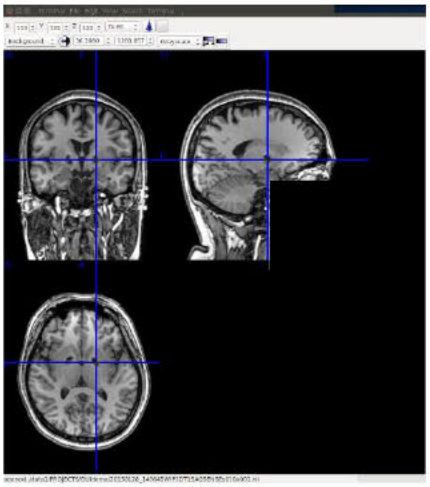
Examples of Problematic Structural Scans



Head Movement
(and intensity non-uniformity)



Scanner Artifact



Incidental Finding

Figure 15. Examples of problematic structural scans

2. Reorientation

If you want to use the structural scans during fMRI processing, you will need to run segmentation prior to normalization. Segmentation will likely fail if the structural scans are not oriented to the SPM templates. Details about reorienting the data can be found in the non-automated preprocessing manual on cfrifs02/WoodwardLab/Manuals/PreprocessingManual. **Note. This step is NOT optional as is described in that manual. Segmentation on scans that are not well oriented will often fail.**

3. GUI Data Setup

Once you have visually inspected the structural scans and reoriented them, you can proceed with the GUI Structural Data Setup.

- Click on the Structural button under Data Setup
- Select your directory and enter your wildcards
    - The "T1 Prefix" includes the entire structural filename up until the subject ID (which must be consistent across subjects). The GUI will add the subject ID to this prefix and search for files that match this and the other parameters.
        - In this example, the structural filenames are "WOODWARD_MCT_3DT1_[Subjet ID]*.ni", so the prefix is "WOODWARD_MCT_3DT1_"
    - Type refers to the type of structural file you want. Dcm2nii created three files:
        - Raw: no prefix added to filename (preferred)
        - Cropped: "c" prefix added to filename
        - Cropped and oriented: "co" prefix added to filename
        - There is also an option for FSL BET (Brain Extracted) images (not tested)
    - Extension refers to whether your structural scans are in .img or .nii format.

- These options simply tell the GUI which files to look for in the structural subject folders.

- Next click on Segmentation

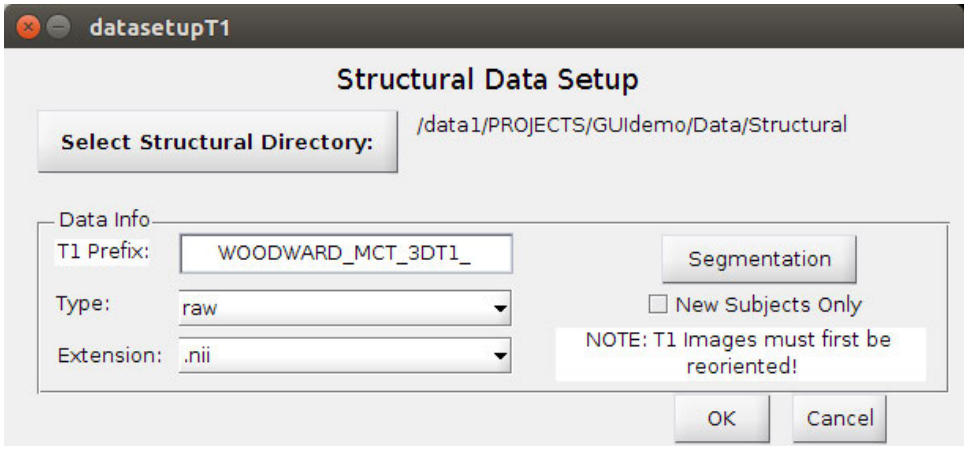


Figure 16. Structural Data Setup

4. Segmentation

This step segments the structural images into grey matter, white matter and CSF, and also corrects for intensity non-uniformities and outputs segmentation parameters to be used in the normalization step. **Your structural images must be properly oriented for this step to work.**

To run segmentation, simply click the Segmentation button in the Structural Data Setup after inputting the relevant parameters described above. This step will output several new files in the subject's structural directory, including the segmented images which should be examined in MRIcron to make sure the process worked (see Figure 17. checkseg script). Any errors will be output to the command window as well as a file called "Segmentation_errors.txt" located in the Segmentation directory in the Preprocessing folder (or whatever your working directory is).

As with visually inspecting the structural images, you can use a BASH script to automate inspection of the segmented images. The script below will load the original structural image and overlay the three segmented images output from the step in MRIcron. You will again likely need to change the structural filenames so they correspond to your data. If there are any problems with segmentation, make sure the data was reoriented correctly.

NOTE: If you have already run segmentation before and are preprocessing additional subjects, you may click on "New Subjects Only". This will look for segmentation output files (m*.nii) and skip the subject if one is found. If you do not click this, it will go through all subjects, overwriting any already-segmented data.

File Edit View Search Tools Documents Help

Open Save

```
#!/bin/bash
# Run this script to check segmentation in MRIcron on all subjects within a user-defined directory.
# Script goes through all folders (assuming subject folders with anatomical images inside) loading the specified image wildcard and the three segmented images (c1, c2, c3) as overlays. Once you close MRIcron, it will open the next subject.

loc=$(zenity --file-selection --directory --title "Select Structural Directory")

for subj in `ls ${loc}`
  do
    find ${loc}/${subj} -name WOODWARD*.nii -printf "%f\n"
    mricron ${loc}/${subj}/WOODWARD*.nii -o ${loc}/${subj}/c1WOODWARD*.nii -o ${loc}/${subj}/c2WOODWARD*.nii -o ${loc}/${subj}/c3WOODWARD*.nii -x
done
```

sh Tab Width: 8 Ln 5, Col 81 INS

Figure 17. checkseg script

Examples of Segmented Images Viewed in MRIcron

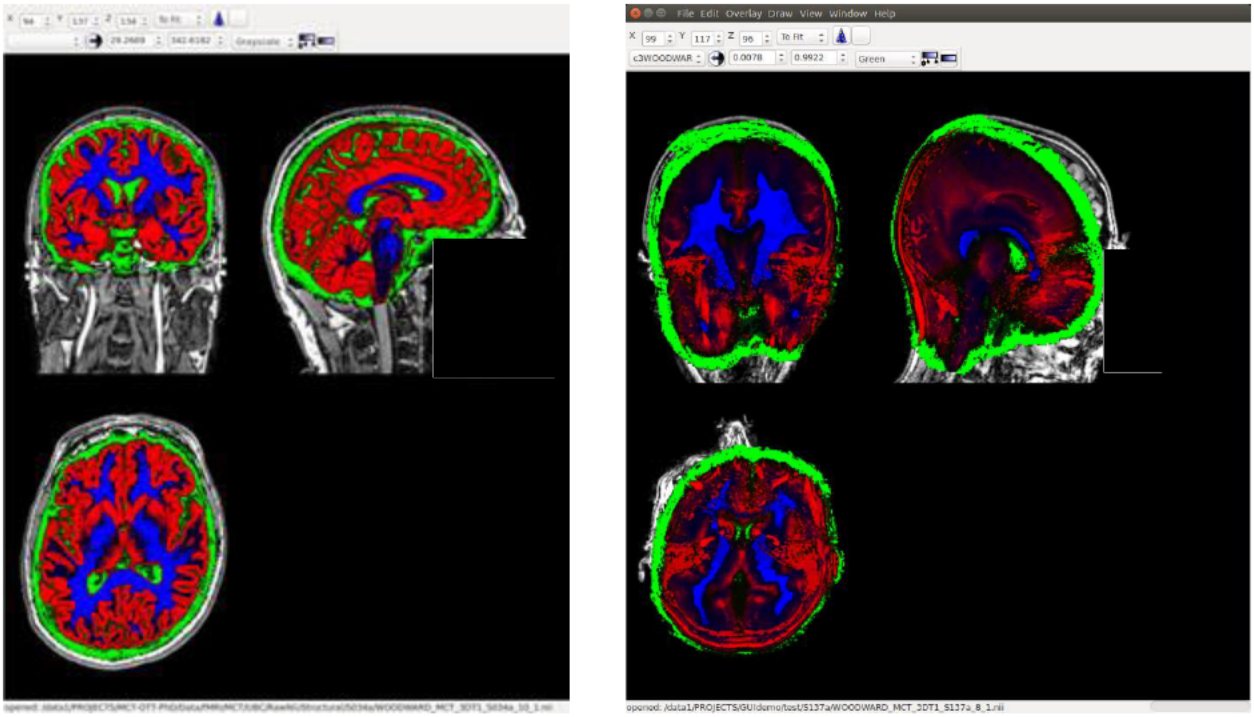


Figure 18. Good vs Bad Segmentation

GOOD **BAD**

- Once you are satisfied with the segmentation results, click OK in the Structural Data Setup to return to the main GUI.

5. Preprocessing

The rest of the preprocessing steps can be run one at a time or all at once, which will simply run the next step after all subjects have completed the previous step.

First, decide whether you want to preprocess all subjects or only subjects who have not been run before by clicking on “All Subjects” or “New Subjects” under Subjects to Run in the top right corner of the GUI. If you click on New Subjects, it will first search for the output files of each step before running a subject (e.g., a*.nii files for Slice timing, mean*.img files for Realignment, etc), and will skip that subjects if files are found. This is useful if you are adding subjects to data that has already been preprocessed.

To run a step, simply click on the checkbox next to the step name. If there are any options for you to define, a pop-up window will prompt you to input them. The default values listed in the pop-up windows are for the OTT/MCT Study in the UBC scanner. Below are the options you will be asked to define in each step. Any additional parameters follow the instructions listed in the non-automated preprocessing manual.

a. Slice Timing Correction

- Reference Slice

b. Realignment

c. Coregistration

d. Normalization

e. Smoothing

6. Data Checking

a. Segmentation

(See above Fig 17 Fig 18)

b. Realignment

Click on Check Head Movement and see other manual

c. Coregistration

Look at .ps file

d. Normalization

Click on Check Normalization

7. Common Errors