CATNAP User Manual v1.0

July 6, 2006 Nera-Lee Patel and Bennett Landman

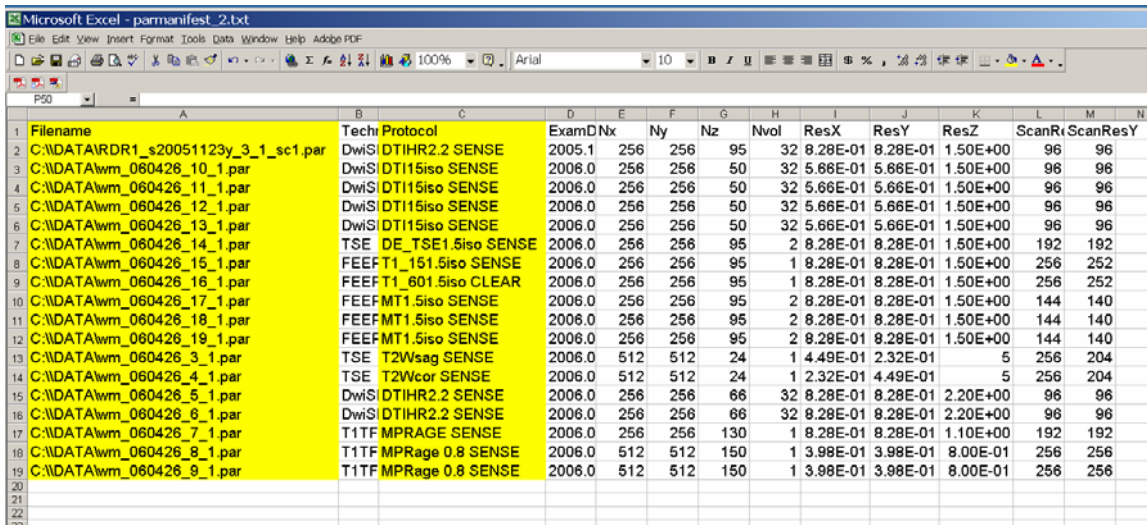Note:
- Items in green text are only sample value. Your values may be different.

I. CreateParManifest -- Creates a spreadsheet to show all data files:
1) Open MATLAB.
2) Set path to where your data is stored.
3) If the function createParManifest is not in current directory, map its folder to a network drive and then type in **addpath(genpath('F:\');**
4) Use the **createParManifest** function.
5) Now, a text file has been created containing information about each of the data files.

II. Viewing Data Files:
1) Open the parManifest text file in Excel.
2) To choose the appropriate files, concentrate on the first and third columns, the filename and the protocol:

Automatically, we have the type of file with its corresponding name, which makes it possible to distinguish the files for selection in CATNAP.

3) The following is a table of the various protocol types

Protocol Type	Meaning	
DTI1.5	diffusion with resolution 1.5	Diffusion
DTIHR2.2	diffusion with resolution 2.2	
DE_TSE1.5	dual scan echo	
T1_151	Absolute T1 with flip angle 15	
T1_601	Absolute T1 with flip angle 60	
MPRAGE	MPRAGE scan with 1mm res	Structural Files
MPRage 0.8	MPRAGE scan with 0.8mm res	
T2Wsag	Scout pulses	
T2Wcor	Scout pulses	
MT	Magnetic Transfer	

III-IV. Generating CATNAP Processing Script and Running
1) In the MATLAB command window, start CATNAP by typing in **CATNAP**, and the following window appears:

Diffusion File List

Structural File List

Alignment File

Select .m
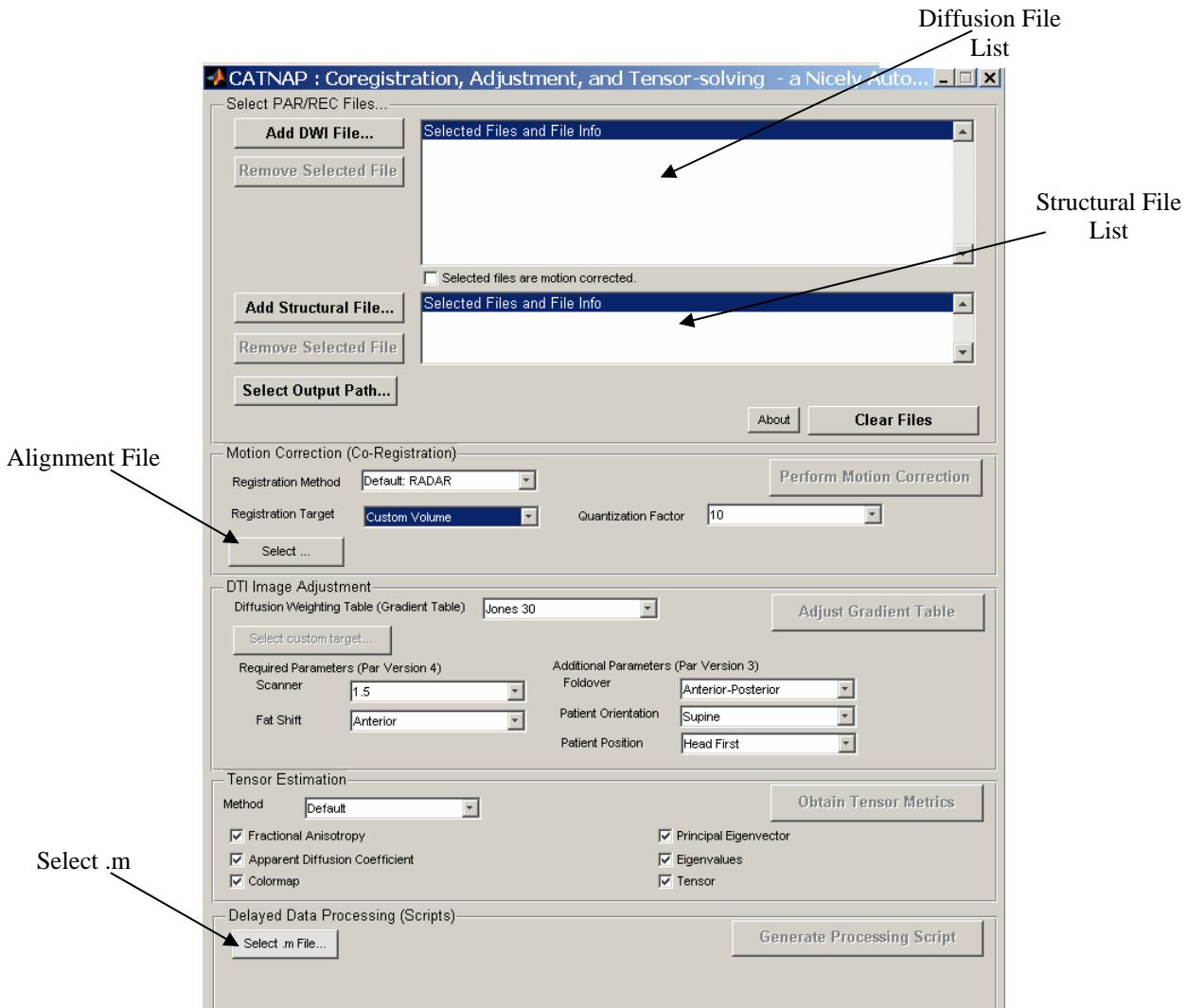
2) First add diffusion files with the <u>higher resolution</u> to the Diffusion File List and add all structural files (excluding the scout pulses) to the Structural File List. Be patient, as loading files takes a little while.

3) The following parameters record how the subject was scanned. Choosing them correctly allows CATNAP to adjust and register the images correctly:
Registration Method: 'Default: RADAR'
Registration Target: 'Custom Volume'
Quantization Factor: 1.0
Gradient Table: 'Jones 3D'
Scanner: '3.0'
Fat Shift: Posterior
Foldover: Anterior-Posterior
Patient Orientation: Supine
Patient Position: Head First
Method: Default
Tensor Estimation check boxes: All Checked

4) Click the select button to select the Alignment file, and choose the file that had the DE-TSE protocol.

5) Click the Select Output Path… button to choose the location of where data will be stored after it gets processed.

6) Click the Select.m File button and type in a name for the processing script as an .m file i.e. myfilestoprocess.m.

7) Finally, click the Generate Processing Script button, and the .m file will appear where specified.

8) Now, the next resolution set of diffusion files need to be added to the processing script, so on the CATNAP window, click the Clear Files button.

9) Add the next highest resolution files to the Diffusion Files List.

10) Leave the same parameters set.

11) Click the Generate Processing Script button, and the remaining diffusion files will be appended to the processing script.

12) To process the files, simply open the processing script in MATLAB and run it. **Note that data processing is a very time-consuming operation, so it is best to first generate the processing script and then run it overnight.**

13) The processed files will appear in the folder that was specified in step 8, and within this folder, CATNAP will create the subfolders 'dti', 'xfm' and 'scaled' to further organize the data.

V. To Process Structural Files
1) Set path to the path where structural files are stored.
2) Open the parManifest file created earlier.
3) Structural files are processed with the following commands:

Protocol	Command
T1	**ComputeAbsoluteT1**
DE-TSE	**ComputeAbsoluteT2**
MPRAGE	**ComputeMean**
MT	**ComputeMTR**

VI. Masking: float and argb data

*Colormap(.cmap) images are argb, while all the rest are floats.*

Creating Mask
1) Open MATLAB and set the current directory to the subfolder 'scaled' that contains the processed data.
2) At command prompt, type in **dir *.rec** to generate the list of .rec files.
3) Look for the .rec file corresponding to the first 2.2 resolution diffusion file listed in the ParManifest text:
> At the prompt load this file with the **loadPARREC** function with the first argument containing the filename corresponding to the 2.2 resolution file (do not include the file extension) and the second argument being 1, indicating that the image mask will be produced with the first b0 volume. The command will resemble the following:
> b0vol=loadPARREC('wm_061605_5_1_sc1', 1);
4) To check what type of data is stored in b0vol, type in **fields(**b0vol**)**.
5) We see that b0vol stores several fields, including the one of interest, scans.
6) Run a movie of this volume with the **floatSliceMovie** function, with the only argument being the variable the scan was stored into, and the scan number, 1:
> floatSliceMovie(b0vol.scans{1});
7) Now, view the image and determine the threshold at which the outside of the matter becomes distinguishable from the actual brain image. Be sure to note the order of magnitude of this threshold (encircled in yellow below).
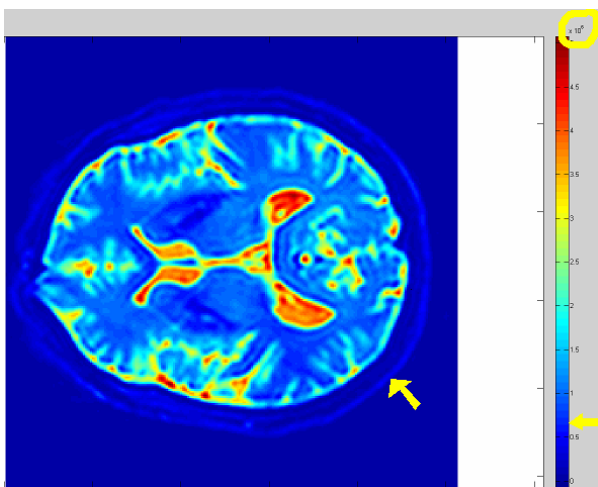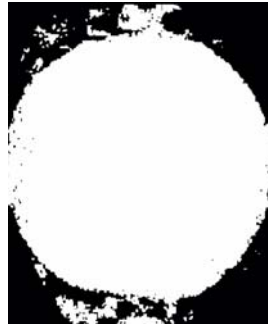
8) Now view the video again with the floatSliceMovie command, adding the resolution and 'bone' to the arguments.
9) Observe the video, looking for uniformity in the cerebellum – no holes. The following are examples to better illustrate the point. They are images of slice 40.

High Threshold   Correct Threshold   Low Threshold

If you observe holes, the threshold was too high. If you cannot identify structures in the brain, the threshold was too low. Repeat steps 6-9 until the ideal threshold is chosen.

10) Now, create the mask by using the voxels of higher threshold:
```
mask= b0vol.scans{1}>0.5e6;
```
Float Images
- To load the float images, use the **loadFloat** function.
- To mask loat images and write them to a file, use the **MaskWrite** function.

ARGB Images
- Load the cmap images with the **loadCmap** function
  *Note there are strict memory requirements to run this function –if memory error occurs, quit MATLAB and restart. Do not use unnecessary java applications within MATLAB.
- Use the **maskcmap** function to mask the image and write data to a file.

Viewing images, use the **imagesc** function with the following parameters:

Protocol	
AbsT1	imagesc(absT1(:,:,50), [0 4000]);
AbsT2	imagesc(absT2(:,:,50), [0 200]);
MPRAGE	imagesc(mprage(:,:,50));
MT	imagesc(mt(:,:,50), [0 .55]);
fa	imagesc(fa(:,:,50));
cmap	imagesc(cmap(:,:,1,50));