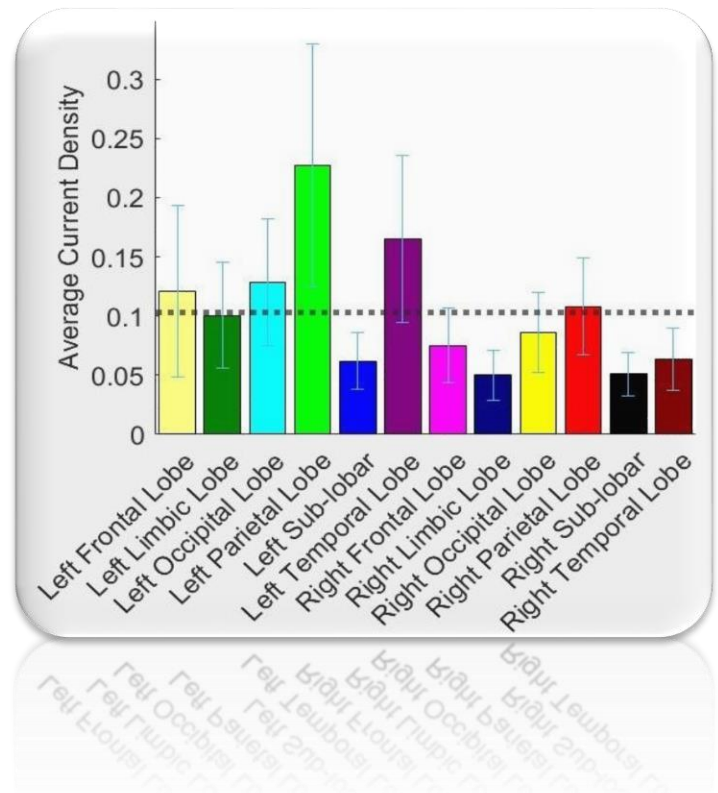
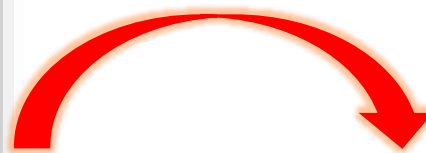
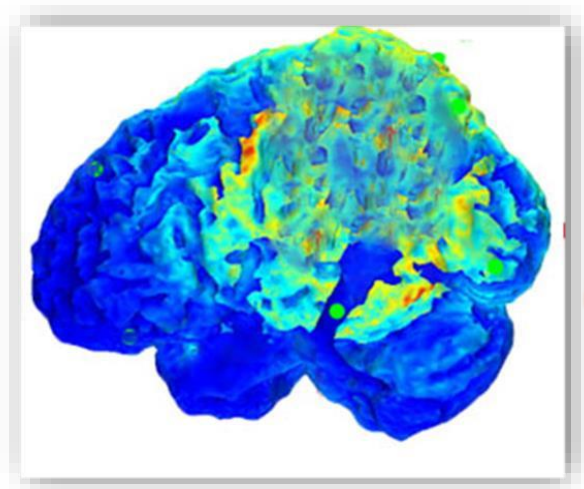


# **Systematic Approach For tDCS Analysis (SATA)**

SATA version 1.0

User's Manual (August 2019)



In collaboration with

# Table of Contents

Chapter 1: Introduction and Background .....	2
1.1 Introduction.....	2
1.2 Background.....	2
Chapter 2: Installation .....	3
2.1 Software Download .....	3
2.1.1 Install MATLAB and Java Runtime Environment (JRE) .....	3
2.1.2 Downloading SATA .....	3
2.2 Setting up SATA .....	4
2.2.1 Introduction to the software COMETS.....	4
2.2.2 Introduction to the software ROAST.....	4
2.3 Installing the SATA app .....	5
Chapter 3: Using SATA .....	5
3.1 Running SATA .....	5
3.2 How to use SATA.....	6
3.2.1 Generating the data in COMETS and saving the output.....	7
3.3 Using the Functions in SATA GUI .....	9
3.3.1 Average CD per Lobe .....	9
3.3.2 Coordinates with MCD .....	16
3.3.3. Overlapping Coordinates .....	17
3.4 Using the SATA Functions in MATLAB Command Window .....	20
3.5 Using Roast from SATA.....	21
3.5.1 Choosing ROAST generated data for SATA analysis .....	24
3.5.2 Importing pre generated ROAST data into SATA .....	24
Acknowledgement.....	26
References.....	27

# Chapter 1: Introduction and Background

## 1.1 Introduction

SATA stands for Systematic Approach for tDCS Analysis. SATA is essentially a post processing software that acts on the outputs from two conventionally used tDCS software like COMETS (Lee, Jung, Lee, & Im, 2017) and ROAST (Huang, Datta, Bikson, & Parra, 2019). The principles of SATA are detailed in the paper (Bhattacharjee et al., 2019)

SATA has been developed using MATLAB vR2019a\*\* (The MathWorks Inc., Natick, MA, US). It takes inputs from the software COMETS and ROAST. It uses FieldTrip (Oostenveld, Fries, Maris, & Schoffelen, 2011) and SPM12 (“SPM - Statistical Parametric Mapping,”) to post process the input data. It uses the brain template from Talairach Client (Lancaster et al.1997, 2000) to superimpose the processed output on the cortical areas of the brain.

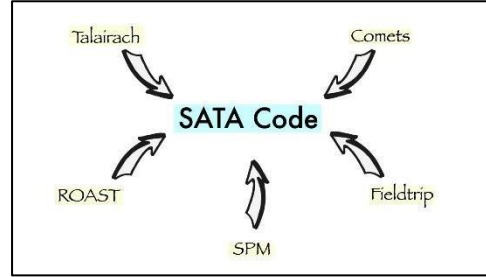


Fig. 1: Modularity of SATA

For any details or queries please contact:

[clinicalbrainlab@gmail.com](mailto:clinicalbrainlab@gmail.com)

## 1.2 Background

Different cognitive behavior have different pathways in the brain associated with them. Sometimes these pathways are situated very close to each other. For example reading has two pathways in the brain, the dorsal pathway comprising the superior temporal gyrus (STG), temporal-parietal angular gyrus (AG), supramarginal gyrus (SMG), and Inferior frontal gyrus (IFG) is involved in grapheme to phonology conversion (sublexical) and the ventral pathway consisting of the fusiform gyrus (FFG) to middle\inferior temporal gyrus (MTG) is involved in lexico-semantic functions associated with reading. To specifically target either pathway independently, the tDCS montages will need to be configured to maximize selectivity to the cortical regions in the respective pathways. This is where SATA comes into the picture. It has 3 functions, (i) average magnitude of current density (MCD) in each cortical lobe, (ii) number of overlapping coordinates, and (iii) cortical areas with high MCD. The measures (i) and (iii) help ascertain the current spread by each montage within a cortical lobe, and (ii) verifies the overlap of the spread of current between a pair of montages. SATA specifically aims to facilitate the montage selection process for selectively stimulating the two reading routes. We hope the systematic approach might be helpful in selecting appropriate tDCS montages for other cognitive behavior. For details refer to the paper (Bhattacharjee et al., 2019)

# Chapter 2: Installation

## 2.1 Software Download

### 2.1.1 Install MATLAB and Java Runtime Environment (JRE)

- 1) We have tested the SATA.app on MATLAB version R2019a and R2017a. Moreover, SATA functions are compatible with Linux and Mac. However, we are not sure about the appropriate working of the .app for versions below MATLAB R2017a. Also the user must ensure the compatibility of the software COMETS and ROAST in the operating systems linux or Mac before using SATA. Nevertheless, if the user has pre-generated data from COMETS or ROAST, he/she may opt to work with the command lines of SATA (refer to section 3.4).
- 2) Currently version of SATA uses only the inbuilt head model of both COMETS and ROAST.
- 3) MATLAB vR2019a can be downloaded at <https://in.mathworks.com/downloads/>.
- 4) To download JRE, please follow the link <https://www.java.com/en/download/>
  - Click Java Download, this will redirect you to the page offering you the appropriate version of Java depending on your operating system.
  - Click Agree and Start Free Download. After the file completed downloading, follow the instructions to complete installation of JRE.

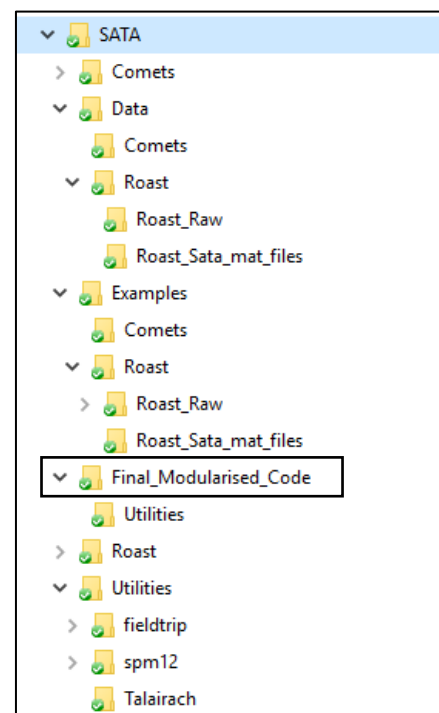
### 2.1.2 Downloading SATA

The full SATA package can be downloaded from here <https://doi.org/10.21979/N9/DMWPZK>  
Or alternatively the main file “Final\_modularised code” can be download from Github <https://github.com/ClinicalBrainLab/SATA>.

However the GUI version needs downloading of the software COMETS, ROAST, SPM, FIELD TRIP, and TALAIRACH CLIENT and placing these software in the format shown. The link to download these software has been mentioned at the end of the manual.

#### ***Important Note:***

- 1) Please make sure that there is no space in your folder naming convention. (Causes errors while running ROAST).
- 2) Ensure that the Talairach.jar file in the directory SATA\Utilities\Talairach is opened using Java.



## 2.2 Setting up SATA

Open the folder that has been downloaded by following the steps under section 2.1.2. The downloaded SATA folder will have two folders named COMETS and ROAST with these software packaged inside it. Alternatively, the user can download these software (as indicated in sections 2.2.1 and 2.2.2) and save them in the respective folders. If the user decides to simulate the montages in either COMETS or ROAST, he or she may choose anyone of them. Here it is important to mention SATA has been developed in such a modularized way that it can work on any one of the software or on both (COMETS/ROAST).

### 2.2.1 Introduction to the software COMETS

COMETS v2.0 is provided with the main folder under SATA\Comets. Alternatively you can download COMETS v2.0 and use it in case the user wants to use COMETS outside the SATA interface. Follow the link to download COMETS:

<http://cone.hanyang.ac.kr/BioEST/Kor/COMETS.html>

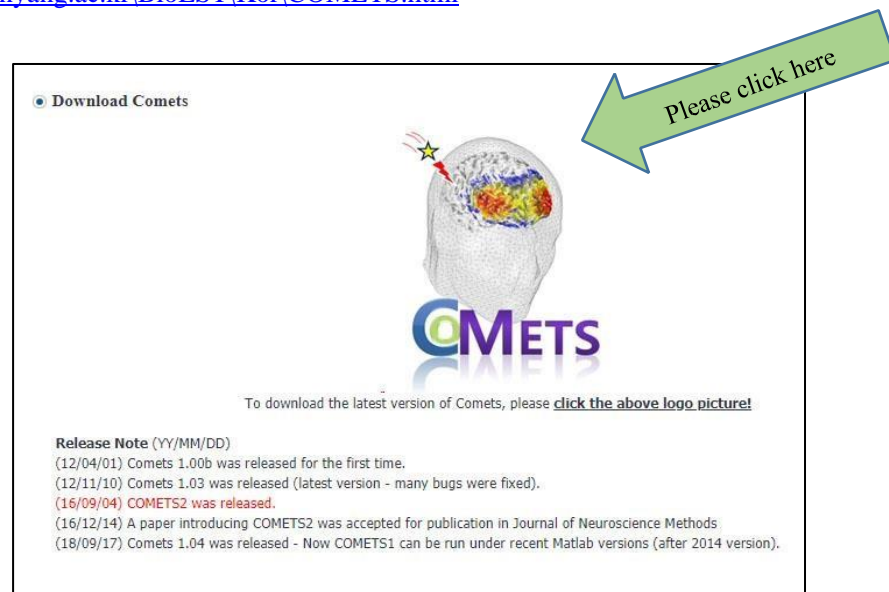


Fig 2: Downloading COMETS

- Click the picture to start the download.

### 2.2.2 Introduction to the software ROAST

If you plan to use ROAST, ROAST that can run in SATA environment has been provided inside the folder (SATA/roast).

To run roast in SATA environment, we have made the following changes:

- Replaced iso2mesh with the latest version available
- Replaced getDp with the latest version available

Alternatively, the original version of ROAST can be downloaded from this link:  
<https://www.parralab.org/roast/>

We have provided option about how to import the data generated by the ROAST outside the SATA environment into SATA for analysis in section 3.5.1 of this manual.

## 2.3 Installing the SATA app

Open MATLAB

Navigate to the SATA\Final Modularized Code Folder. Double click the SATA.mlappinstall file. A new window as show below will open up. Click Install. SATA is now ready to run.



Fig 3: Install the SATA app

To start SATA from the MATLAB's command line##  
`matlab.apputil.run('SATAAPP');`

## Chapter 3: Using SATA

### 3.1 Running SATA

Open MATLAB. Navigate to the apps panel.

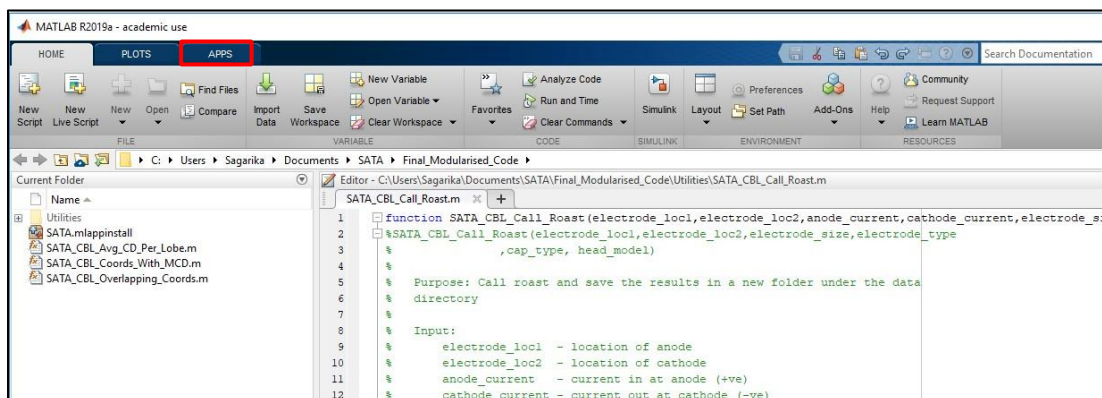


Fig. 4(a): Running SATA

Click the following SATA icon to launch the software.



Fig. 4(b): Running SATA

## The SATA app works well with MATLAB vR2017a, MATLAB vR2019a and above. Alternatively if the user is working with older versions of MATLAB, he/she may use the command lines (section 3.4). In order to use SATA from command line the data must be generated before with COMETS /ROAST.

This opens up the SATA Graphical User Interface (GUI).

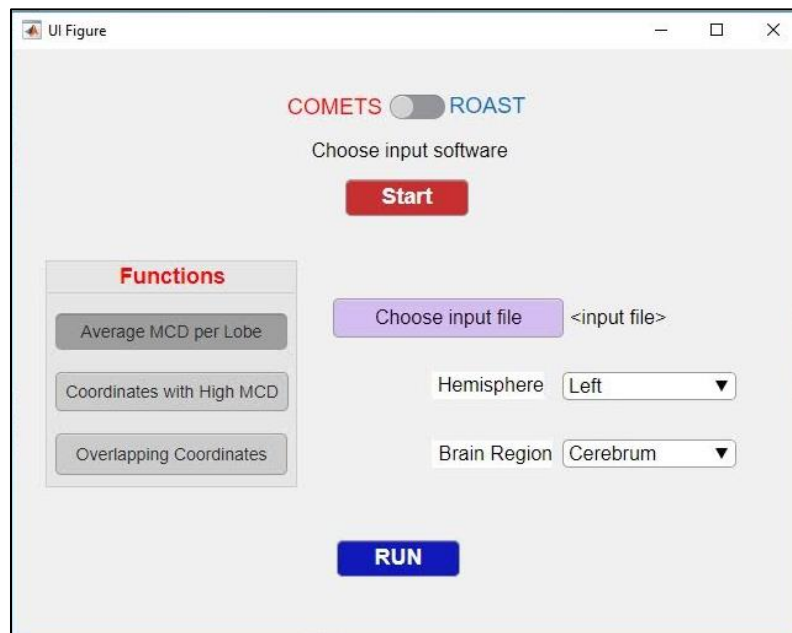


Fig 4 (C): The GUI interface of SATA

### 3.2 How to use SATA

SATA is flexible. A user will need to first simulate the Montages by using either COMETS or ROAST before using the main functions of SATA. Since these software are integrated with SATA GUI, a user can do this from the SATA GUI. Alternatively, they can run COMETS or ROAST independently (Outside SATA GUI) and import the data later.

If the user chooses to use the SATA app, the below interface with a switch to choose either COMETS or ROAST will appear. By default, the switch is set to COMETS.

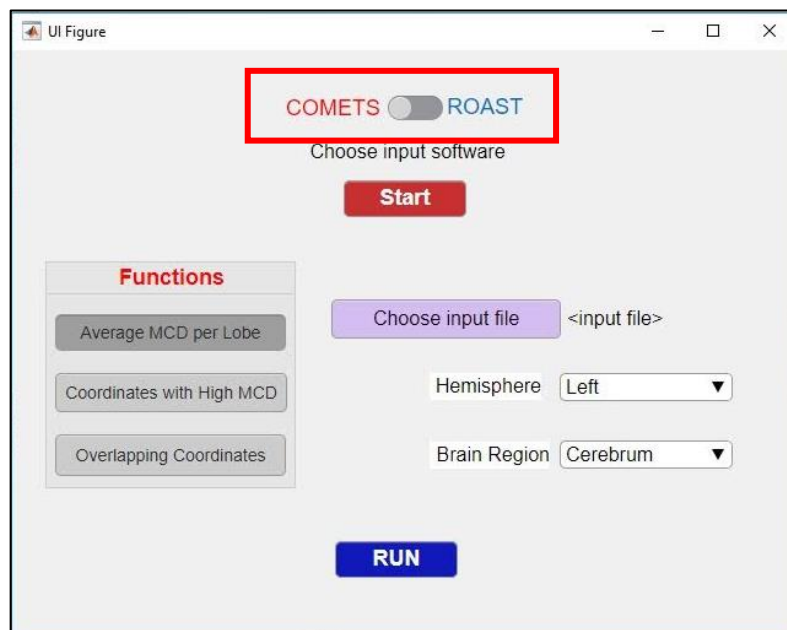


Fig. 5: Choose COMETS or ROAST with the switch

After the user chooses **COMETS** or **ROAST**, then he/she can click the *Start* Button.

For details on COMETS please refer: <http://cone.hanyang.ac.kr/BioEST/Kor/COMETS.html>



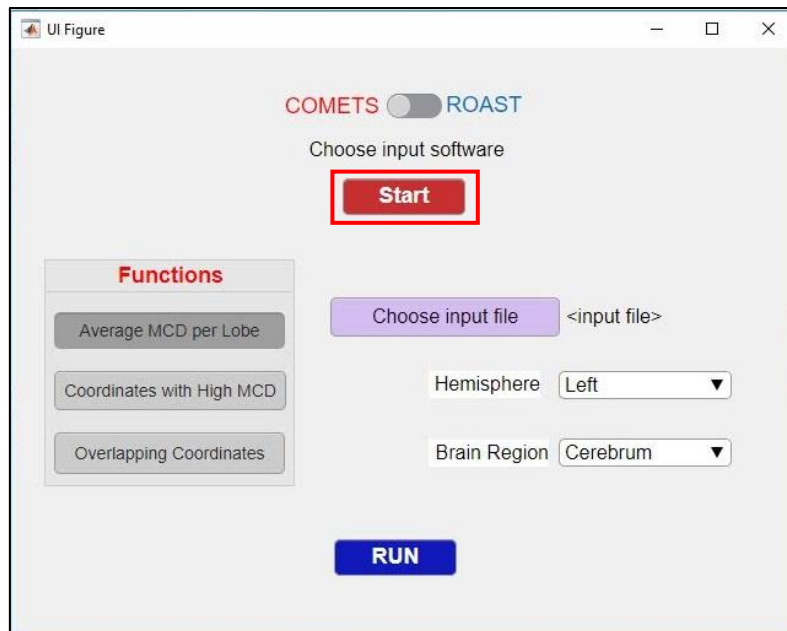


Fig. 6: Click Start button after choosing COMETS or ROAST

### 3.2.1 Generating the data in COMETS and saving the output.

The following manual demonstrates how to use the COMETS to simulate any montage. [http://cone.hanyang.ac.kr/BioEST/Kor/Comets/Comets2\\_manual.pdf](http://cone.hanyang.ac.kr/BioEST/Kor/Comets/Comets2_manual.pdf). Please follow the following steps in order to save the data from COMETS to be used by SATA.

It is very important for the users to know the format for saving the simulation output data. This will help the user to identify the montage. The naming convention adopted for SATA is as follows:

For COMETS: Anode\_Cathode\_(electrode dimension 1)\_(electrode dimension 2)

For ROAST: Refer to **section 3.5.2**

Example: Anode 10:10 EEG position: CP5

Cathode 10:10 EEG position: CZ

Pad dimension (length x breadth): 5 cm x 5 cm

Final Name: CP5\_CZ\_5\_5

For example, please refer to the files saved under SATA\Examples

1. After the simulation is complete as shown in the COMETS manual, uncheck the scalp box ☐ Scalp. This will result into stripping of the scalp from the cortex. Now, place the cursor on any point in the cortex and do a “right click of the mouse” as shown below. This will lead to appearance of a dialogue box with “Export Cursor data to Workspace” on the selected location.



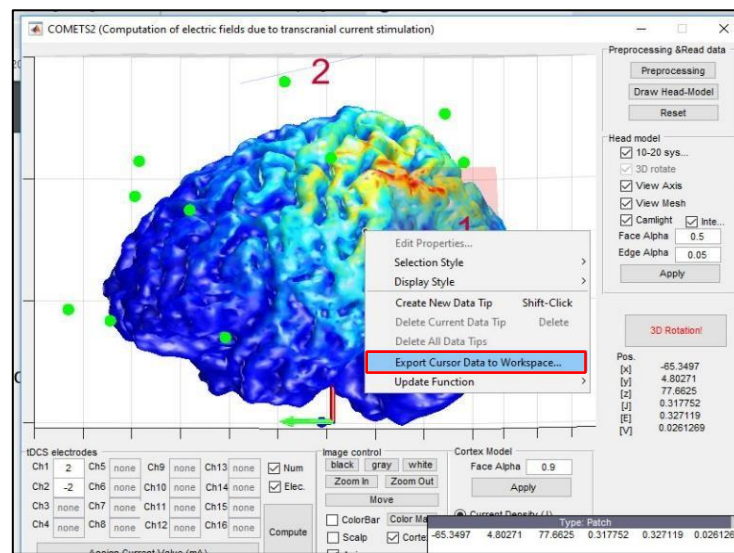


Fig 7: Exporting the data from COMETS to matlab workspace

2. This will be followed by opening of an empty dialogue box with “Enter the variable name”, where save the data following the naming convention described above.

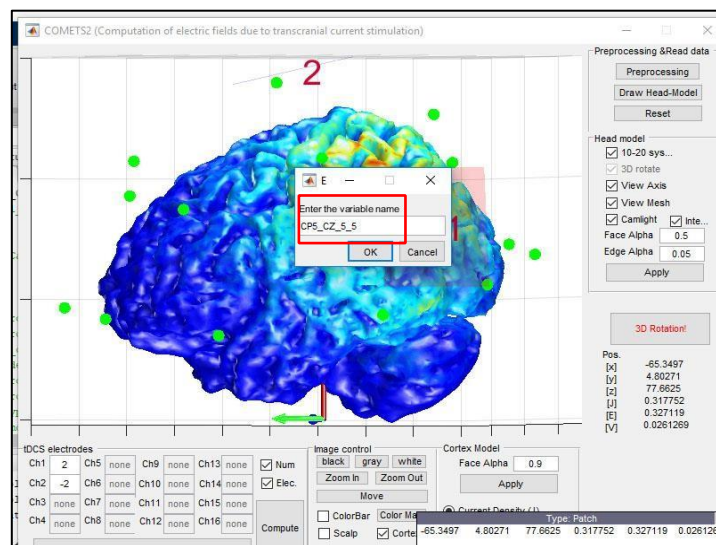


Fig 8: Naming convention of the exported data from COMETS

3. Following this the data file will be saved in the matlab workspace. The data from the matlab workspace needs to be saved under the folder /SATA/Data again following the naming convention described above.

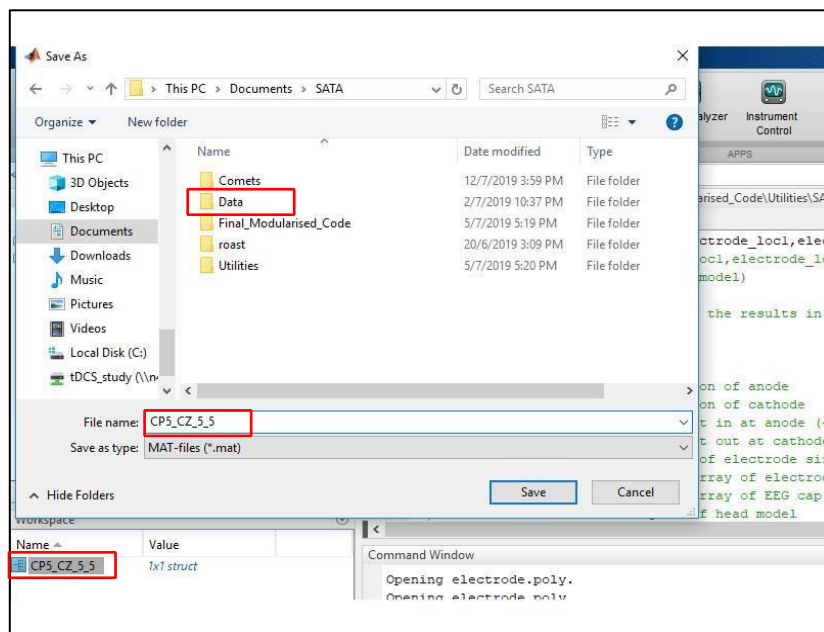


Fig 9: Saving the data from Matlab inside the data folder

**Please Note:** COMETS should not be closed until the output is saved.

### 3.3 Using the Functions in SATA GUI

Currently three functions are available. A user can choose any of the three functions to decide an optimal Montage for tDCS experiments. For details, please refer to (Bhattacharjee et al., 2019)

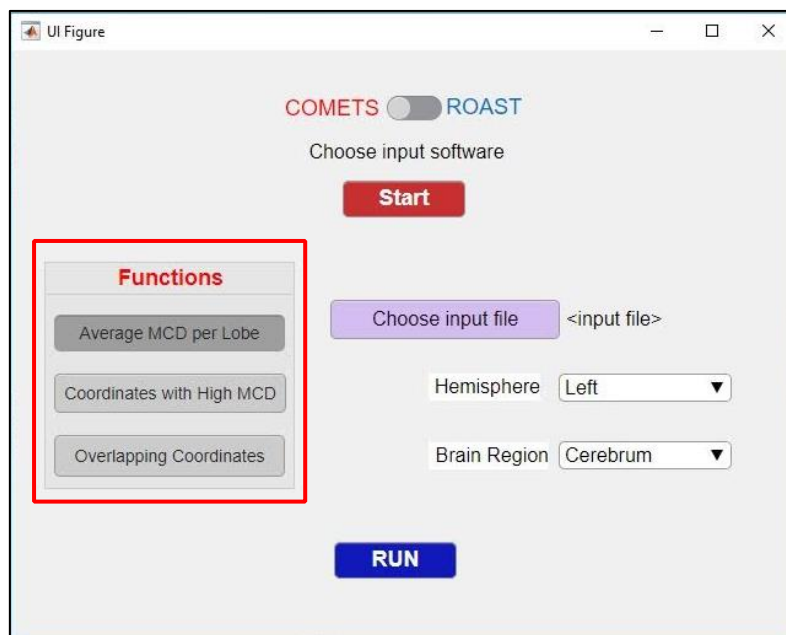


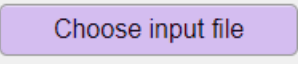
Fig. 10: Functions in SATA GUI

#### 3.3.1 Average MCD per Lobe

The function *Average CD per Lobe* calculates the mean and standard deviation of the Current Density (Mean  $\pm$  Std) values for each of the cortical lobes for a given Montage.

The following steps explain how to run the *Average CD per Lobe* function for a given montage. (We will show it with an example of the montage CP5\_CZ\_5\_5 found in the SATA\examples\Comets directory)

**Step 1:** Click on the  button

**Step 2:** Choose an input file. For this click the  button

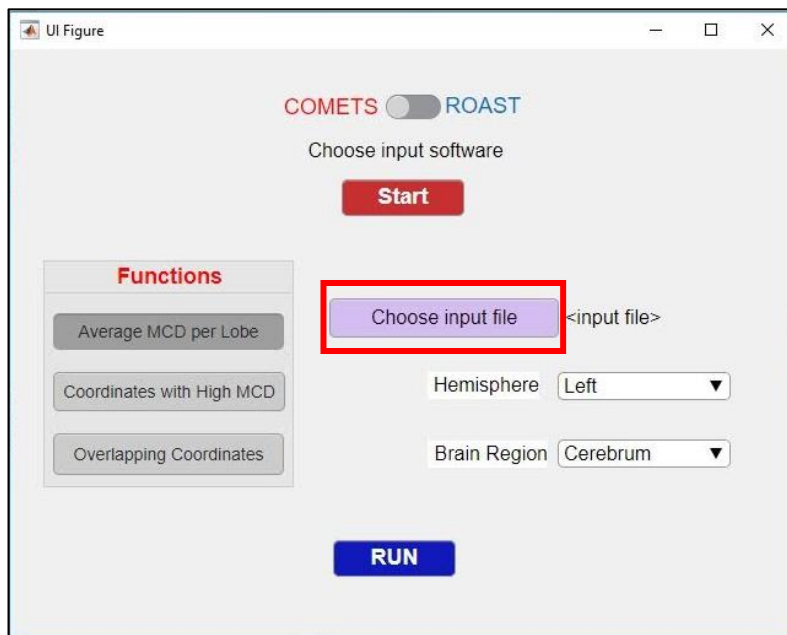


Fig. 11: Choose input file button to select a montage

Navigate to the SATA\examples\Comets directory. Select the CP5\_CZ\_5\_5 file and click open.

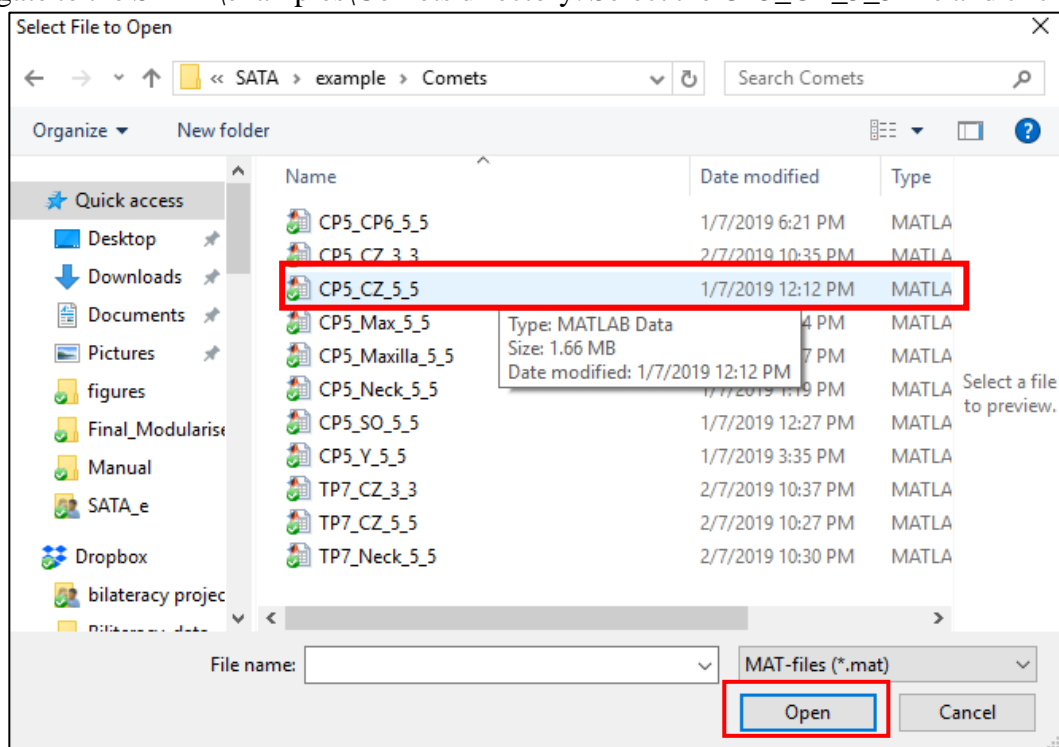


Fig 12: Choosing the required file.

**Step 3:** Choose the hemisphere (Left & Right) and brain region (Cerebrum). The user can choose between left, Right, and Left & Right options for the hemisphere dropdown. Similarly

the user can also choose to select either Cerebrum, Cerebellum or Brainstem from the Brain Region dropdown.

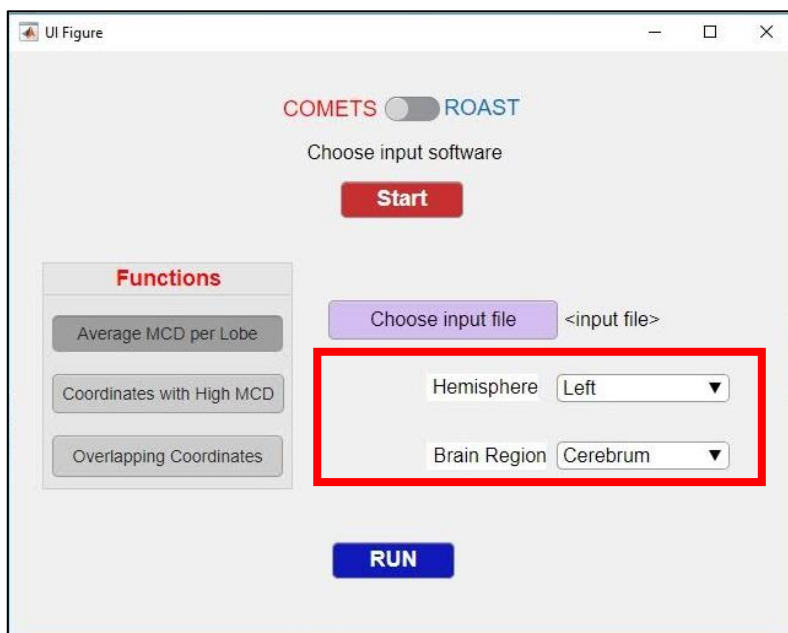


Fig. 13: Choosing Hemisphere and Brain region.

**Step 4:** Click Run

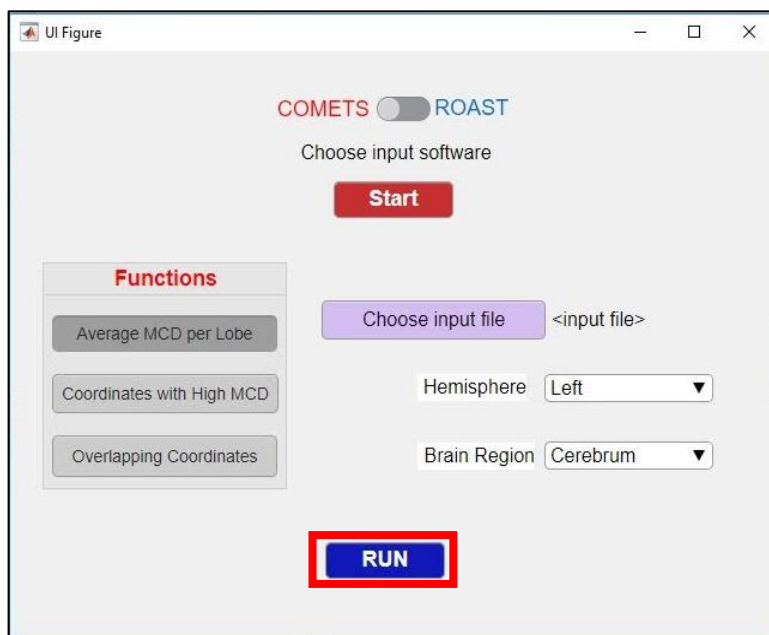


Fig 14: Clicking the Run button

### Step 5: Using Talairach client

After clicking the RUN button, a pop-up will open along with the Talairach Client app. Close this pop-up only after you have finished using the Talairach Client.

For details on Talairach client refer: <http://www.talairach.org/manual.html>

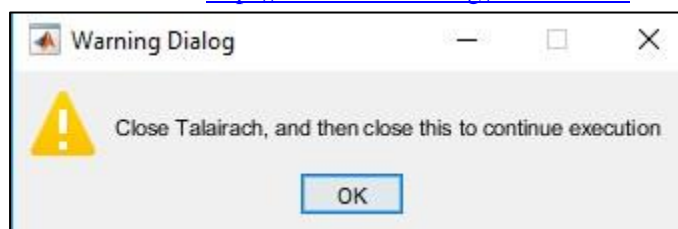


Fig 15: The Talairach pop up that reminds of closing the Talairach client.

In the Talairach Client GUI: Click the *from file* radio (circular) button

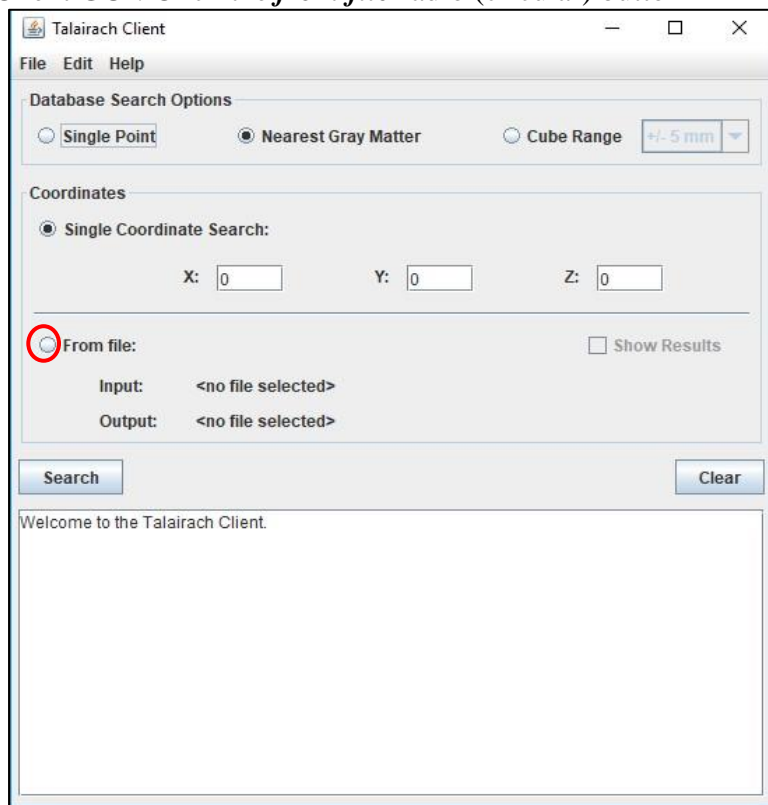


Fig. 16: The GUI of Talairach Client

If users are not in the Talairach directory, they must navigate into the Talairach directory inside the SATA/Utilities folder.

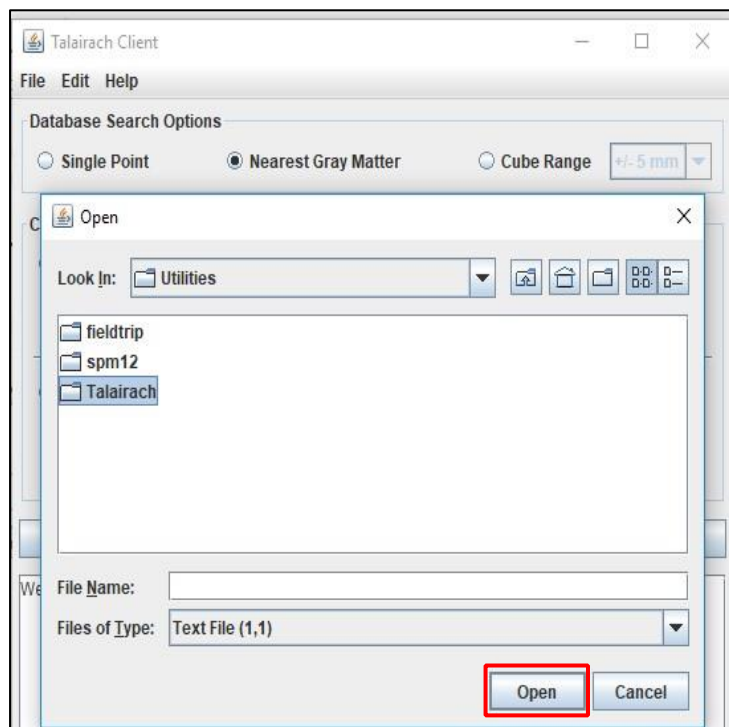


Fig 17: Navigate to the Talairach folder inside the SATA/Utilities folder.

Choose the input file inside the Talairach folder. (**Choose this input file.txt** | open)

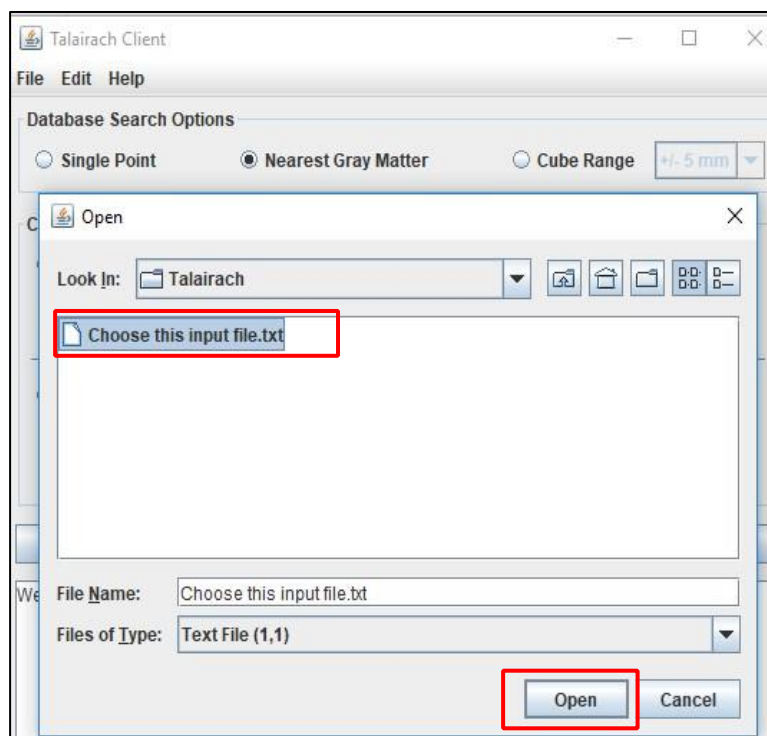


Fig 18: Choosing the input file

Click the *Search* button.

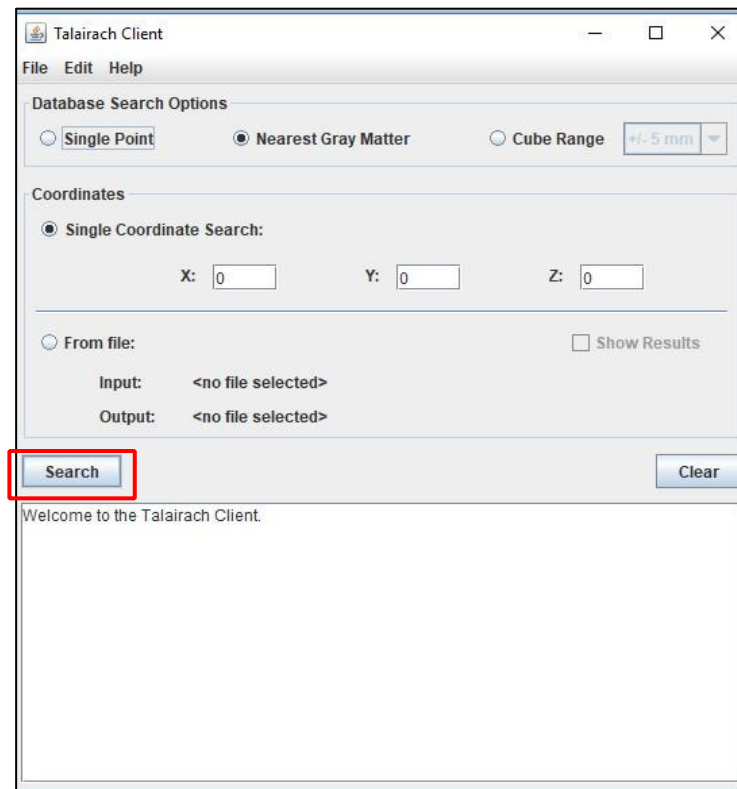


Fig. 19: Clicking the Search

A progress pop-up will appear to show that the Talairach Client is processing the input file. Please wait for Talairach Client to finish processing.

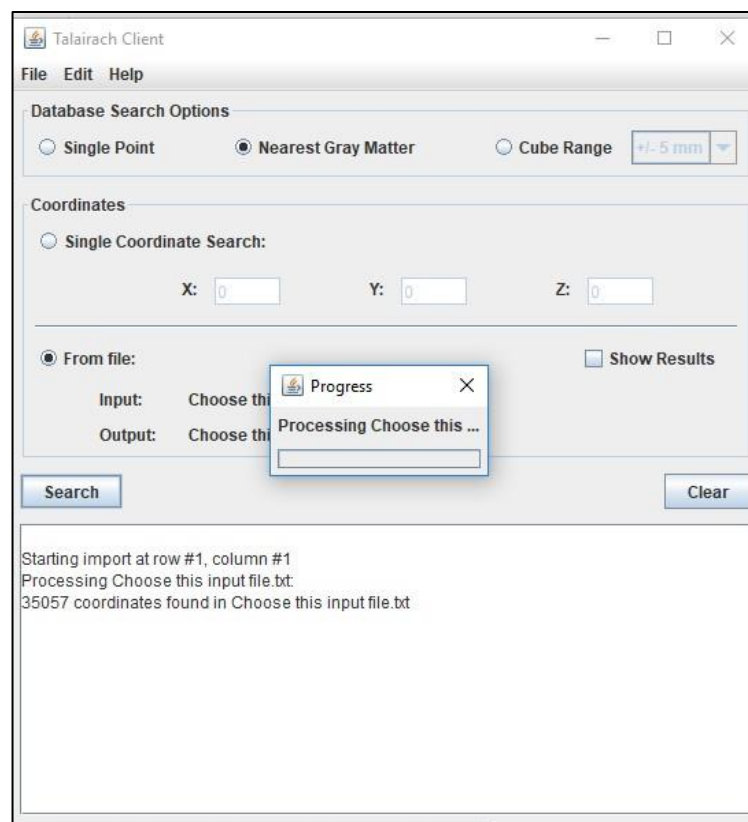


Fig. 20: Talairach progress bar



Once the file has been processed, the *Progress* dialog box will close automatically. Then close the Talairach Client and close the *Warning Dialog* popup.

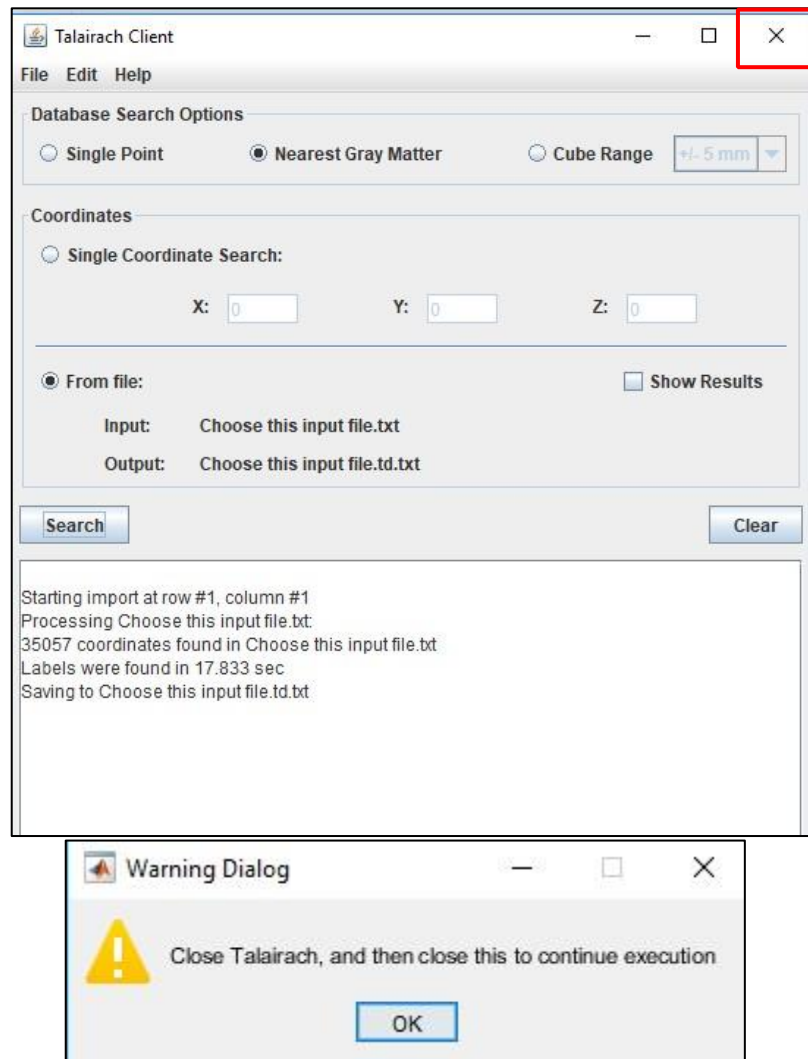


Fig. 21: Closing Talairach Client

**Step 6:** The Results will appear on the screen. The results display the mean  $\pm$  STD values of the different cortical lobes of the brain. It will also show the average of the mean current density across all the cortical lobes (dotted line).

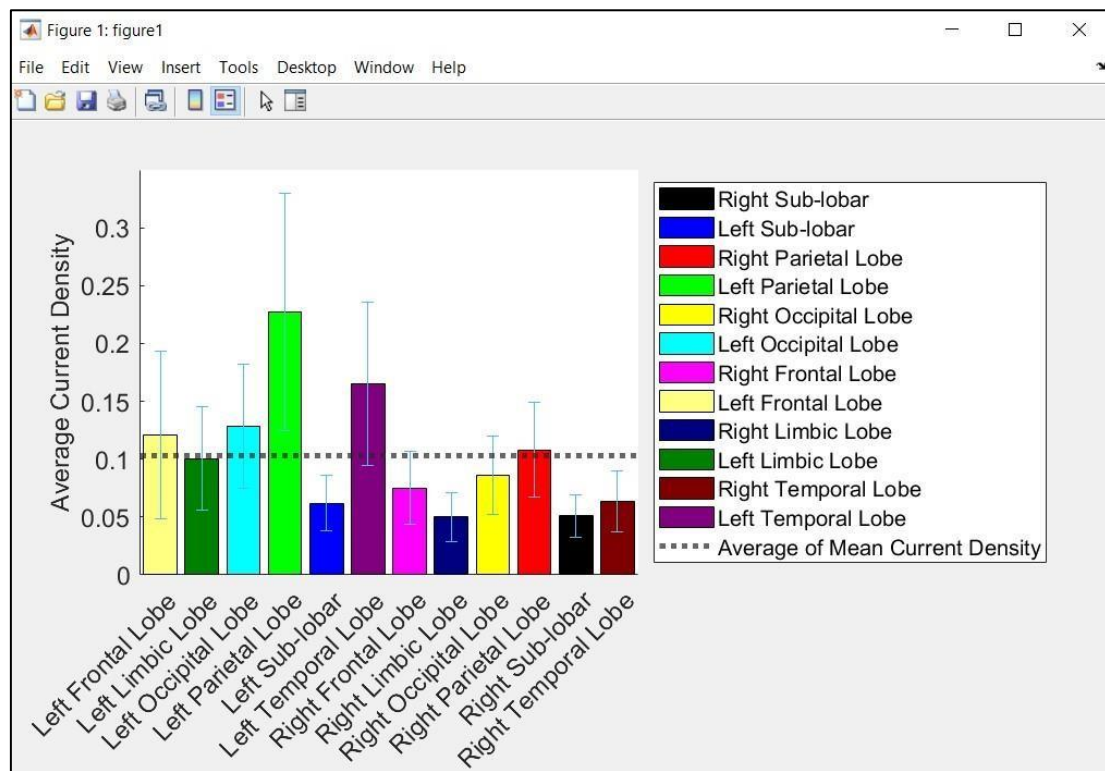


Fig. 22: The output of the Average CD per Lobe Function

### 3.3.2 Coordinates with High MCD

The function *Coordinates with MCD* plots the weighted current densities of the different gyri as a decreasing bar plot and overlays the weighted current densities on the cortical surface map of the brain.

Weighted current density (WCD) is defined as the number of coordinates (N) of a given gyrus multiplied by the mean current density (MCD) of that gyrus.

$$WCD = N \times MCD$$

The following steps explain how to run the *Coordinates with MCD* function for a given montage. We will explain this function with the same example – CP5\_CZ\_5\_5. For reference the user can find this test montage in the SATA\examples\Comets directory.

Coordinates with High MCD

**Step 1:** Click on the button.

After clicking the button the protocol will follow the same steps (steps 2-5) mentioned in **section 3.3.1** under *Average CD per Lobe*. However final output in step 6 will be different as mentioned below.

**Step 6:** The Results will appear on your screen. You will find a bar chart as well as the overlay of the bar chart on the cortical surface map of the brain. The warmer colours represent higher Weighted Current Density values.

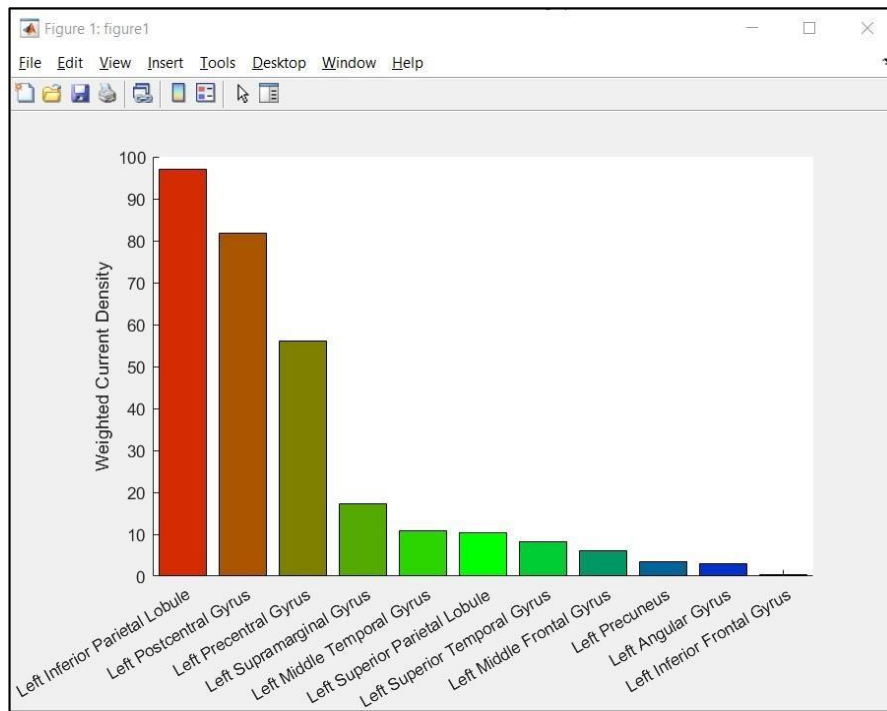


Fig. 23(a): Bar plot of Weighted Current Density for different cortical gyri

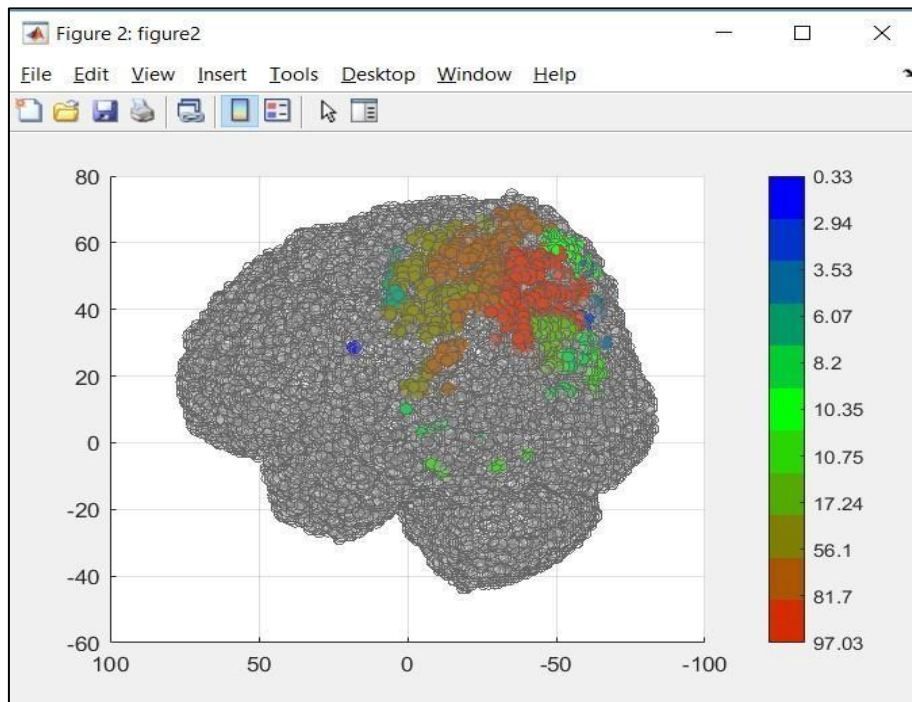


Fig. 23(b): Weighted Current Density bar chart overlaid on the brain surface map

### 3.3.3. Overlapping Coordinates

The function *Overlapping Coordinates* plots the coordinates that are common across two input montages. For example, suppose we have two *Montages A* and *B* stimulating different brain regions and we want to find out which regions are being stimulated by both these Montages. In this case, the function *overlapping coordinates* can be used to find out the common cortical areas stimulated by both Montages.

The following steps explain how to run the *Overlapping Coordinates* function for two given montages. (We will show this with two montages CP5\_CZ\_5\_5 and TP7\_CZ\_5\_5 from the directory SATA\examples\Comets)

Click on the button.

Step 1: Choose the first input file (In this case, we choose CP5\_CZ\_5\_5)

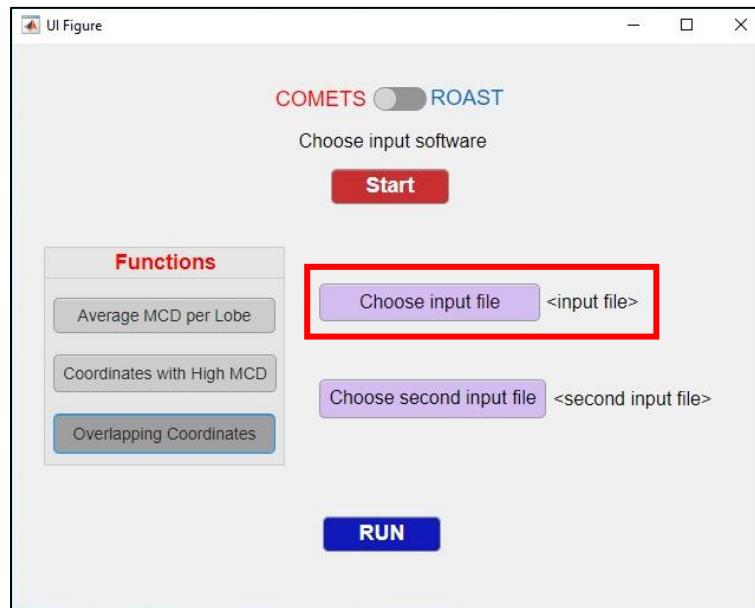


Fig 24: Choose the first input file

Step 2: Choose the second input file (In this case, we choose TP7\_CZ\_5\_5)

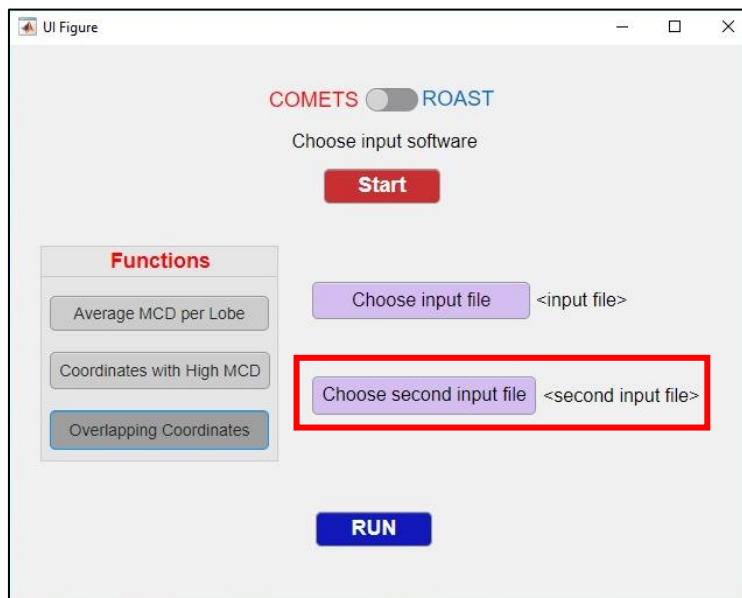


Fig. 25: Choosing the second input file

Step 3: Click on the *Run* button

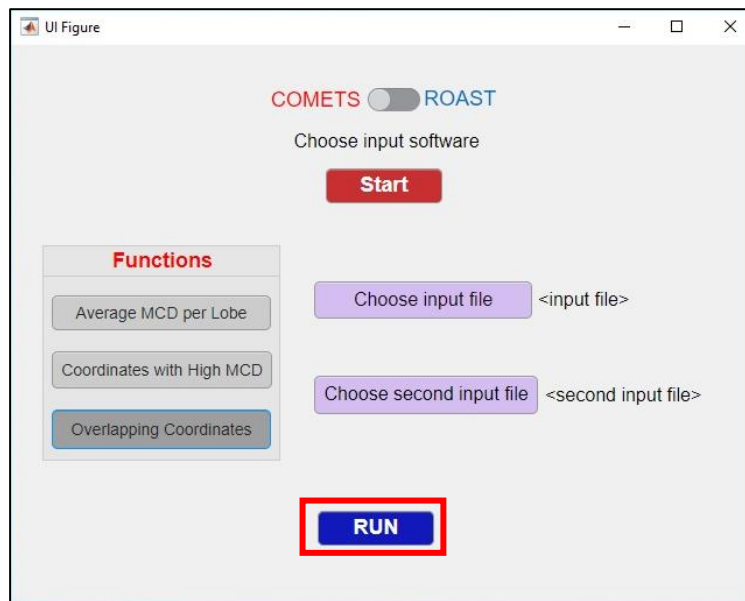


Fig 26: Click the RUN button

Step 4: The Results will appear on the screen. This figure will show the overlay of both montages on the cortical surface of the brain. The first Montage will be represented by yellow dots and the second by cyan dots. The regions that are stimulated by both montages will be represented by red dots.

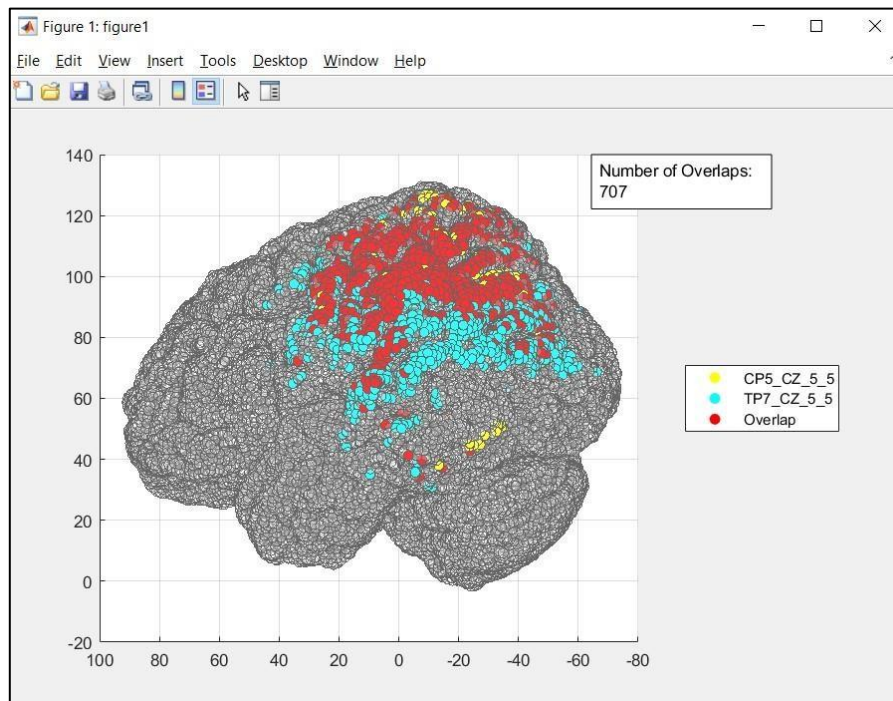


Fig. 27: Output of the Overlapping Coordinates function representing the overlap regions and the number

### 3.4 Using the SATA Functions in MATLAB Command Window

The above GUI based functionality can also be operated using MATLAB's Command Window.

#### IMPORTANT NOTE:

1. To run the commands, the current working directory must be changed to SATA\Final\_Modularised\_Code.
2. Defaults inputs for every function is assumed to be from COMETS

To start SATA from the MATLAB's command line

```
matlab.apputil.run('SATAAPP');
```

Please add SATA to your MATLAB path as shown in the example.

(Replace the value of P with the path of SATA)

Example:

```
P = "C:\Users\Documents\SATA"; % demo genpath (P);
```

To load a particular Montage (\*.mat), use the load function (Commands shown based on the SATA\example folder).

```
load ("SATA\example\Comets\CP5_CZ_5_5.mat");
```

#### ***Average MCD per lobe***

Main inputs: Montage and Brain region

```
SATA_CBL_Avg_MCD_Per_Lobe (CP5_CZ_5_5,"Cerebrum");
```

For Roast: SATA\_CBL\_Avg\_MCD\_Per\_Lobe (CP5\_CZ\_5\_5,"Cerebrum",true);

#### ***Coordinates with high MCD***

Main inputs: Montage, Brain region, and threshold

```
SATA_CBL_Coords_With_High_MCD (CP5_CZ_5_5,"Cerebrum", 0.5);
```

#### ***Overlapping Coordinates***

Main inputs: Montage 1, Montage 2, threshold 1, threshold 2    load

```
("SATA\example\Comets\TP7_CZ_5_5.mat");
```

```
SATA_CBL_Overlapping_Coords (CP5_CZ_5_5, TP7_CZ_5_5, 0.5, 0.5);
```

For Roast: SATA\_CBL\_Overlapping\_Coords (CP5\_CZ\_5\_5, TP7\_CZ\_5\_5, 0.5, 0.5, true);

### 3.5 Using Roast from SATA

This example illustrates how to use ROAST from SATA.

Although while installing ROAST some errors might be encountered, for which the ROAST developers must be referred. Otherwise, we could provide you our version of ROAST on request.

If ROAST is installed successfully inside the empty ROAST folder of SATA, ROAST can be used with the SATA GUI to generate the data. Though at present (while using SATA GUI) we restrict users to choose only two electrodes where the electrode size must be of the same dimensions. For example if anode is a pad electrode of size A x B then the cathode should also be a pad electrode of size A x B.

To run ROAST, set the *input software* switch to **ROAST**. Click the *Start* button.

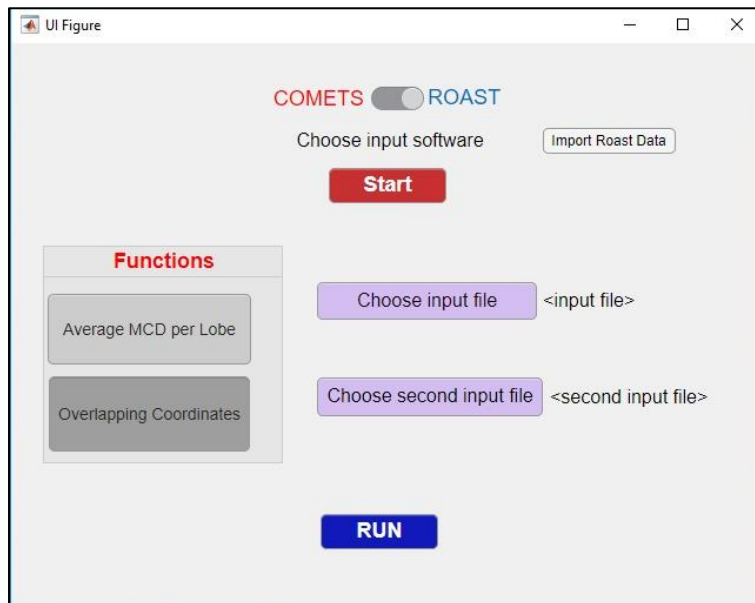


Fig. 28: Click the Start button to use ROAST

A pop up will appear.

If the user wants to use the default settings, he/she may click Yes. Otherwise Roast will work with the user defined settings. (For ease, we will explain with the default settings).

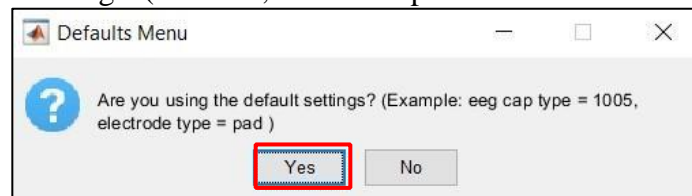


Fig. 29(a): ROAST inputs as pop-ups

Once the user clicks Yes, A new window will open up. In this window the anode position must be chosen. For this example, we choose the anode as CP5.



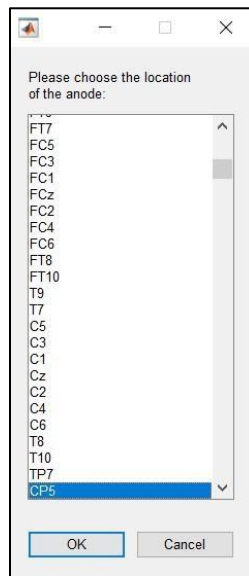


Fig. 29(b): ROAST inputs as pop-ups

In the next window the current flowing in at the anode must be entered. For this example, enter the current value as 2mA. However one can choose any value between 0 and p. p may be any desired upper limit.



Fig. 29(c): ROAST inputs as pop-ups

Similarly, in the next window the cathode position must be chosen. For this example, choose the Cathode as F3.

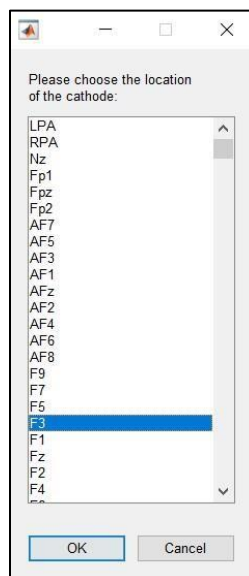


Fig. 29(d): ROAST inputs as pop-ups

In the next window the current flowing in at the anode must be entered. For this example, enter the current value as -2mA.

Kindly note that the current flowing in at the anode must be equal to the current flowing out at the cathode. Hence, it must be a negative value.

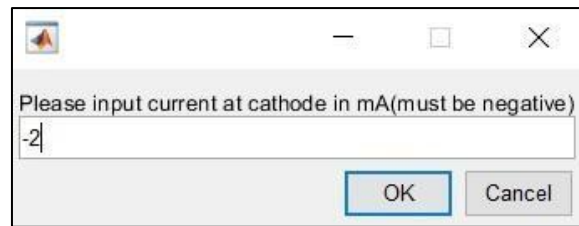


Fig. 29(e): ROAST inputs as pop-ups

In the next window the dimensions of the electrode must be entered.

For this example, input the dimension of the electrodes as 50 50 3. Please note that the dimensions are specified in mm. Also, note that there must be a space between the 3 numbers.

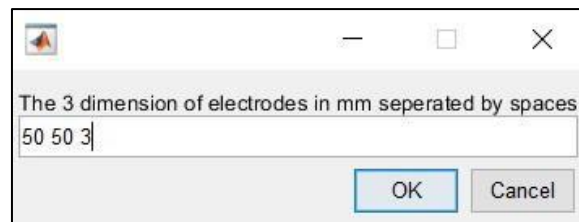


Fig. 29(f): ROAST inputs as pop-ups

\*As mentioned in ROAST the pad electrode must have thickness of at least 3 mm.

\*\*As a special case if a user select disc electrodes, he/she should enter only 2 value, specifying radius and height.

Please read the ROAST manual for details.

Once all the dialog boxes have been closed ROAST will start operating. You may find the below output appearing on your command line.

```
=====
STEP 1 (out of 6): SEGMENT THE MRI...
=====
```

Fig. 30(a): Start of ROAST in MATLAB Command Line

This takes a while (about 10 minutes). The end of the data generation is indicated by:

```
=====
generating 3D renderings...
generating slice views...
=====ALL DONE=====
```

Fig. 30(b): End of ROAST in MATLAB Command Line

### 3.5.1 Choosing ROAST generated data for SATA analysis

The output from ROAST is converted into a \*.mat format to be used for SATA analysis. The SATA usable ROAST output file can be found at the folder:

SATA/Data/ROAST/Roast\_Sata\_mat\_files. Please choose the appropriate input file using the *choose input file* button.

Please check that while running ROAST data, the *Choose input Software* switch must be set to ROAST.

### 3.5.2 Importing pre generated ROAST data into SATA

Some users may prefer to use ROAST outside of SATA GUI and use the ROAST generated output for SATA analysis. For this the data must be imported to SATA as follows.

First, save the ROAST output files for a given Montage in a folder. Name the folder in the format that is adopted in SATA as follows:

Anode\_Cathode\_(electrode dimension 1)\_ (electrode dimension 2)

Eg: Anode 10:10 EEG position: CP5

Cathode 10:10 EEG position: CZ

Pad dimension (length x breadth x width): 50 mm x 50 mm

Final Name: CP5\_CZ\_50\_50

After the user has saved the ROAST data in a folder named in the SATA format, he or she can import the ROAST data as shown below:

Click the *import Roast Data* button.

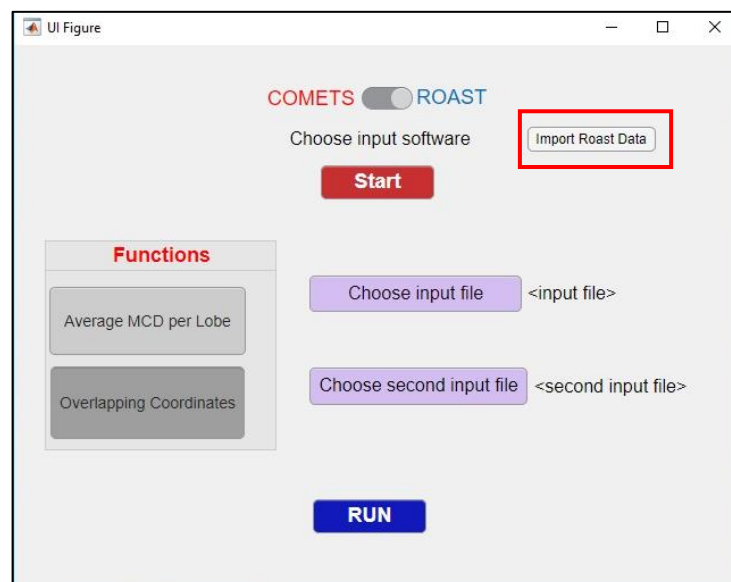
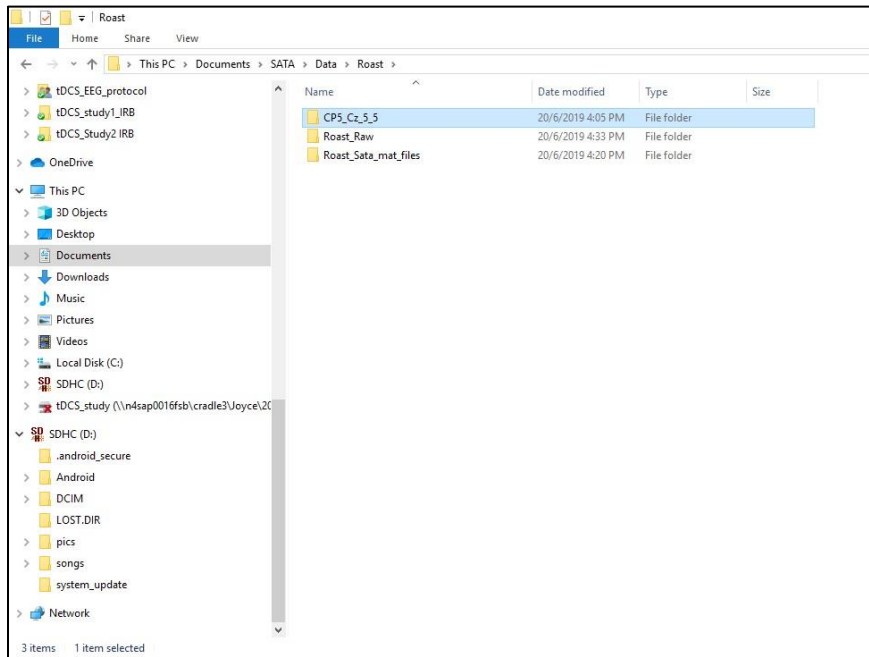


Fig 31: Import the Roast data



Navigate to the folder where the outputs from ROAST are saved when ROAST is not used from the SATA GUI.

Fig. 32: Choosing desired folder to be imported

This creates a copy of the ROAST folder in the SATA\Data\Roast\ Roast\_Raw directory and a \*.mat file to be used by SATA in the SATA\Data\Roast\Roast\_Sata\_mat\_files.

# Acknowledgement

We thank the following colleagues for their kind helps during SATA developing and manual revising

1. **Dr. Rajan Kashyap**, Centre for Research and Development in Learning (CRADLE), Nanyang Technological University, Singapore
2. **Prof. Brenda Rapp**, The Johns Hopkins University, Krieger School of Arts and Sciences, Baltimore, United States,
3. **Prof. Kenichi Oishi**, Department of Neurology, The Johns Hopkins University, School of Medicine, Baltimore, United States,
4. **Prof. John E. Desmond**, The Johns Hopkins University, School of Medicine, Baltimore, United States
5. **Prof. SH Annabel Chen**, Department of psychology, School of Social Sciences, Nanyang Technological University, Singapore, Centre for Research and Development in Learning (CRADLE), Nanyang Technological University, Singapore, Lee Kong Chian School of Medicine (LKC Medicine), Nanyang Technological University, Singapore

We also thank the developers of the following software and toolboxes whose source codes or file formats were referenced during our package developing:

1. Matlab: <https://www.mathworks.com/products/matlab.html>
2. SPM: [www.fil.ion.ucl.ac.uk/spm/](http://www.fil.ion.ucl.ac.uk/spm/)
3. Feildtrip <http://www.fieldtriptoolbox.org/>
4. Talairach client <http://www.talairach.org/client.html>
5. Comets2 <http://cone.hanyang.ac.kr/BioEST/Kor/Comets.html>
6. ROAST: <https://www.parralab.org/roast/>

This work was supported by JHU-NTU grant and HASS Incentive grant, NTU, Singapore.

# References

- Bhattacharjee, S., Kashyap, R., Rapp, B., Oishi, K., Desmond, J., & Chen, S. (2019). *Simulation Analyses of tDCS Montages for the Investigation of Dorsal and Ventral Pathways*. Scientific Reports (in press)
- Huang, Y., Datta, A., Bikson, M., & Parra, L. C. (2019). Realistic vOlumetric-Approach to Simulate Transcranial Electric Stimulation – ROAST – a fully automated open-source pipeline. *BioRxiv*.  
<https://doi.org/10.1101/217331>
- Lancaster, J. L., Rainey, L. H., Summerlin, J. L., Freitas, C. S., Fox, P. T., Evans, A. C., ... Mazziotta, J. C. (1997). Automated labeling of the human brain: A preliminary report on the development and evaluation of a forward-transform method. *Human Brain Mapping*, 5(4), 238–242.
- Lancaster, J. L., Woldorff, M. G., Parsons, L. M., Liotti, M., Freitas, C. S., Rainey, L., Fox, P. T. (2000). Automated Talairach atlas labels for functional brain mapping. *Human Brain Mapping*, 10(3), 120–131.
- Lee, C., Jung, Y.-J., Lee, S. J., & Im, C.-H. (2017). COMETS2: An advanced MATLAB toolbox for the numerical analysis of electric fields generated by transcranial direct current stimulation. *Journal of Neuroscience Methods*, 277, 56–62.
- Oostenveld, R., Fries, P., Maris, E., & Schoffelen, J.-M. (2011). FieldTrip: Open Source Software for Advanced Analysis of MEG, EEG, and Invasive Electrophysiological Data. *Computational Intelligence & Neuroscience*, 1–9. <https://doi.org/10.1155/2011/156869>
- SPM - Statistical Parametric Mapping. (n.d.). Retrieved July 5, 2019, from  
<https://www.fil.ion.ucl.ac.uk/spm/>