CAT12 process on T1 data for the study of 'Disease Progression Modeling'

Last update May 5, 2022

Yuchao Jiang, Wei Cheng, Jianfeng Feng
yuchaojiang@fudan.edu.cn , wcheng.fdu@gmail.com , jianfeng.feng@warwick.ac.uk

The following steps will allow you to perform a standard voxel-based morphometry (VBM) on the structural MRI data (i.e., high-resolution T1 image) and then extract 17 ROI values from the AAL3 atlas for each subject. The 17 ROIs include frontal lobe, temporal lobe, parietal lobe, occipital lobe, insula, cingulate, sensorimotor, Broca's area, cerebellum, hippocampus, parahippocampus, amygdala, caudate, putamen, pallidum, nucleus accumbens and thalamus.

Running VBM by standard version CAT12 in MATLAB

Download and installation

- The CAT12 Toolbox runs within SPM12 and Matlab versions 7.4 (R2007a) to 9.7 (R2019b). That is, SPM12 must be installed and added to your Matlab search path before the CAT12 Toolbox can be installed (https://www.fil.ion.ucl.ac.uk/spm/software/spm12/).
- Download the **latest CAT12 toolbox (cat12_latest.zip)** from the website (http://www.neuro.uni-jena.de/cat12/) and unzip the CAT12 Toolbox. You will get a folder named "cat12", which contains various Matlab files and compiled scripts. Copy the folder "cat12" into the SPM12 "toolbox" folder. Please remove all older versions of CAT12 before installation.
- PS: CAT12 manual is available at the website (http://www.neuro.uni-jena.de/cat12/CAT12-Manual.pdf).
- If you don't have MATLAB, we also provide the standalone version of CAT12. Please see following part of '***Running VBM by standalone version CAT12 without MATLAB***'. The advantage of the standalone version is that no Matlab license is required. Only the (free) Matlab Runtime has to be downloaded. We highly recommend using standalone version because it is easier to use and more stable.

Starting and Toolbox

- Start Matlab.
- Start SPM12 (i.e., type "spm fmri")
- Select "cat12" from the SPM menu (see Figure 1). You will find the drop-down menu between the "Display" and the "Help" button (you can also call the Toolbox directly by typing "cat12" on the Matlab command line). This will open the CAT12 Toolbox as an additional window (Fig. 2).

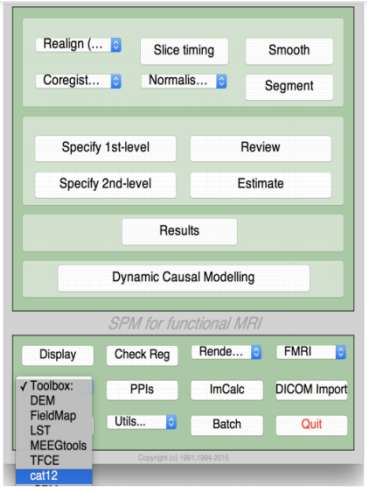

Figure 1: SPM12 menu

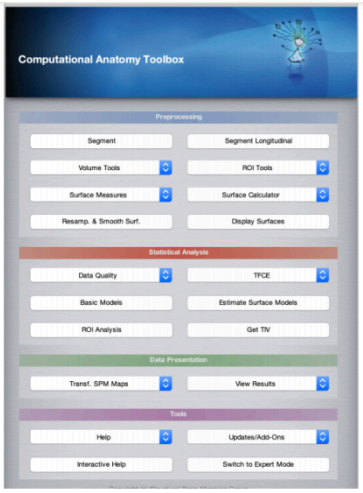

Figure 2: CAT12 Window

Processing Data by VBM

- Start '**Segment**', see Fig 3.
- '**Volumes**' -> *input all subject's T1 images*.
- '**Split job into separate processes**' -> Number of parallel operations. If you do not want to run

processes in the background then set this value to 0.

- '**Surface and thickness estimation**' -> Please set to 'No' in order to reduce computing time
- '**Process Volume ROIs**' -> Please set to 'Altases/own atlas maps/aal3.nii'. see Fig 4.
- Keep the default values for other parameters
- Run Batch
- Ps: It takes about 30~40 minutes to process one subject's data by the PC (i5-8500 CPU @ 3.00GHz, 16G RAM) without parallel operations.

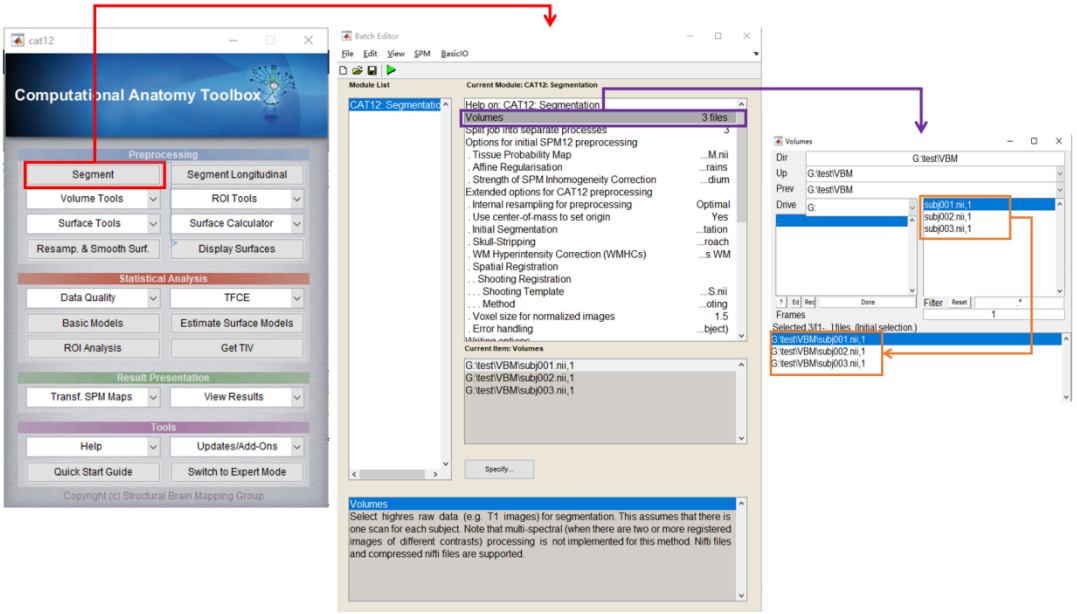

Figure 3. VBM parameters (a)

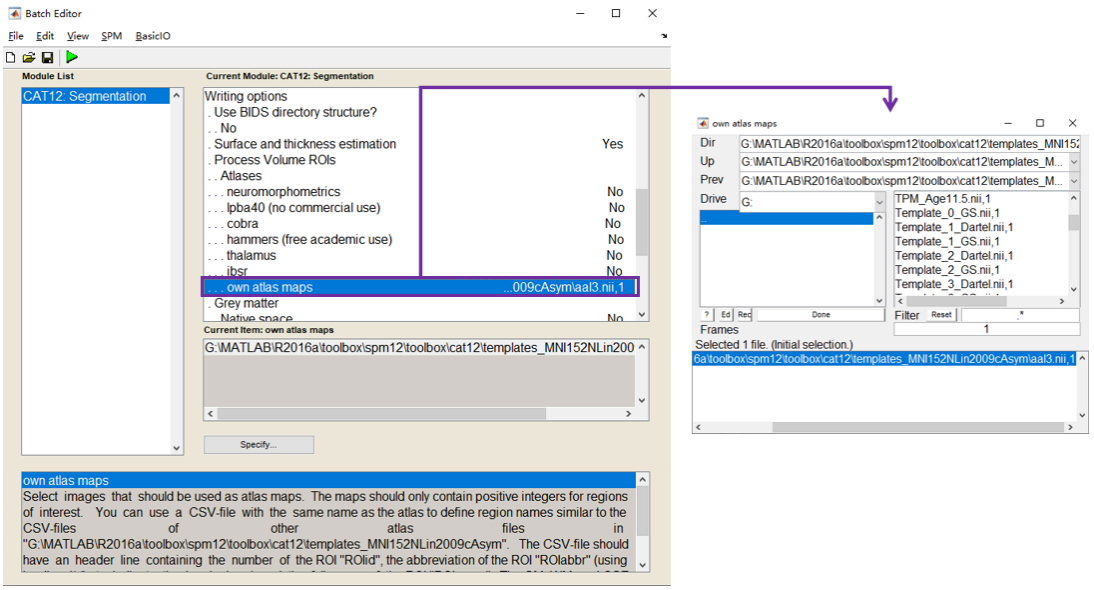

Figure 4. VBM parameters (b)

Estimate Mean ROI values of AAL3 atlas

- Start '**ROI TOOLS**', Choose 'Estimate Mean Value inside ROI for External Analysis'. see Fig 5.
- '**XML files**' -> *input all subject's .../label/catROI_*.xml*.
- '**Output directory**' -> Select a directory where files are written.
- Run Batch

- The output will be saved in ‘ ROI_catROI_aal3_Vgm.csv ’.

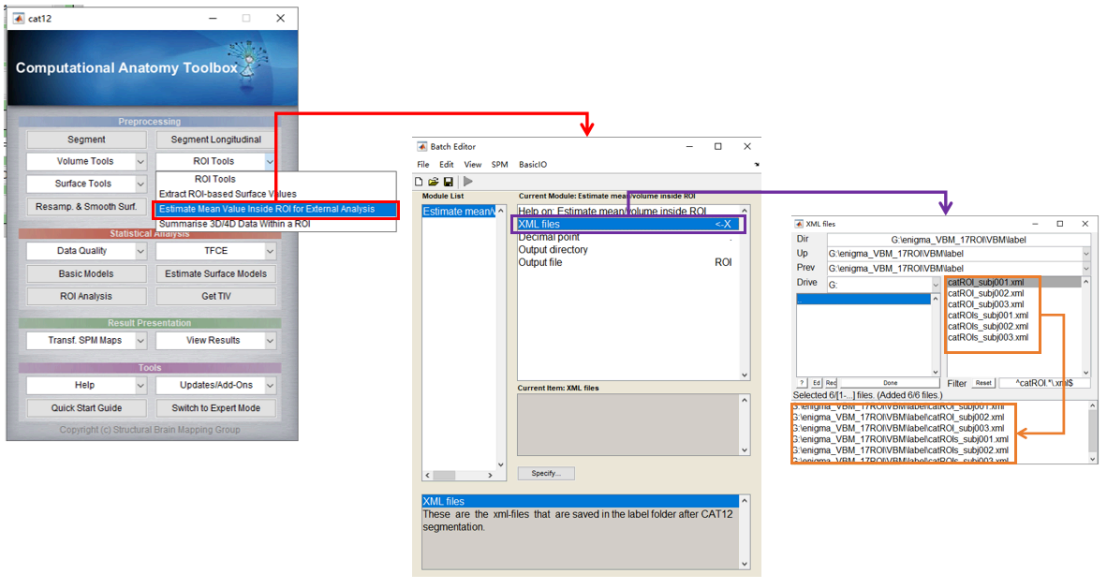

Figure 5. Estimate Mean ROI values of AAL3 atlas and save in csv-file

Estimate Total Incranial Volume (TIV)

- Start ‘**Get TIV**’. see Fig 6.
- ‘**XML files**’ -> *input all subject’s .../report/cat_*.xml*.
- Run Batch
- The output will be saved in ‘ TIV.txt ’.

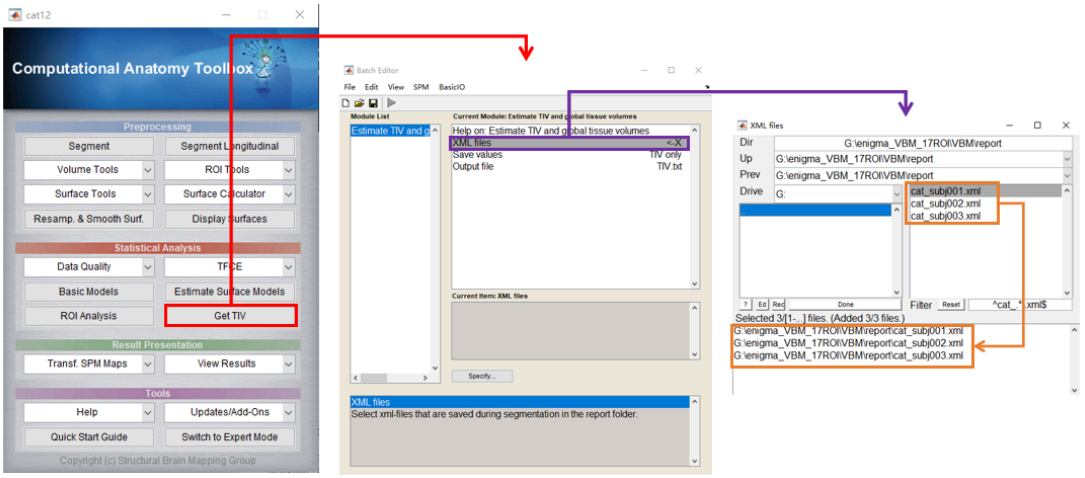

Figure 6. Estimate TIV and save in TIV.txt

Running VBM by standalone version CAT12 without MATLAB

If you don't have MATLAB, we also provide the standalone version of CAT12 without MATLAB. Standalone version of CAT12 allows to run CAT12 without the need for Matlab. (https://sites.google.com/view/enigma-cat12/enigma-cat12/standalone)

Installation of the standalone version of CAT12

- Download latest CAT12 standalone version for macOS (cat12_latest_R2017b_MCR_Mac.zip), Linux (cat12_latest_R2017b_MCR_Linux.zip), or Windows (cat12_latest_R2017b_MCR_Win.zip) (copy the redirected link to your browser if the download does not start automatically or load the latest file for your system with the name cat12_latest_R2017b_MCR*). (http://www.neuro.uni-jena.de/cat12/)
- Open the MCR_v93.webloc link in the installed folder, which automatically downloads the **Matlab Runtime for R2017b**. Or download from (https://ssd.mathworks.com/supportfiles/downloads/R2017b/deployment_files/R2017b/installers/glnxa64/MCR_R2017b_glnxa64_installer.zip)
- Install the Matlab Runtime with the "v93" folder preferably in the same folder. https://www.mathworks.com/help/compiler_sdk/dotnet/install-the-matlab-runtime.html

2. Start the MATLAB Runtime installer.

Platform	Steps
Windows	Double-click the file `setup.exe` from the extracted files to start the installer.

Linux	At the terminal, type:
	`sudo -H ./install`
	Note
	You may need to allow the root user to access the running X server:
	`xhost +SI:localuser:root`
	`sudo -H ./install`
	`xhost -SI:localuser:root`

macOS	At the terminal, type: `./install` **Note** You may need to enter an administrator user name and password after you run `./install`.

Note

If you are running the MATLAB Runtime installer on a shared folder, be aware that other users of the share may need to alter their system configuration.

For example how to install ***standalone version of CAT12*** **and Matlab Runtime** on a Linux Centos 7 system:

I have downloaded the CAT12 standalone version: CAT12.8.1_r1980_R2017b_MCR_Linux.zip ,
and unzip it at the terminal using the unzip command.
unzip CAT12.8.1_r1980_R2017b_MCR_Linux.zip
and rename the directory **CAT12.8.1_MCR** .

Then, I double-click the file **CAT12.8.1_MCR/MCR_v93.webloc**, which automatically downloads the Matlab Runtime for R2017b (i.e., MCR_R2017b_glnxa64_installer.zip). Or downloads at the terminal, type:
wget
https://ssd.mathworks.com/supportfiles/downloads/R2017b/deployment_files/R2017b/installers/glnxa64/MCR_R2017b_glnxa64_installer.zip

Then, I move the MCR_R2017b_glnxa64_installer.zip inside this folder CAT12.8.1_MCR, unzipped it at

the terminal, type:
cd /.../CAT12.8.1_MCR
unzip CAT12.8.1_r1980_R2017b_MCR_Linux.zip

Then, type:
./install
Or type:
./install -mode silent -agreeToLicense yes -destinationFolder /.../CAT12.8.1_MCR/v93
And install Matlab Runtime to a new directory **/.../v93** and move **v93** inside the foloder **CAT12.8.1_MCR**

Now it says “Preparing installation files …”
“Installing …”

Use of the standalone version of CAT12
For the following examples, we assume that the CAT12 standalone version was installed in "/software/CAT12.8.1_MCR" and the Matlab Runtime in "/software/CAT12.8.1_MCR/v93" and your T1-data are located in "/data/enigma-data/raw" and are named "sub*.nii.gz" or ''sub*.nii''. See the help text of standalone/cat_standalone.sh for more examples and a description of additional options.

Preprocessing
/software/CAT12.8.1_MCR/standalone/cat_standalone.sh -m /software/CAT12.8.1_MCR/v93 \
-b /software/CAT12.8.1_MCR/standalone/cat_standalone_segment_enigma.m \
/data/enigma-data/raw/sub*.nii

Conventional preprocess takes ~50 minutes for one subject using above code. Because surface estimation is not necessary in this study, we suggest to preprocessing data using the following code, which could greatly reduce the computation time (~10 minutes per subject for macOS).
 # Preprocessing without surface estimation (suggested)
/software/CAT12.8.1_MCR/standalone/cat_standalone.sh -m /software/CAT12.8.1_MCR/v93 \
-b /software/CAT12.8.1_MCR/standalone/cat_standalone_segment_enigma.m -a \
"matlabbatch{1}.spm.tools.cat.estwrite.output.surface = 0;" \
/data/enigma-data/raw/sub*.nii

Parallel computation (Optional)
The standalone version of CAT12 supports parallel computation. Parallelize CAT12 ENIGMA preprocessing by splitting all sub*.nii files into 8 jobs (processes) and save log file in /tmp folder. Please use the script standalone/cat_parallize.sh:
/software/CAT12.8.1_MCR/standalone/cat_parallelize.sh -p 8 -l /tmp \
-c "-m /software/CAT12.8.1_MCR/v93 -b
/software/CAT12.8.1_MCR/standalone/cat_standalone_segment_enigma.m" \
/data/enigma-data/raw/sub*.nii

Estimate mean volume values inside ROI
Setting the current directory for saving the result (csv-file)
cd /data/enigma-data/raw/CAT12.8.1/

Extracting mean volumes (in *.xml) for all selected data and saved in a csv-file. The csv-file is named "ROI_" followed by the atlas name and the name of the measure (e.g. Vgm). Please note the multiple quotes to define the ROI name. Please note the multiple quotes for parameter a1.
/software/CAT12.8.1_MCR/standalone/cat_standalone.sh -m /software/CAT12.8.1_MCR/v93 \
-b /software/CAT12.8.1_MCR/standalone/cat_standalone_get_ROI_values.m \

```
/data/enigma-data/raw/CAT12.8.1/label/catROI*.xml -a1 " 'ROI' "
```

Estimate total intra-cranial volume (TIV)

Setting the current directory for saving the result (csv-file)

```
cd /data/enigma-data/raw/CAT12.8.1/
```

Save TIV values for all selected data in TIV.txt.

```
/software/CAT12.8.1_MCR/standalone/cat_standalone.sh -m /software/CAT12.8.1_MCR/v93 \
-b /software/CAT12.8.1_MCR/standalone/cat_standalone_get_TIV.m \
/data/enigma-data/raw/CAT12.8.1/report/cat_*.xml -a1 " 'TIV.txt' " -a2 "1" -a3 "1"
```

Save TIV values for all selected data in TIV.txt. Please note the multiple quotes to define the output name with a1. The parameter a2 allows you to save only the TIV (1) or additionally to save also the global volumes for GM, WM, CSF, and WM hyperintensities (0). With parameter a3 you can add file names to the 1st column: 0 - save values only; 1 - add file name; 2 - add folder and file names. Please note that parameter a3 is available only from version r1987 and you must add the parameter a2 in addition.

Extract 17 ROI values from AAL3 atlas

The AAL3 regional volumes were saved in the ROI_ aal3_Vgm.csv (Figure 7).

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	roiID	1	2	3	4	5	6	7	8	9	10	11	12
2	names	lPreCG	rPreCG	lSFG	rSFG	lMFG	rMFG	lIFGoperc	rIFGoperc	lIFGtriang	rIFGtriang	lIFGorb	rIFGorb
3	subj001	12.1758	12.2474	14.7598	14.7479	16.4613	17.2204	4.7082	4.6934	8.4856	6.4706	2.3805	2.4221
4	subj002	11.6806	11.1551	13.292	14.9798	16.1126	15.9039	3.3176	4.6916	7.3663	6.796	2.4951	2.3174
5	subj003	11.4648	11.5364	14.0488	14.0369	15.7503	16.5094	3.9972	3.9824	7.7746	5.7596	1.6695	1.7111
6	subj004	11.5336	11.0081	13.145	14.8328	15.9656	15.7569	3.1706	4.5446	7.2193	6.649	2.3481	2.1704
7													

Figure 7. ROI_ aal3_Vgm.csv

Now we only need to extract the areas of interest and merge some of them into the following 17 Areas (see the following table). Here, we provide a R script to achieve the sum.

Areas	Labels	Which AAL3 regions were merged as the target areas of interested
1	Hippocampus	SUM (41:42), i.e., the sum of column 41 to 42 in ***ROI_ aal3_Vgm.csv***
2	Parahippocampus	SUM (43:44)
3	Amygdala	SUM (45:46)
4	Caudate	SUM (75:76)
5	Putamen	SUM (77:78)
6	Pallidum	SUM (79:80)
7	Thalamus	SUM (121:150)
8	Accumbens	SUM (157:158)
9	Cingulate	SUM (37:40 and 151:156)
10	Frontal	SUM (3:6 and 11:14 and 17:32)
11	Parietal	SUM (63:74)
12	Temporal	SUM (59:60 and 83:94)
13	Occipital	SUM (47:58)
14	Insula	SUM (33:34)
15	Cerebellum	SUM (95:120)
16	Sensorimotor	SUM (1:2 and 15:16 and 61:62)
17	Broca area	SUM (7:10)

----------**R script to sum ROIs**----------

```
AAL3_id <- c(1:170)
id_df <- as.data.frame(AAL3_id)
id_df$new_id[id_df$AAL3_id %in% c(41:42)] <- 1
id_df$new_id[id_df$AAL3_id %in% c(43:44)] <- 2
id_df$new_id[id_df$AAL3_id %in% c(45:46)] <- 3
id_df$new_id[id_df$AAL3_id %in% c(75:76)] <- 4
id_df$new_id[id_df$AAL3_id %in% c(77:78)] <- 5
id_df$new_id[id_df$AAL3_id %in% c(79:80)] <- 6
id_df$new_id[id_df$AAL3_id %in% c(121:150)] <- 7
id_df$new_id[id_df$AAL3_id %in% c(157:158)] <- 8
id_df$new_id[id_df$AAL3_id %in% c(37:40, 151:156)] <- 9
id_df$new_id[id_df$AAL3_id %in% c(3:6, 11:14, 17:32)] <- 10
id_df$new_id[id_df$AAL3_id %in% c(63:74)] <- 11
id_df$new_id[id_df$AAL3_id %in% c(59:60, 83:94)] <- 12
id_df$new_id[id_df$AAL3_id %in% c(47:58)] <- 13
id_df$new_id[id_df$AAL3_id %in% c(33:34)] <- 14
id_df$new_id[id_df$AAL3_id %in% c(95:120)] <- 15
id_df$new_id[id_df$AAL3_id %in% c(1:2, 15:16, 61:62)] <- 16
id_df$new_id[id_df$AAL3_id %in% c(7:10)] <- 17

id_df <- na.omit(id_df)
rownames(id_df) <- NULL
data_raw <- read.csv("/.../ROI_aal3_Vgm.csv", header = T, stringsAsFactors = F)
```

```
data_new <- matrix(nrow = nrow(data_raw), ncol = 17)
for (i in sort(unique(id_df$new_id))) {
  data_new[, i] <- rowSums(data_raw[, id_df$AAL3_id[id_df$new_id == i] + 1])
}
data_new <- cbind(data_raw[, 1], as.data.frame(data_new))
colnames(data_new) <- c("names", "Hippocampus", "Parahippocampus", "Amygdala", "Caudate", "Putamen", "Pallidum", "Thalamus", "Accumbens", "Cingulate", "Frontal", "Parietal", "Temporal", "Occipital", "Insula", "Cerebellum", "Sensorimotor", "BrocaArea")
write.csv(data_new, "/.../ROI_17_Vgm.csv", row.names = F, na = " ")
```
