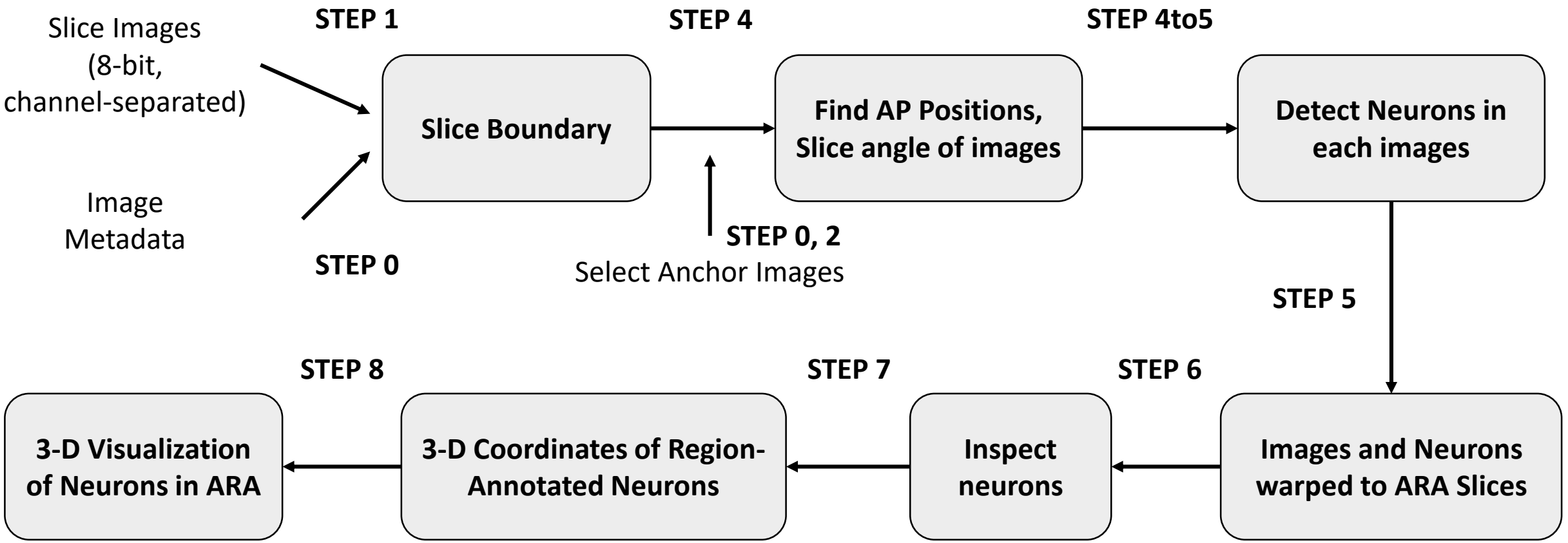
AMaSiNe User Manual

Jun Ho Song, Woochul Choi
Correspondence: Prof. Se-Bum Paik
sbpaik@kaist.ac.kr

Manual version: 20200429

Algorithm pipeline of AMaSiNe

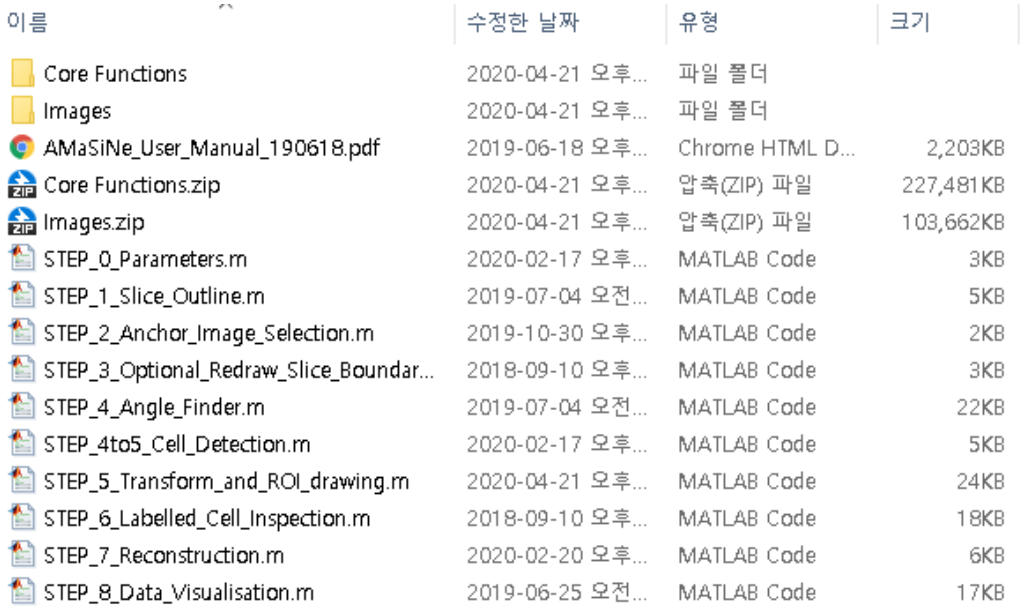


Preparation

0. MATLAB (version > 2017a) is required

1. MATLAB packages are required
   - Computer Vision Toolbox
   - Image Processing Toolbox
   - Parallel Computing Toolbox

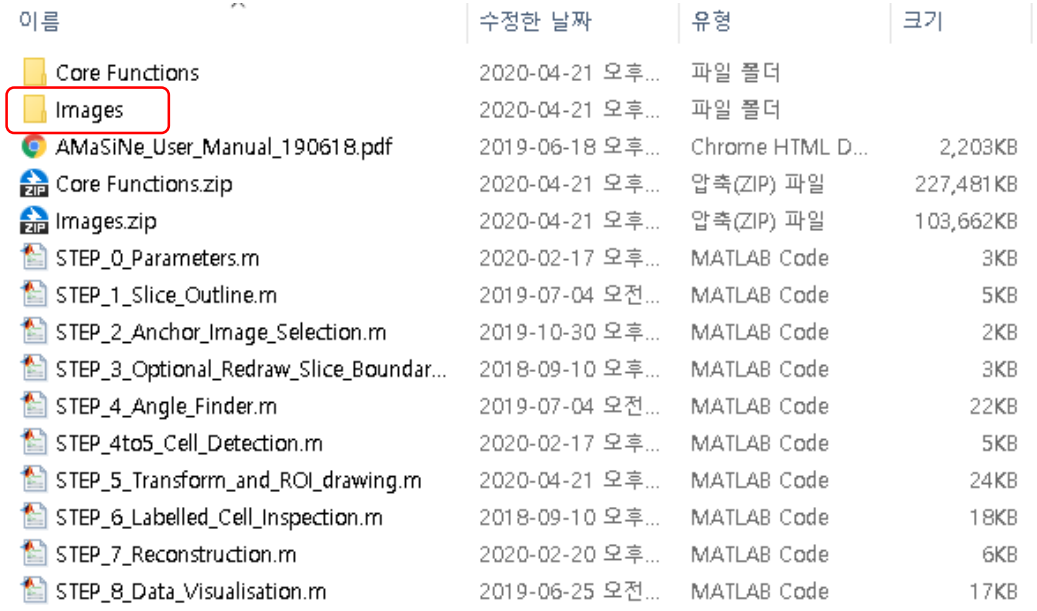
2. Make a new empty folder and unzip the “Core Functions.zip” and “Images” in the same folder



이름	수정한 날짜	유형	크기
Core Functions	2020-04-21 오후...	파일 폴더	
Images	2020-04-21 오후...	파일 폴더	
AMaSiNe_User_Manual_190618.pdf	2019-06-18 오후...	Chrome HTML D...	2,203KB
Core Functions.zip	2020-04-21 오후...	압축(ZIP) 파일	227,481KB
Images.zip	2020-04-21 오후...	압축(ZIP) 파일	103,662KB
STEP_0_Parameters.m	2020-02-17 오후...	MATLAB Code	3KB
STEP_1_Slice_Outline.m	2019-07-04 오전...	MATLAB Code	5KB
STEP_2_Anchor_Image_Selection.m	2019-10-30 오후...	MATLAB Code	2KB
STEP_3_Optional_Redraw_Slice_Boundar...	2018-09-10 오후...	MATLAB Code	3KB
STEP_4_Angle_Finder.m	2019-07-04 오전...	MATLAB Code	22KB
STEP_4to5_Cell_Detection.m	2020-02-17 오후...	MATLAB Code	5KB
STEP_5_Transform_and_ROI_drawing.m	2020-04-21 오후...	MATLAB Code	24KB
STEP_6_Labelled_Cell_Inspection.m	2018-09-10 오후...	MATLAB Code	18KB
STEP_7_Reconstruction.m	2020-02-20 오후...	MATLAB Code	6KB
STEP_8_Data_Visualisation.m	2019-06-25 오전...	MATLAB Code	17KB

Preparation

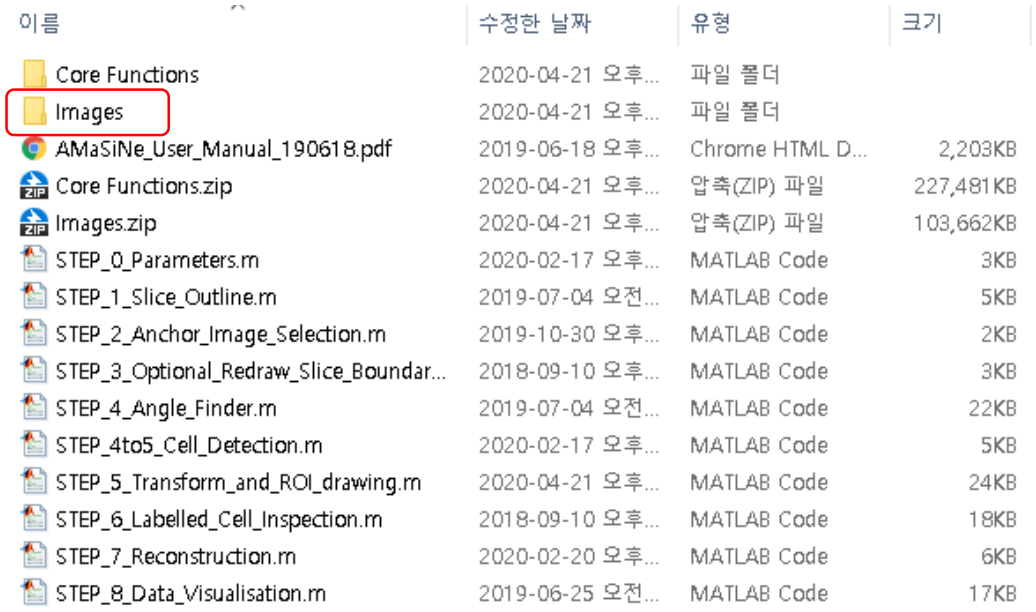
3. Images folder should contain images to analyze

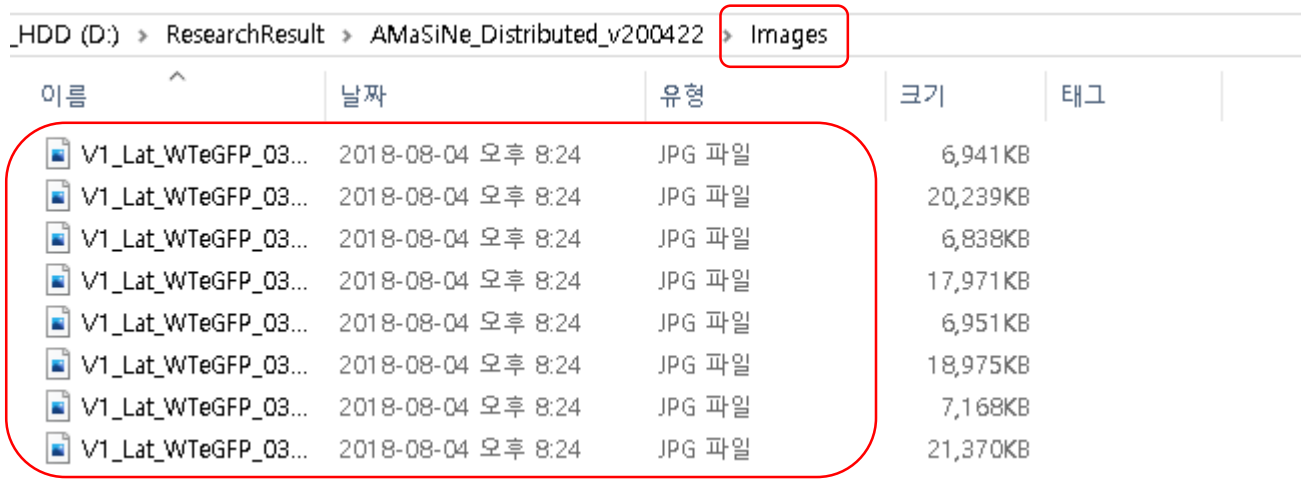


Preparation

4. In Images folder, the images from a “single” brain should be copied

Make sure that other images are not included in the folder (including mother folder)





5. Open MATLAB and change your directory to mother folder

STEP_0_Parameters

- There are 17 parameters to set before moving on;

  Set once, you will need to change only 3~5 parameters
  to analyze image sets from different mouse brains
  (but from same imaging conditions)

- Double Click and open the script “STEP_0_Parameters.m”

STEP_0_Parameters

1. Image directory

Set "**main_folder_dir**" as the folder directory that contains all the functions and images (mother folder)

e.g. main_folder_dir = 'C:\Users\WChoi\Documents\MATLAB\AMaSiNe_Tutorial'

```
%% 1. Image Directory
main_folder_dir='C:₩Users₩WChoi₩Documents₩MATLAB₩AMaSiNe_Tutorial';
```

2. Image names and slice orders

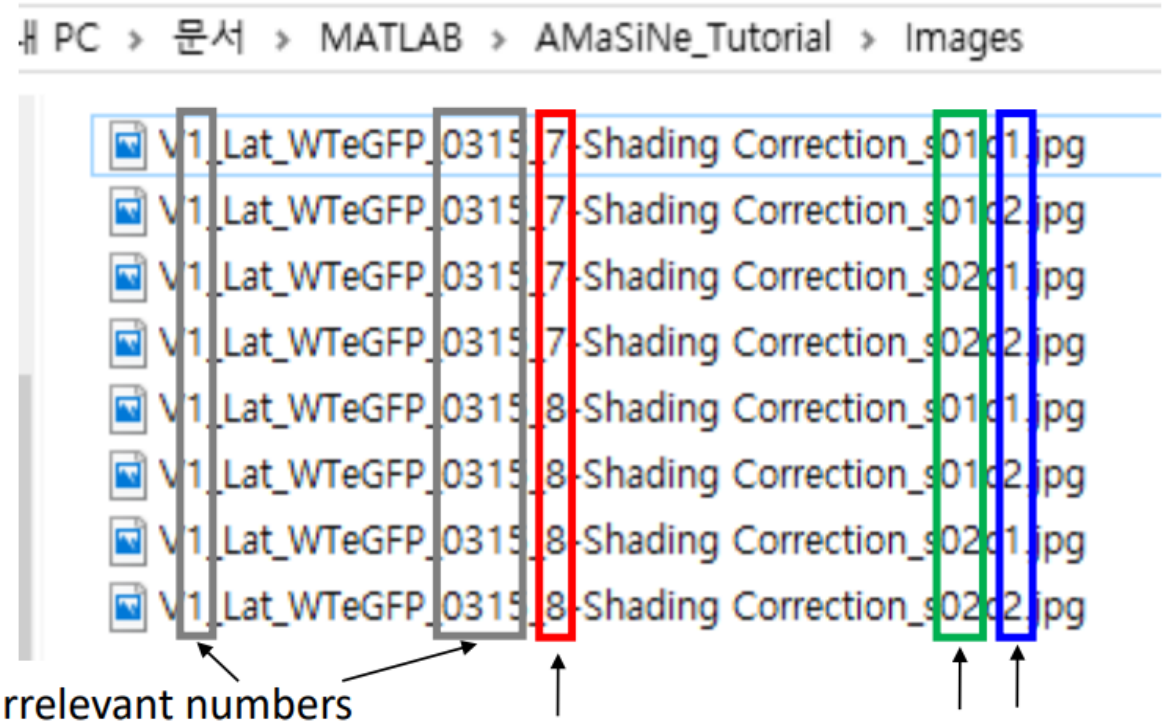
img_format : set it as either 'tif' or 'jpg', depending on your image format
Slice_AP_orPA : The direction in which you sliced your mouse brain
If you sliced anterior → posterior, the value is 1
If you sliced posterior → anterior, the value should be -1

```
%% 2. Image names and slice order
img_format ='jpg';  % if tif = 'tif' ; jpg='jpg'
Slice_AP_orPA= 1;              % If the brain is sliced from anterior to posterior, set this value = 1
                               %                          posterior to anterior, set this value = -1
```

STEP_0_Parameters

2. (Continue) Image names and slice orders

slide_digit, scene_digit, channel_digit : determined by your 'filename' format



Let's say you obtained 4 slices of brain
s01 and s02 from 7th slide glass
s01 and s02 from 8th slide glass
and imaged with 2 channels (c1 and c2, i.e. eGFP and DAPI)

For example, you saved your image file with filename

V1_Lat_WTeGFP_0315_7-Shading Correction_s01c1

1st
2nd
3rd : Slide digit
4th : Scene digit
5th : channel digit

Then, you may define variable as

```
slide_digit = 3;
scene_digit = 4;
channel_digit = 5;
```

```
slide_digit=3;
scene_digit=4;
channel_digit= 5;
```

STEP_0_Parameters

3. Anchor image IDs for angle finding

Leave it empty for now, STEP_2 will determine this

4. Image Parameters

These variables should be changed by your image file information

xy_pix = Image pixel size (um/pixel)

Name_Channels = Image channel names

e.g. if channel 1 = eGFP and channel 2 = DAPI

→ *Name_Channels = {'eGFP', 'DAPI};*

Color_Channel_Structure = index of the channel for finding the slicing angles and slice AP pos.

e.g. if you want to find slicing angles and AP position from 'DAPI' stained image in case above,

→ *Color_Channel_Structure = 1;*

Structure_stain = Staining method of the images to be used for angle searching

Choose one of the three : 'DAPI' or 'AutoF' or 'Nissl'

Color_Channel_Interest = Color channel index in which labelled cells are imaged

e.g. if you want to count the neurons in eGFP channel → *Color_Channel_Interest = 2;*

STEP_0_Parameters

5. Detection Parameters

This variables will be used for automatic detection of neurons in the image

soma_radius = rough range of "radius" of labelled soma (um); [min max]

xy_pix_resc_factor = Image downscaling factor, Default 1;
Low value speed up the image transformation by downscaling the original image
but you may risk deteriorating the image quality

cell_det_thresh = Minimum intensity difference between a cell and background for a cell to be detected
Should be larger than 0; default is 0.25 ~ 1.5
Lower value have a better change of detecting dim neurons
but you may detect noise as a cell

cf) you can modify the code in "*SomaDetection0827.m*" to customize/implement algorithm for cell detection

6. Allen Atlas Info

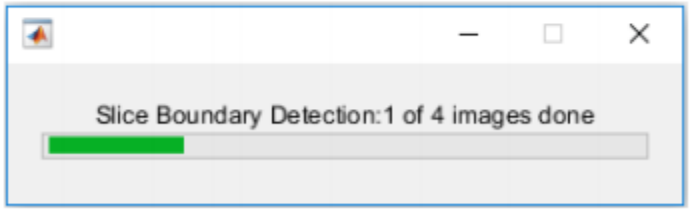
Do not change these parameters unless you use alternative atlas for standard space

STEP_1_Slice_Outline

- This step is to remove image regions other than brain tissues, thus saving computational time and memory

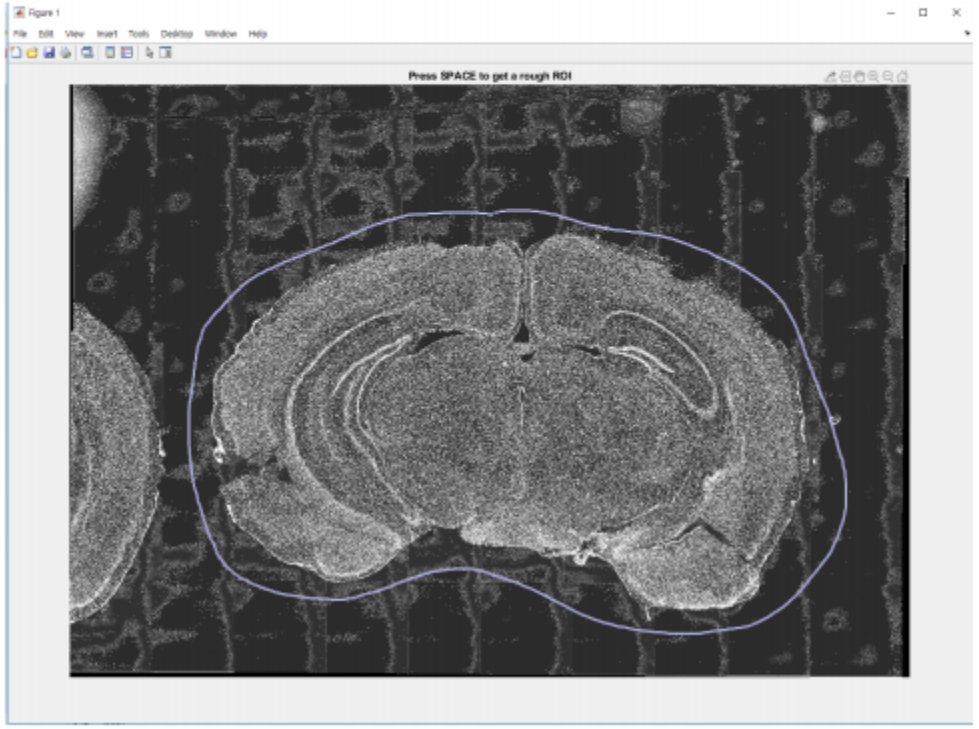
- In the command window, type “STEP_1_Slice_Outline” and press Enter;
(or you can Run the file in the editor window)

- If you run the script, a status bar showing the progress of automatic background removal will pop up



STEP_1_Slice_Outline

- In some cases, automatic slice boundary detection may fail to give you the correct output
  → For these images, you can manually draw “rough” boundaries by clicking-and-dragging
  → The algorithm will search the correct slice boundaries within the rough boundaries you drew

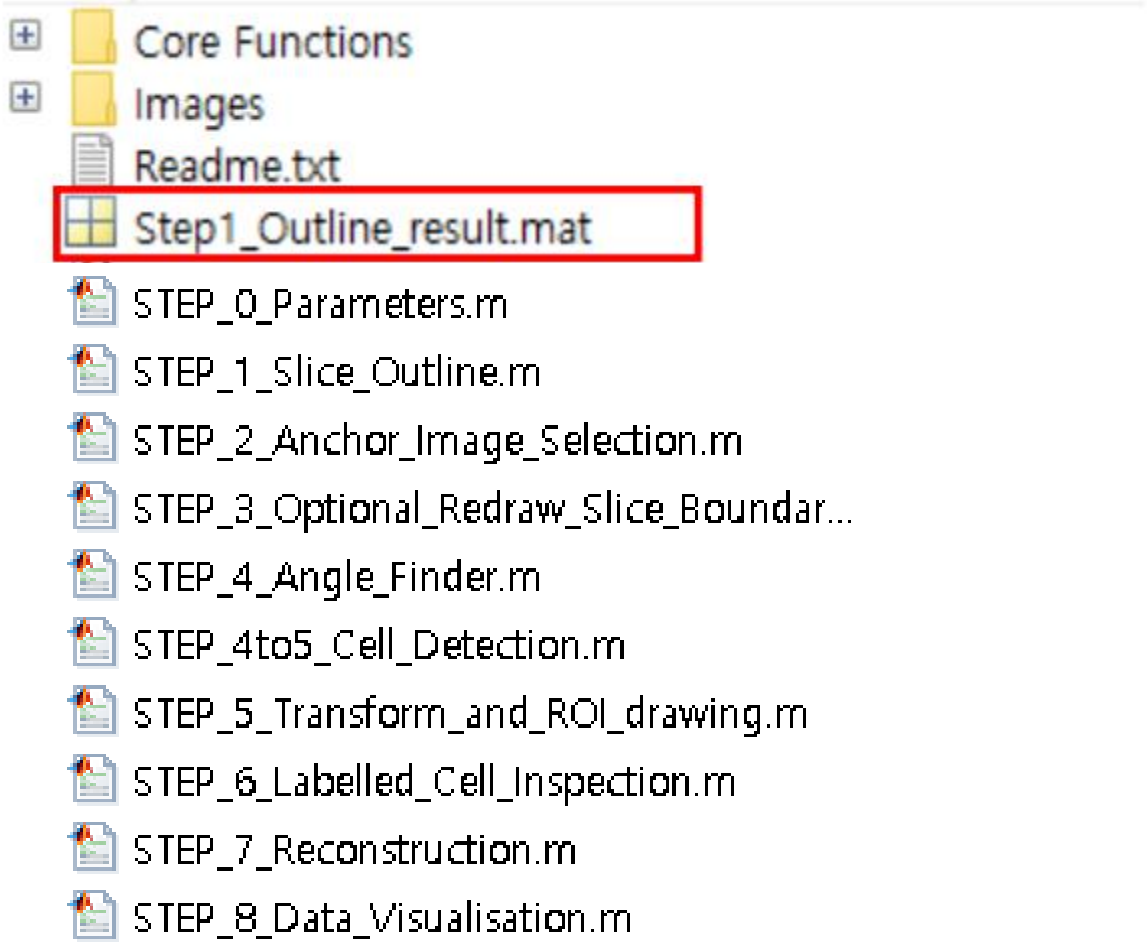


- Click the figure window, and press the space bar and manually draw a rough boundary around slices with mouse

- Images appeared on figure windows are edge-boosted

STEP_1_Slice_Outline

- If you successfully finished this step, a file “Step1_Outline_result.mat” will be saved
  This file contains slice boundary information of each brain slice

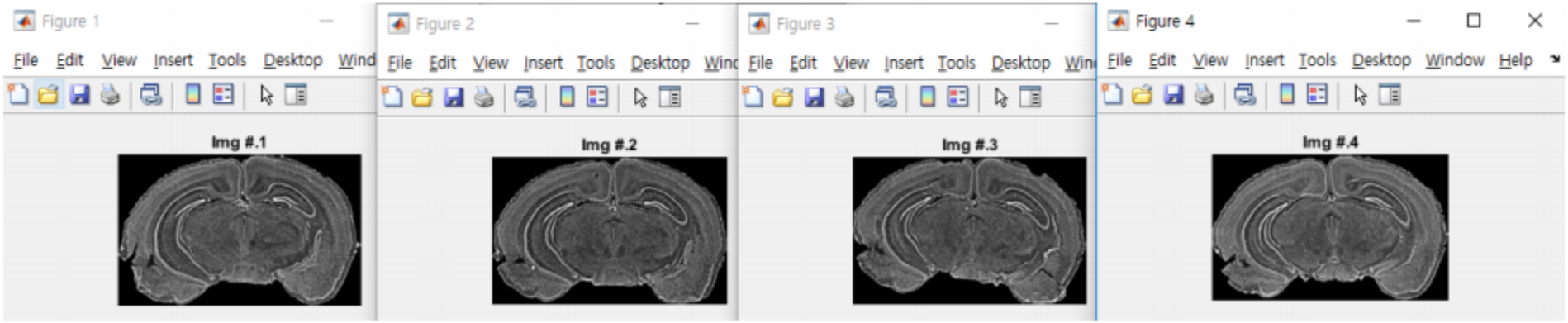


STEP_2_Anchor_Image_Selection

- In this step, you will
  1. Sample “anchor images” to find the slicing angles of the brain and AP positions of individual slices
     - AP positions of un-sampled non-anchor images are interpolated between the anchors

  2. Inspect the automatic slice boundary detection results performed in Step_1
     and choose images the boundaries of which must be redrawn

- In the command window, type “STEP_2_Anchor_Image_Selection” and press Enter (or run the script in editor);
  Image figures with boundaries will pop up

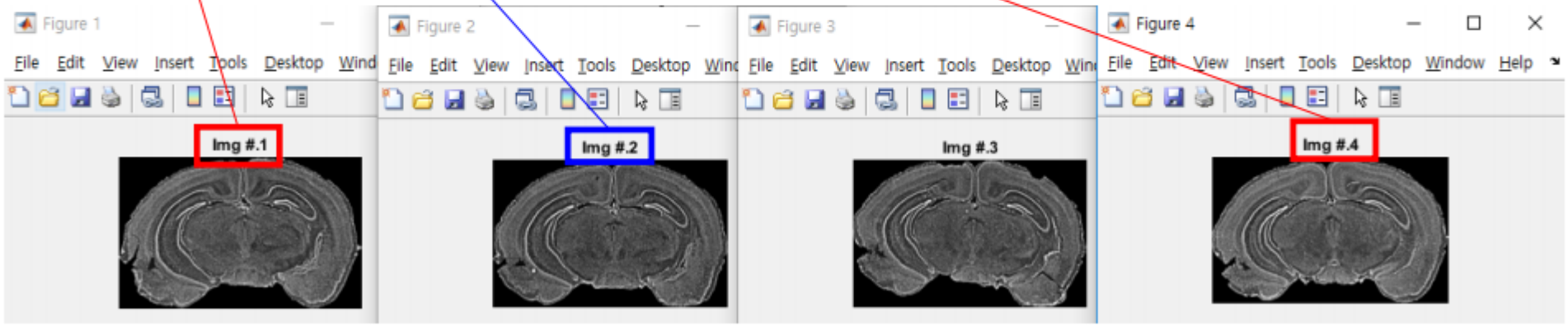


STEP_2_Anchor_Image_Selection

- Once all image figures have appeared, double-click on the “STEP_0_Parameters.m” and open the script

```
%% 3. Anchor Image IDs
anc_img_IDs= sort([1 4]);
img_IDs_reBoundary=[2];
threshold_scale = 1.5;
```

- Type the indices of images samples as “**anchors**”
- (Optional) If boundary is incorrectly drawn in Image #2,
  Type the indices of images to **re-draw the slice boundary**
  If all images are find, leave it as empty ([])

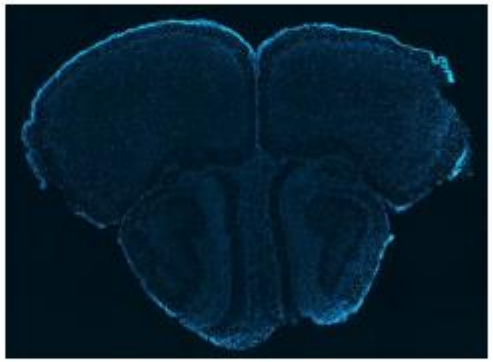


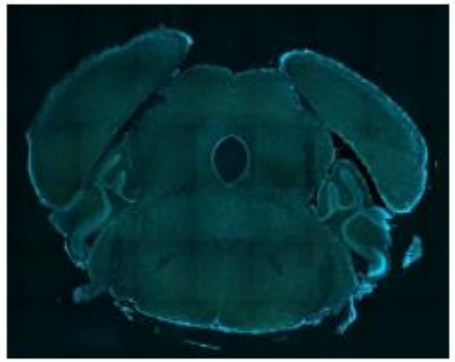
STEP_2_Anchor_Image_Selection

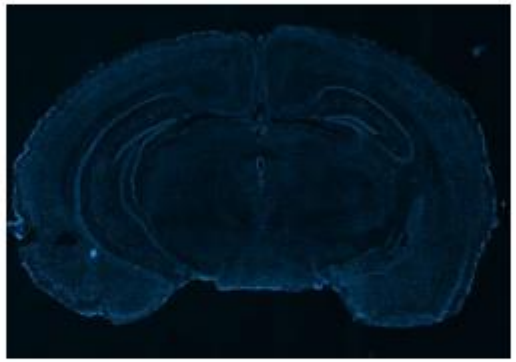
IMPORTANT NOTICE WHEN CHOOSING ANCHOR IMAGES:

1. When sampling anchor images, make sure that **their qualities are good enough**
   → Not-so-great ones will hamper finding the correct slicing angles

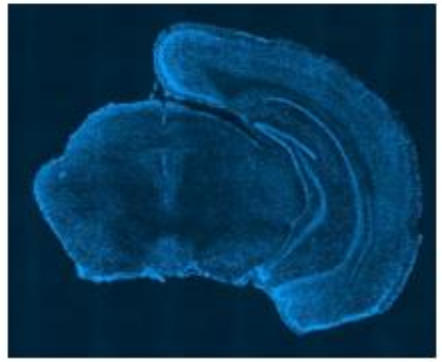
Good







Bad



STEP_2_Anchor_Image_Selection

IMPORTANT NOTICE WHEN CHOOSING ANCHOR IMAGES:

2. The AP positions of **non-anchor images** will be **INTERpolated** between anchor image positions

For example, if you have Images #1,#2,….#7, and set Images #2 & #6 as anchors,
Images #1 and #7 will not be further processed in the later stages,
because their AP positions can’t be interpolated

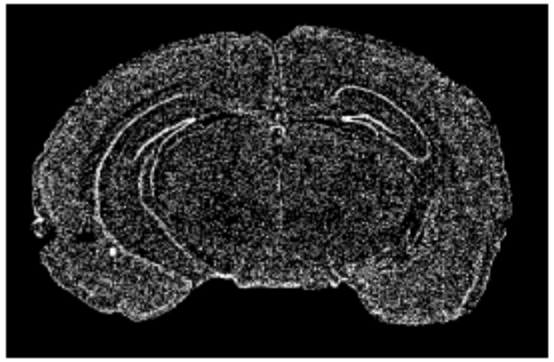
(Optional) STEP_3_Optional_Redraw_Slice_Boundary

- This is an OPTIONAL step to redraw the slice boundaries
  → Change the "threshold_scale" value in STEP_0_Parameters script

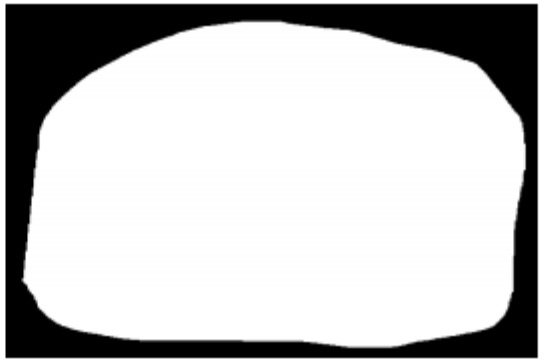
```
%% 3. Anchor Image IDs
anc_img_IDs= sort([1 3]);
img_IDs_reBoundary=[2];
threshold_scale = 1.5;
```

- Run the code, and you will be asked to draw a rough slice boundary;

Example Image #2



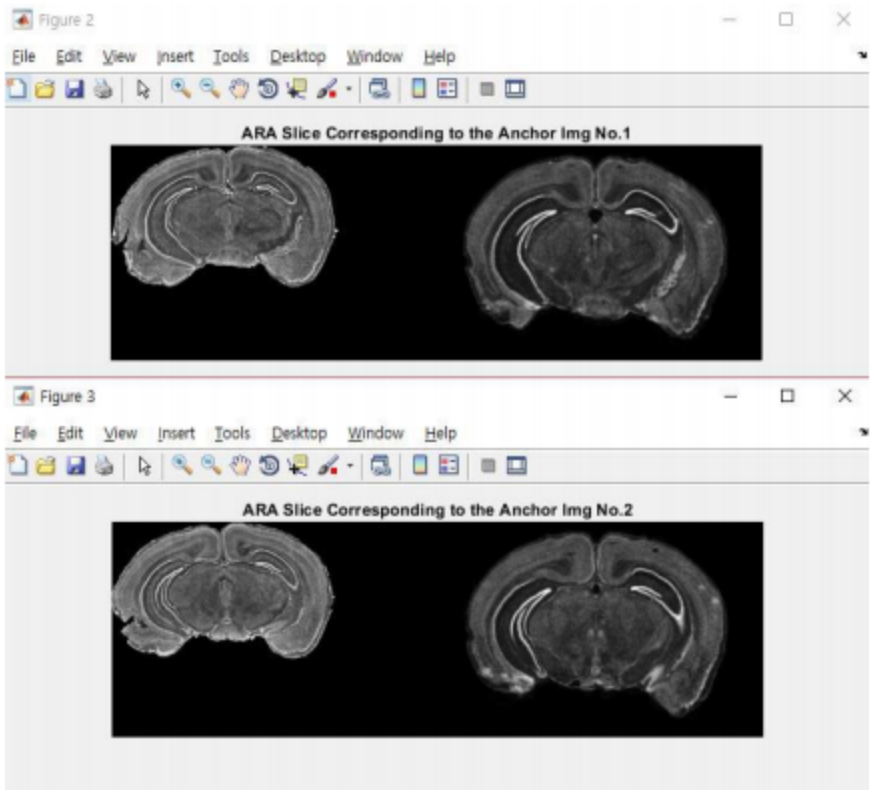
Binary mask of example image in various threshold scale



threshold_scale = **1.5** **0.1** **5**

STEP_4_Angle_Finder

- This script finds the slicing angles of the brain and AP positions of the anchor images sampled.
- In the command window, type “STEP_4_Angle_Finder” (or run the script in editor)

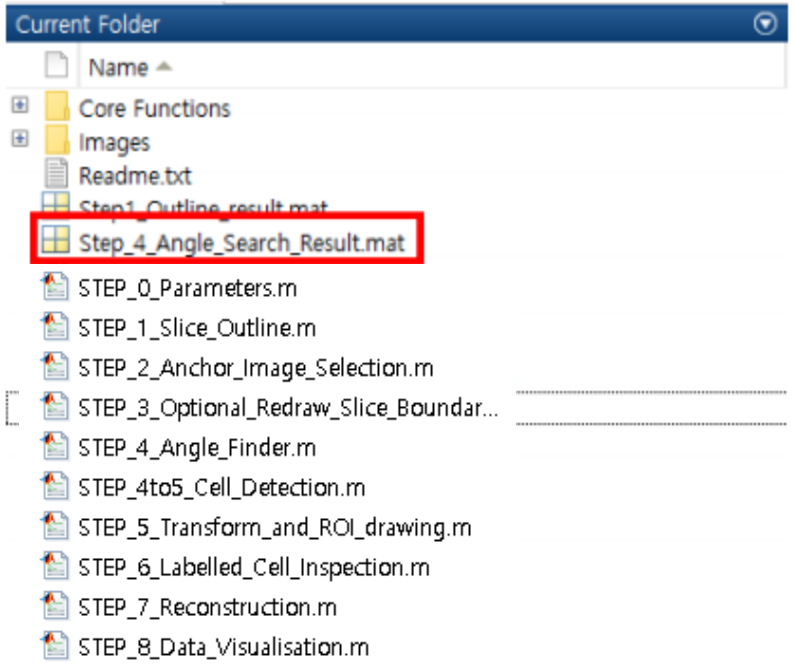



```
>> [yaw_stage5_max pitch_stage5_max]

ans =

     6     4
```

Result : Slicing angles (Yaw and Pitch)

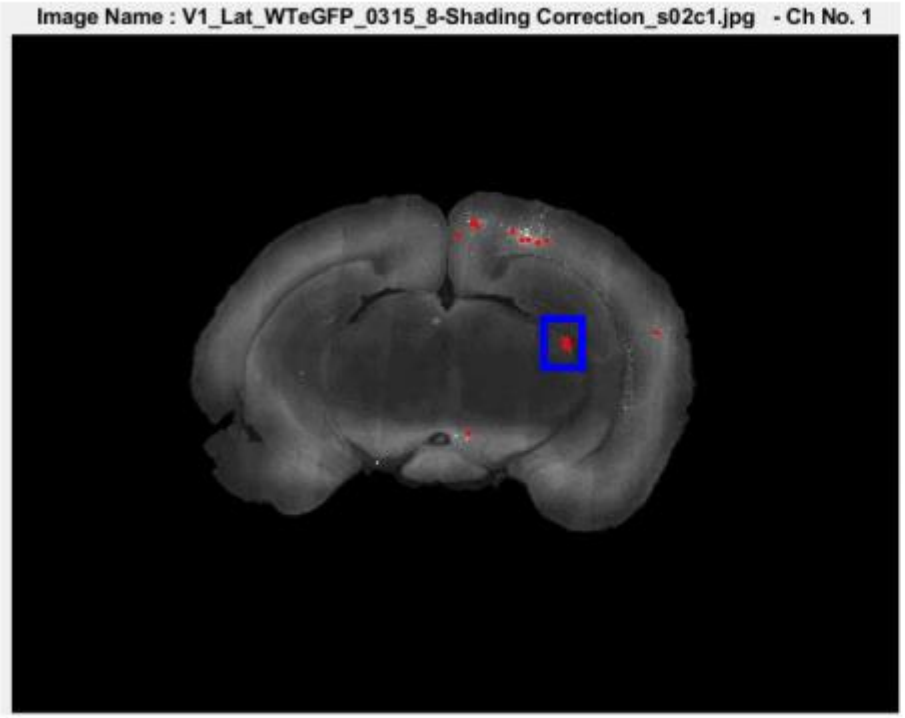


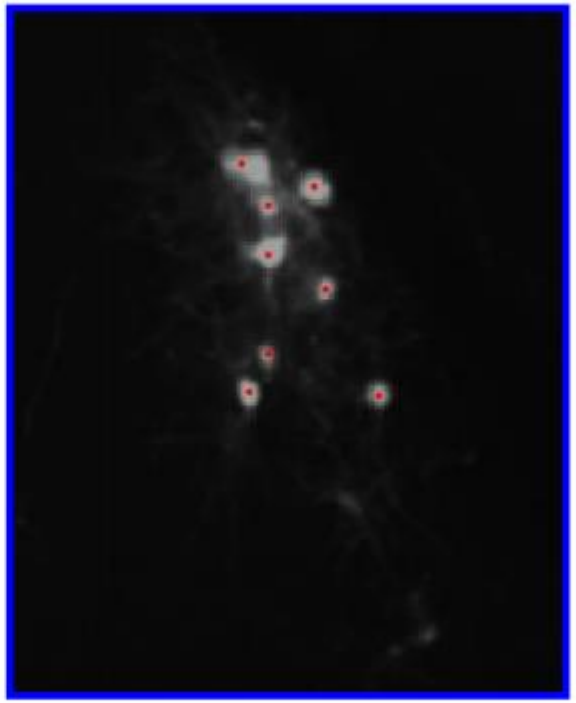
STEP_4to5_Cell_Detection

- This script finds the cells in each image sample
- In the command window, type “STEP_4_Angle_Finder” (or run the script in the editor)
- When finished, “Step_4to5_Cell_Detection_Result.mat” will be saved
- You may need to adjust parameter ‘cell_det_thresh’ in STEP_0_Parameters.m

STEP_5_Transform_and_ROI_drawing

- This script
  1. Projects obtained sampled images onto their corresponding ARA slices
  2. Warps detected neurons from original image to projected images





Result saved in a "STEP_5_Cell_Detection_Result.mat" and Images saved in each folders

STEP_5_Transform_and_ROI_drawing

- Step 4, Step4to5 and Step 5 are the most time-consuming stage
  So you may run the multiple scripts
  (if you already properly adjusted your parameters) by typing:
  >> Step_4_Angle_Finder;
  Step_4to5_Cell_Detection;
  STEP_5_Transform_and_ROI_drawing

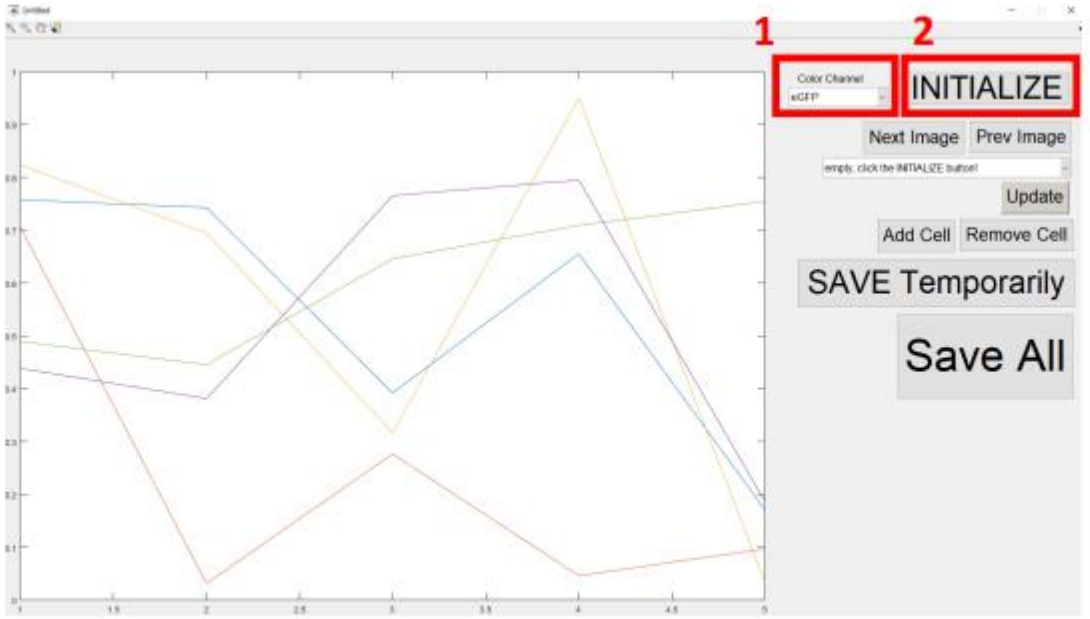
- Computational time depends on your computer specification and data size of the image (resolution highly matters).

- You may check the status after several hours from Step4 ~ Step5

(OPTIONAL) STEP_6_Labelled_Cell_Inspection

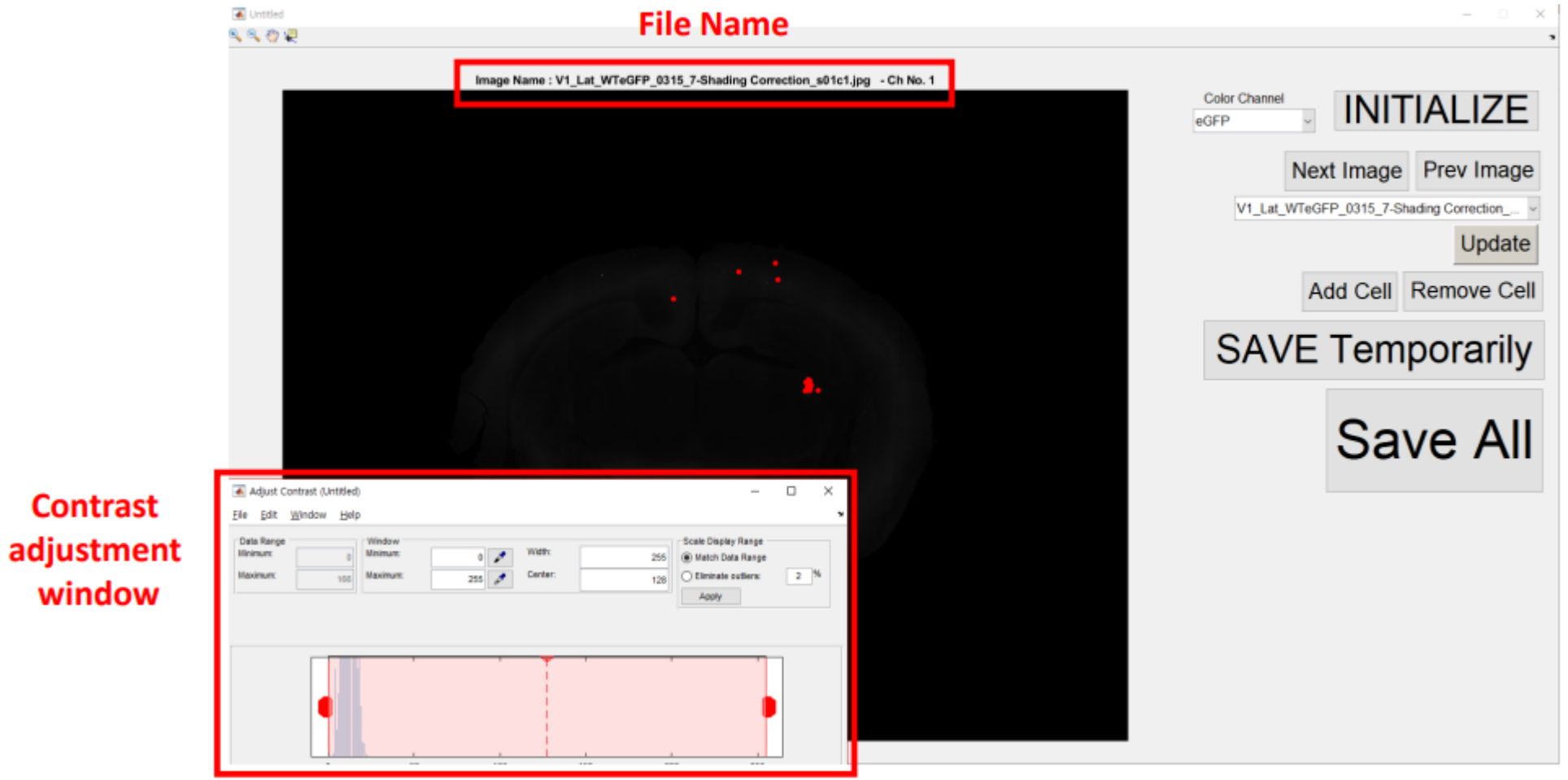
- You may wish to inspect the detected neurons in each image and
  1. delete false-positive neurons
  2. manually mark neurons that are not automatically detected

- In the command window, type “STEP_6_Labelled_Cell_Inspection”
  → A GUI window will appear
  → Set the color channel (if channel number > 1) for inspection then click on the initialize button



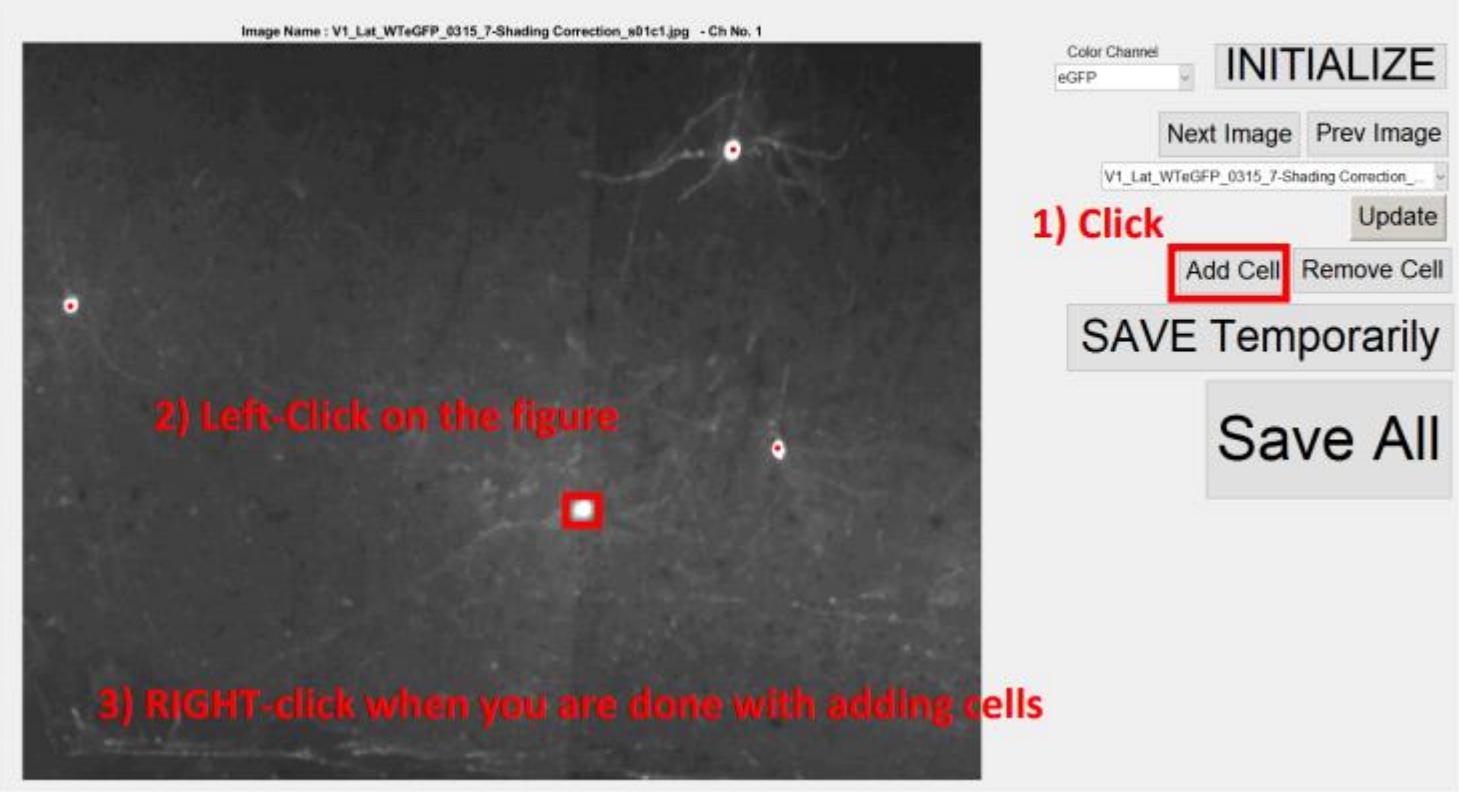
(OPTIONAL) STEP_6_Labelled_Cell_Inspection

- The first image of the ARA-matched image set will appear



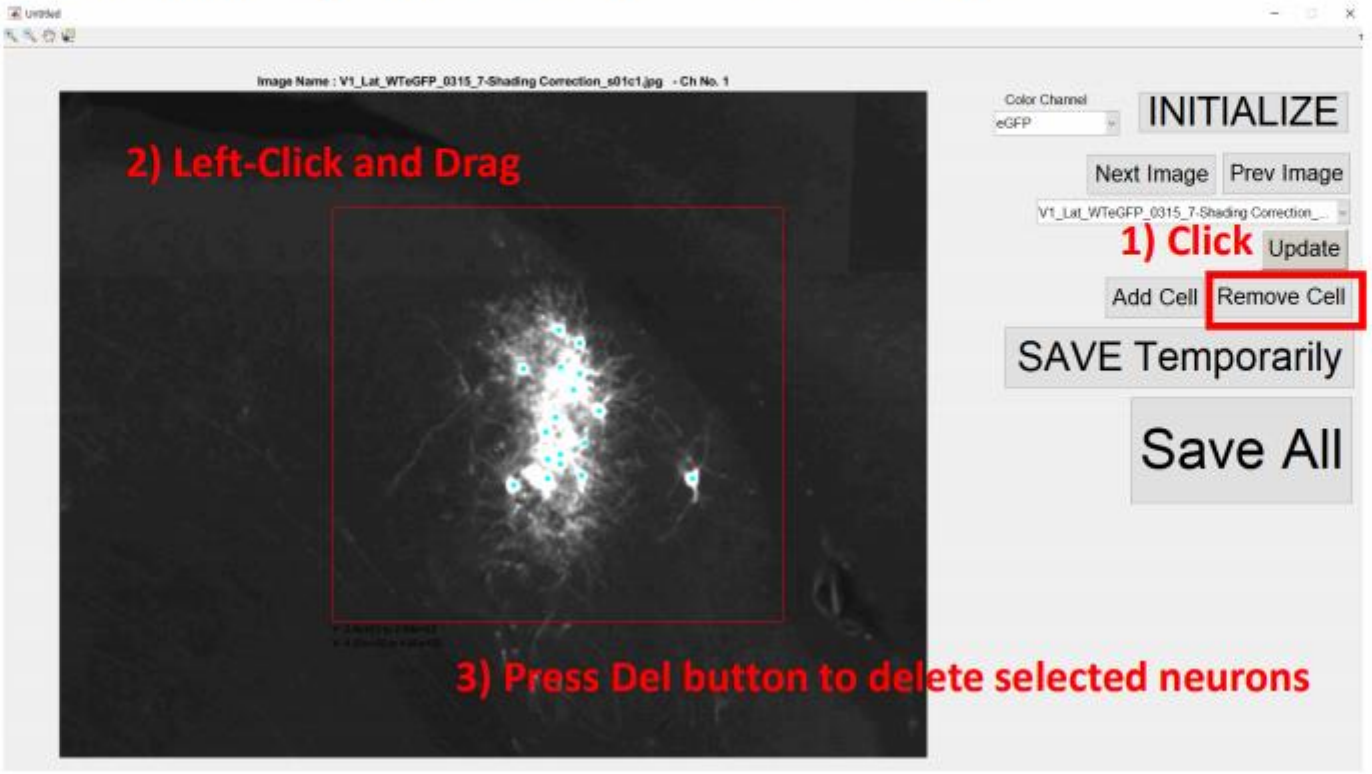
(OPTIONAL) STEP_6_Labelled_Cell_Inspection

- To add a cell,
  1) Click “Add Cell” in right GUI panel
  2) Left-click on the positions of non-detected neuron in the figure
  3) Right-click to terminate adding cell process.



(OPTIONAL) STEP_6_Labelled_Cell_Inspection

- To delete a cell,
  1) Click “Remove Cell” in right GUI panel
  2) Left-click and drag on the positions of unwanted neurons in the figure
  3) Press Del button on the keyboard to delete selected neurons



(OPTIONAL) STEP_6_Labelled_Cell_Inspection

- After the inspection,
  1) Click 'SAVE Temporarily' button; otherwise the edited result will not be updated
  2) Click 'Next Image' or 'Prev Image' to move on
     2-b) You can choose the image in the dropdown menu, and click update button
  3) When you are done with all the images, click 'Save All'.
     Otherwise, the matlab file ("STEP_5_Cell_Detection_Result.mat") will not be updated/saved

2) Click Next Image / Prev Image to move on

2b) Choose the image in dropdown menu & click Update button

1) Click – otherwise the edited result will not be updated

3) Press "Save All" to update and save all neuron data

STEP_7_Reconstruction

- After neuron detection,
  Type “STEP_7_Reconstruction” in the command window (or run the script)
  → This script locates labelled neurons in 3-D ARA and annotate them into different ROIs

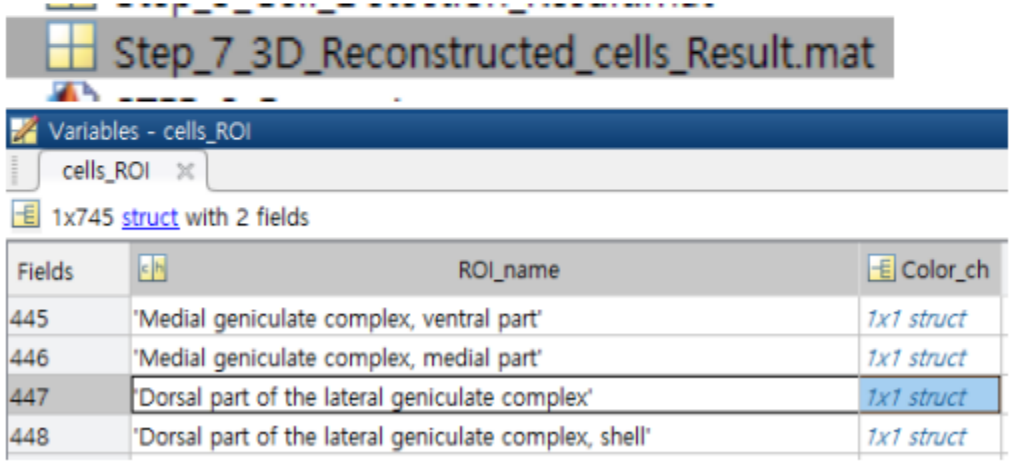
STEP_7_Reconstruction

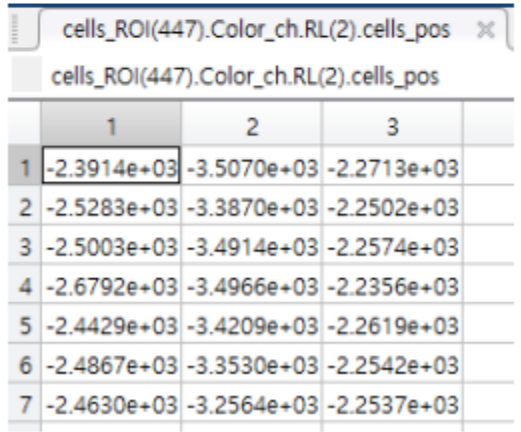
- (Optional) If you need location of labelled neurons in ARA

  1. Load the “STEP_7_3D_Reconstructed_cells_Result.mat”

  2. Navigate to the ROI that you are interested in

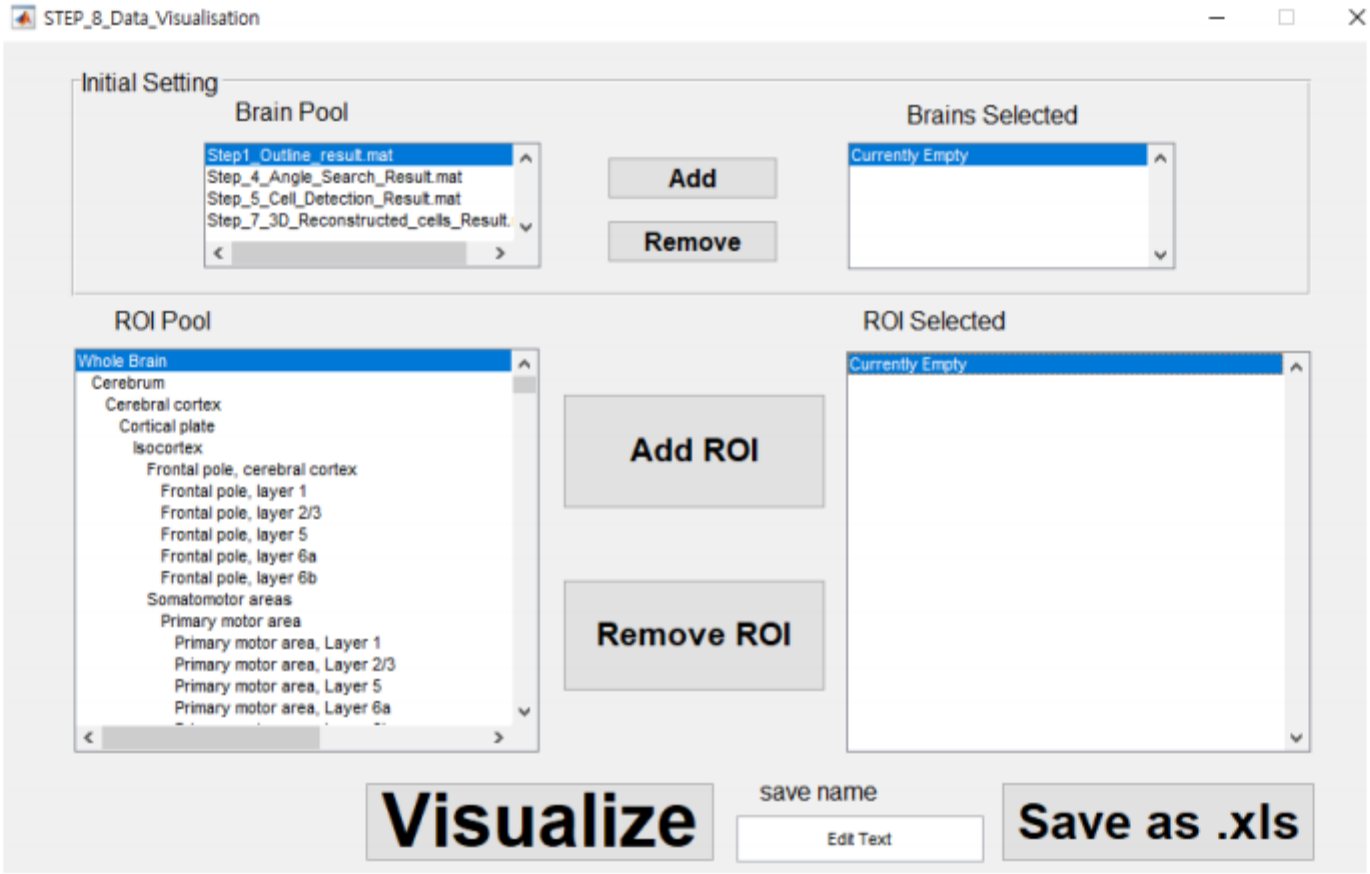
  3. You’ll see the 3-D coordinates of the detected cells in specified ROI
  1st, 2nd and 3rd column of the data corresponds to ML, DV, AP coordinates of neurons (um)





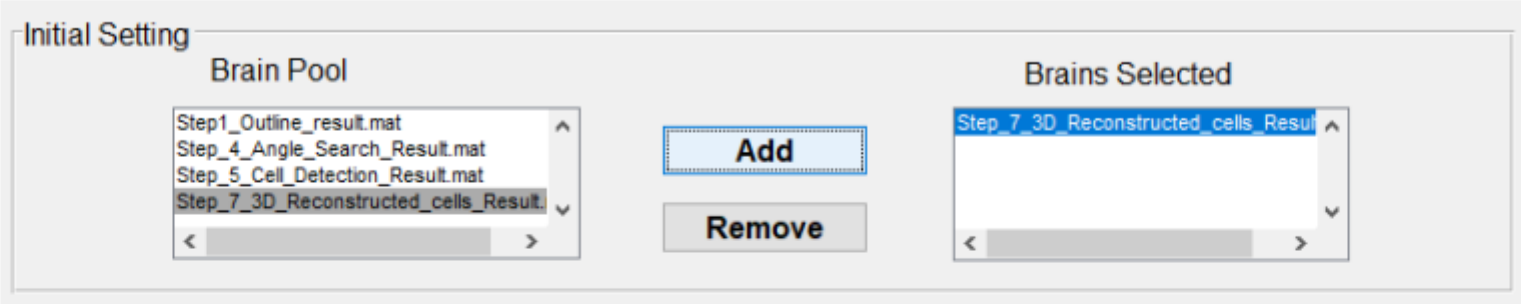
STEP_8_Data_Visualization

- Type “STEP_8_Data_Visualisation”, in the command window (or run the script)
  → GUI window will appear



STEP_8_Data_Visualization

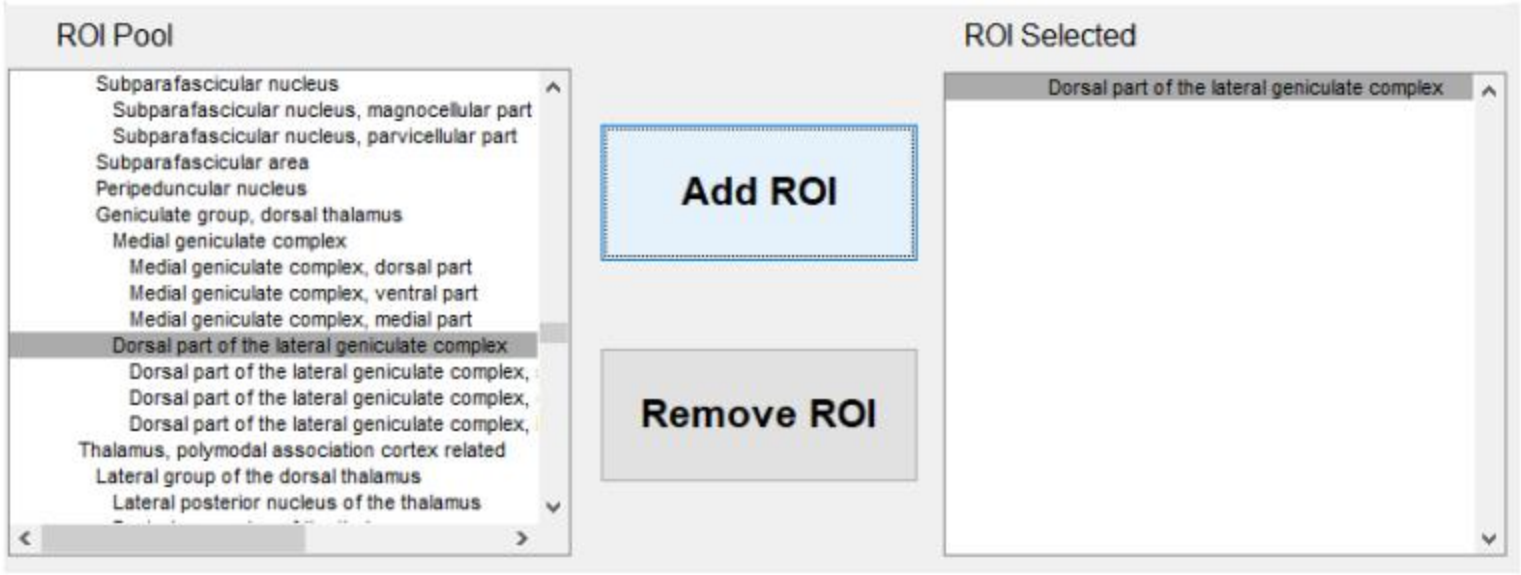
- Add the Step7 results that you like to see in 3D



- IMPORTANT:
  If you want to compare results from multiple mice,
  1) First you have to save the Step_7 results obtained from each mice.
  2) Change the file names.
     e.g. Mouse #1's Step_7_3D_Reconstructed_cells_Result.mat -> Mouse_1_VISp.mat
     Mouse #2's Step_7_3D_Reconstructed_cells_Result.mat -> Mouse_2_VISp.mat
  3) Locate those files in a same folder
  4) Run step 8 script

STEP_8_Data_Visualization

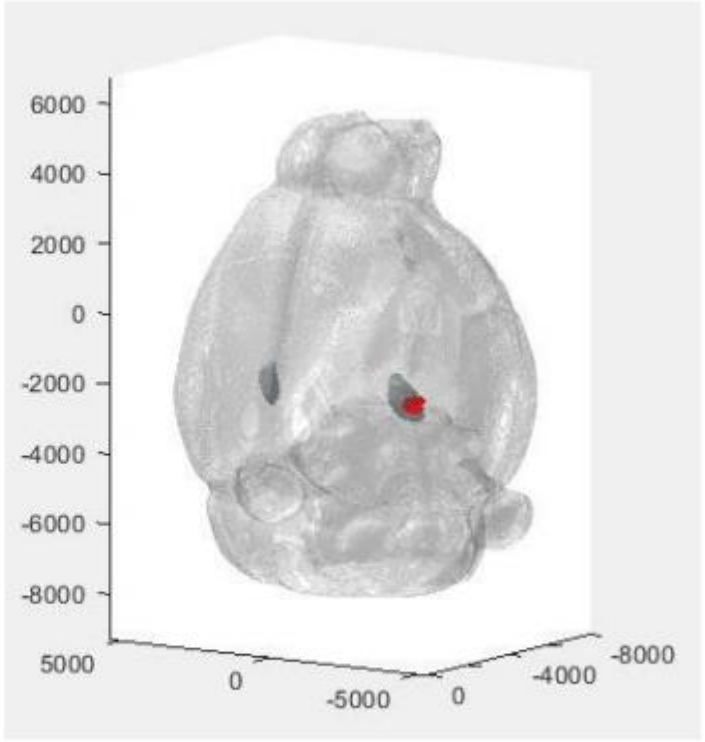
- Specify the ROIs that you want to visualize



- If you want to see the big picture of the result, you can add “whole brain” ROI.
- Do not add all the ROIs, unless you have different purpose (MATLAB may slowed down)

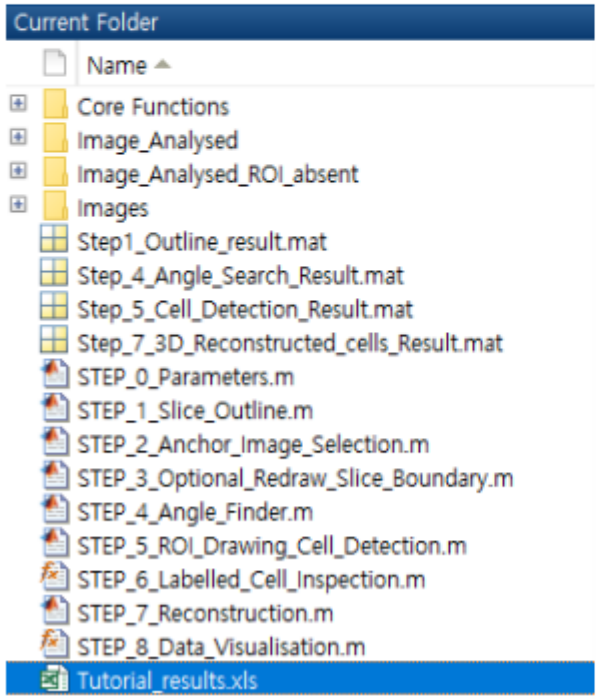
STEP_8_Data_Visualization

- Click on VISUALIZE button



STEP_8_Data_Visualization

- If you want to export counted cell results in each ROI, type the save filename and click “Save as .xls” button



	A	B	C	D
1	ROI name		Step_7_3D_Reconstructed_cells_Result.mat	
2			Left hemi (cells)	Right hemi (cells)
3		Dorsal part of the lateral geniculate complex	0	55
4				
5				
6				
7				
8				

Possible Errors

- Most of the error was caused by incorrect parameters of the image
  Please re-check the parameters you have set in the “STEP_0_Parameters”

- In case when you re-start the analysis and if the error related with the file ‘directory’ occurs,
  1) Go to folder directory
  2) Run STEP_0_Parameters
  3) Re-run the script (ex – STEP 7)

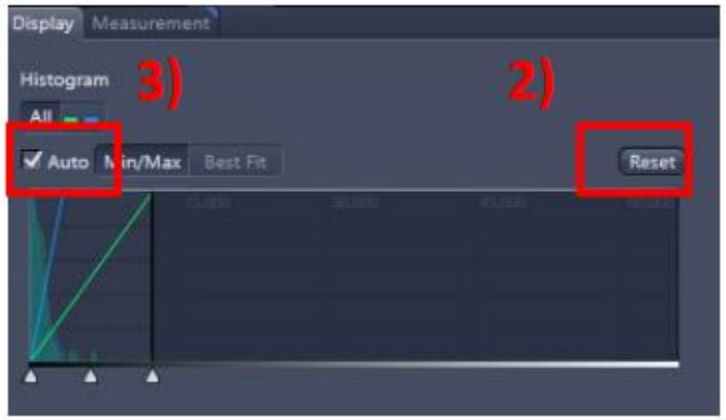
- If you still get errors after trying the aforementioned measures or have any other queries

  contact the corresponding author, Prof. Se-Bum Paik (sbpaik@kaist.ac.kr)

Tips if you use AxioScanner (Zeiss)

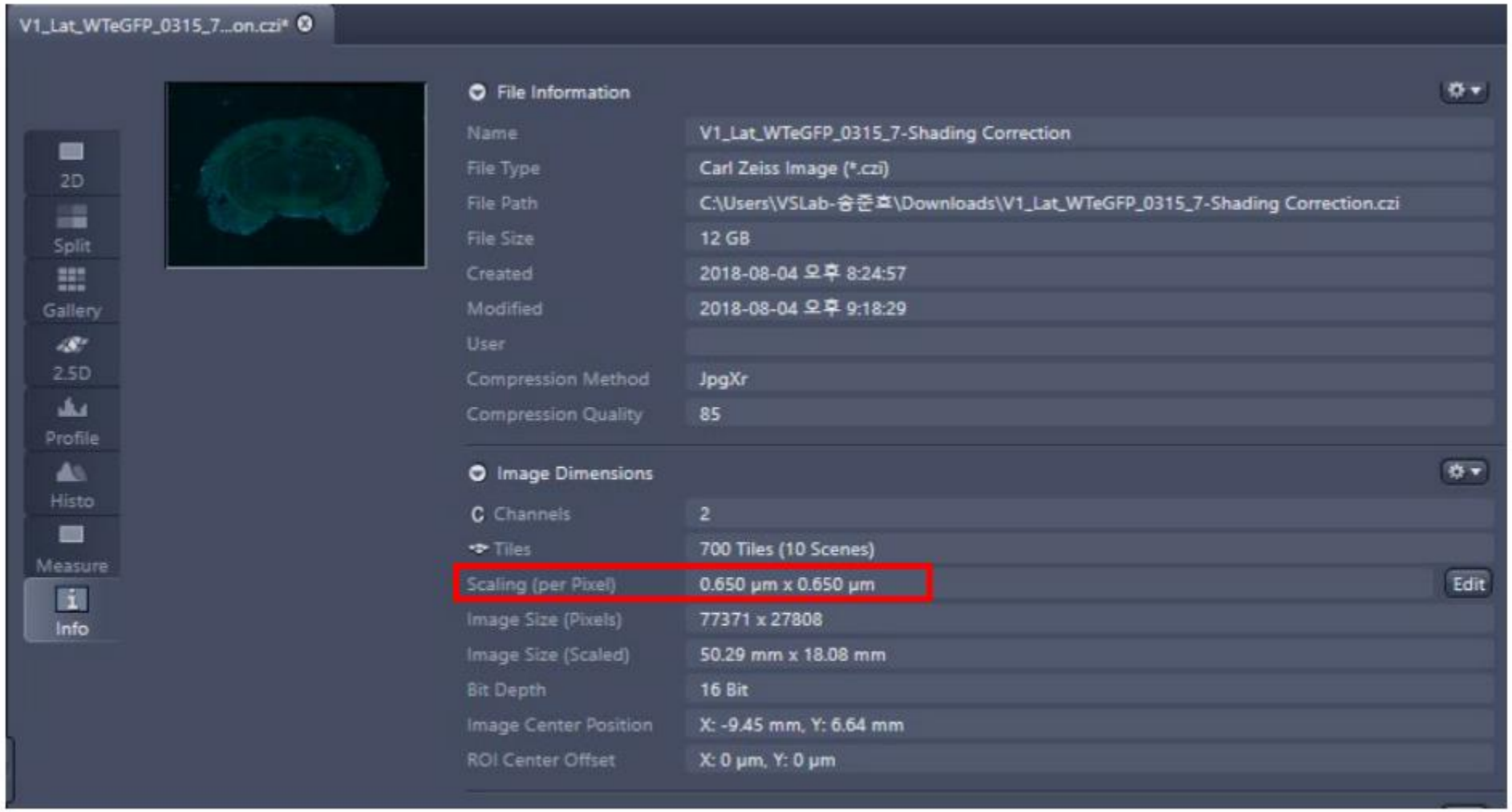
- We have done the following steps when exporting slice images using ZEN

1)



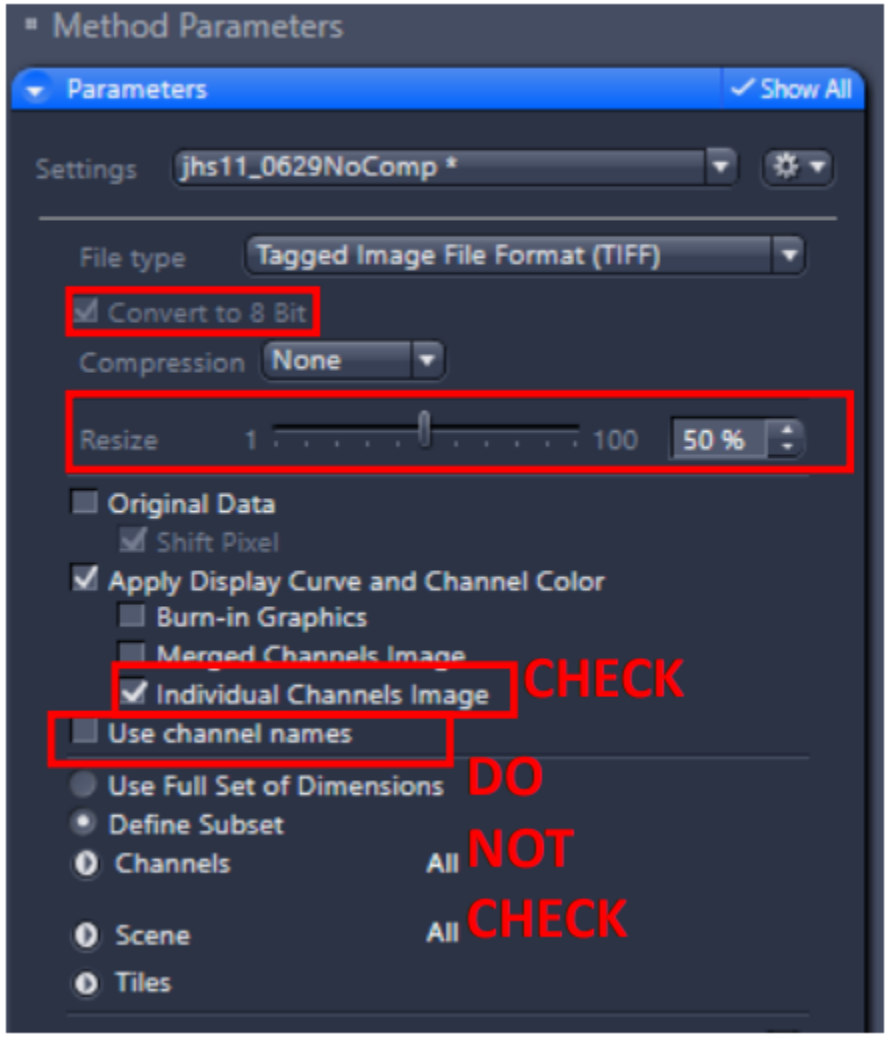
1. Intensity Normalization
   - Check the "Scene" box
   - Click on the Reset button
   - then check the "Auto" box
   - Save the . czi file (ctrl+s for Windows)
     → Intensity normalized for each slices

Tips if you use AxioScanner (Zeiss)



2. Check your image metadata → **Check parameters in STEP_0_Parameters**

Tips if you use AxioScanner (Zeiss)



3. Export Settings
   - Convert to 8-Bit
   - Optional (Resize factor = 50%)
     → This will downsample images by 50%
     → In this case, the xy_pix should be doubled
     i.e. xy_pix = 0.650 *2 in STEP_0
     → This will speed up the computation

   - Please check “Individual channel images “
   - Please DO NOT check “Use channel names”