pyElectrode: manual

Pierre M. Daye

October 3, 2012

1 pyElectrode usage

Neurophysiogists want to place the tip of an electrode in a specific area of the brain. The coordinates of this area can be obtained from standard stereotaxic atlases. However, individual brains will not align with the atlas exactly. Additionally, for chronic recordings, electrodes are placed relative to a chamber attached to the animal's skull. Thus, the user wants to know where the area of interest is in chamber coordinates, not stereotaxic coordinates.

During experiments, the position of the electrode tip where each neuron was recorded is kept. One of the main uses of pyElectrode is to plot the position of these electrode locations onto a correctly oriented MRI slice. These neurons can be visualized and localized from multiple perspectives (e.g. sagittal, coronal). In addition to this primary function, the program has the capacity to (1) prospectively compute the desired chamber hole and location for a neuron that is manually specified on the MRI, (2) visualize the orthogonal "chamber projection" onto cross sections at different depths perpendicular to an electrode penetration, (3) visualize the coronal section corresponding to the electrode tip, and (4) output files in the form of vector graphics.

2 Graphical user interface layout

Figure 2 presents the layout of the graphical user interface. It is composed of three main parts:

Toolbox panel. This panel contains four tabs: Clip MRI (see 4.1), Chamber (see 4.2), View (see 4.3) and External point (see 4.4). Each tab contains tools and functions that will be described below.

View panel. This panel shows the loaded MRI. It is composed of three views:

View A. A slice in the MRI at a level indicated by the blue lines in views B and C (i.e., the color of the lines in the other views corresponds to color of the border of this view).

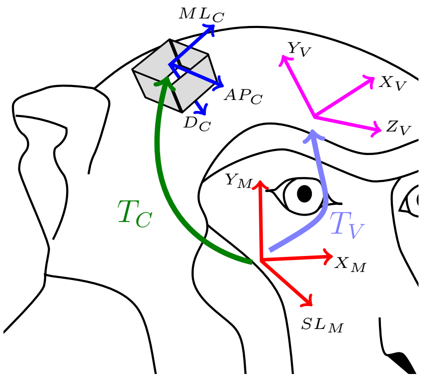


Figure 1: Coordinate systems and transformations. Three coordinate systems are defined. The MRI one (M subscripts and red lines), the chamber one (C subscript and blue lines) and the user view one (V subscript and pink lines). Two transformations between the coordinate systems are also defined. T_C represents the transformation from the chamber coordinates to the MRI coordinates (green arrow) while T_V represents the transformation between the MRI reference frame and the user view coordinate system (purple arrow).

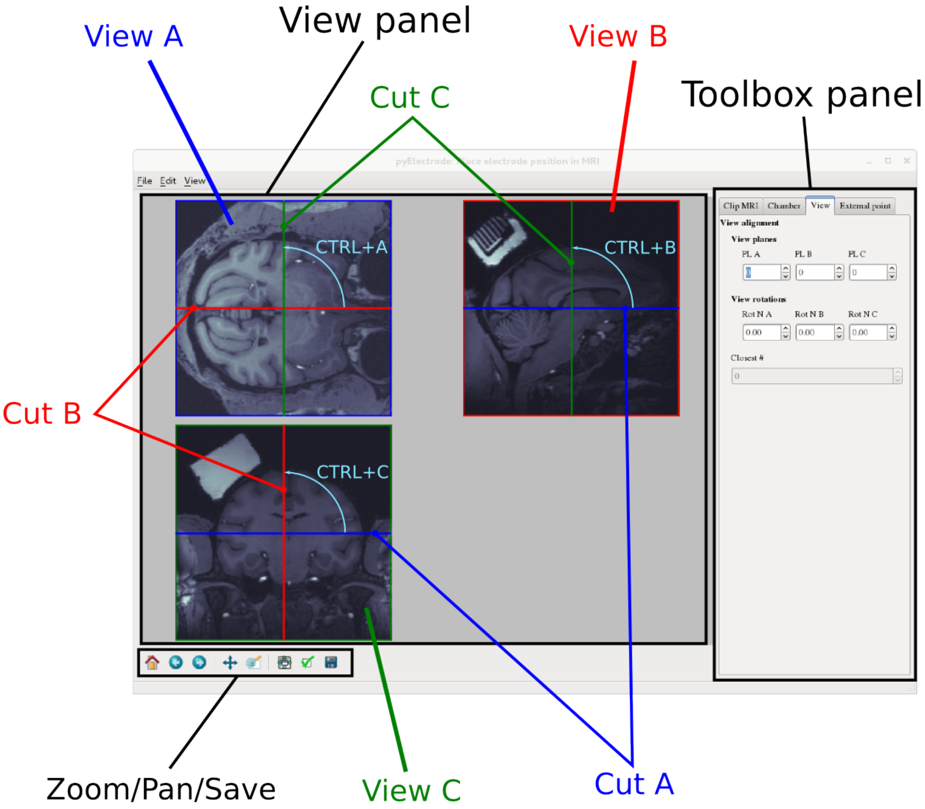
- **View B.** A slice in the MRI at a level indicated by the red lines in views A and C (i.e., the color of the lines in the other views corresponds to color of the border of this view).
- **View C.** A slice in the MRI at a level indicated by the green lines in views A and B (i.e., the color of the lines in the other views corresponds to color of the border of this view).

Zoom/Pan/Save. This panel is used to navigate in the current view or export the current view to a graphic file. The magnifier (🔍) is used to zoom in one of the three different views. The arrows (✥) are used to translate the current zoomed view. The home and forward/backward arrows (🏠 ◀ ▶) are used to redraw the original view (home picture) or to navigate in the zoom/pan history stack (arrows). The floppy drive icon (💾) is used to export the current view into a graphic file.

In addition to these three panels, pyElectrode has three menus at the top:

File

- **Open** Used to open a pyElectrode file (*.pye) (see 3.3).
- **Load MRI** Used to load a series of DICOM format MRI files (see 3.1).



View alignment

View planes

PL A: 0
PL B: 0
PL C: 0

View rotations

Rot N A: 0.00
Rot N B: 0.00
Rot N C: 0.00

Closest #: 0

Figure 2: Graphical user interface.

Save Used to save the current state of the program in a pyElectrode file (*.pye) (see 3.2).

Export plot Export the current view to a graphic file.

Quit Quit the program.

Edit

Grid Add a grid to the three views. The spacing is fixed to one centimeter.

Rot 90 A Rotate the MRI around the normal of view A by 90 degrees as represented by the light blue arrow in View A in Fig 2 (shortcut: CTRL+A).

Rot 90 B Rotate the MRI around the normal of view B by 90 degrees as represented by the light blue arrow in View B in Fig 2 (shortcut: CTRL+B).

Rot 90 C Rotate the MRI around the normal of view C by 90 degrees as represented by the light blue arrow in View C in Fig 2 (shortcut: CTRL+C).

Set record and comment files Choose a name for a record file and comment file (see 4.4.4).

View

Increase brightness Increase the brightness of the MRI images (shortcut: CTRL++).

Decrease brightness Decrease the brightness of the MRI images (shortcut: CTRL+-).

Reset brightness Reset the brightness of the MRI images to their original settings (shortcut: CTRL+R).

3 Load/Import MRI/Save

3.1 Import DICOM files

To import a series of DICOM files acquired during an MRI scan, go to File → Load MRI (shortcut: CTRL+L). Select the folder containing all the images of the scan. Once the folder is selected, you have to specify the file pattern of the MRI files in a second window. For example if your images are named [IM_AAA_0001, IM_AAA_0002, IM_AAA_0003, ..., IM_AAA_0128], type 'IM_AAA_' in the request field. Then the program will load all the DICOM files corresponding to this file pattern and will normalize the size of the voxels by ensuring that the three dimensions of each voxel are equal[1].

[1] Currently, the software needs square pixels in every acquired images. An interpolation is made to ensure that the distance between two slices corresponds to the size of a square pixel.

3.2 Save

You can save the current state of your work in a pyElectrode file (extension: pye) by going to File → Save (shortcut: CTRL+S). A window will open so you can select the folder in which you want to save the file. A second window will open where you can give the name of the file without an extension[2].

3.3 Open

You can load a previous work state in pyElectrode. Go to File → Open (shortcut: CTRL+O). Select the *.pye file you previously saved. All the configuration parameters are restored (chamber orientation/electrode files) but not the last view settings and the external point settings.

4 Toolbox panel

4.1 Clip MRI

Once the MRI has been loaded, the acquired field of view can be reduced by clipping the MRI around the region of interest. This will reduce the amount of data and therefore increase the speed of the computations. Figure 3 represents the views associated with the Clip MRI tab. The clipped MRI will always be a cubic MRI. Once the "Allow Clipping" checkbox has been checked, the following controls are accessible:

PL A- Increase to clip more on the left side of view A and B (see PL A- arrows in Fig. 3).

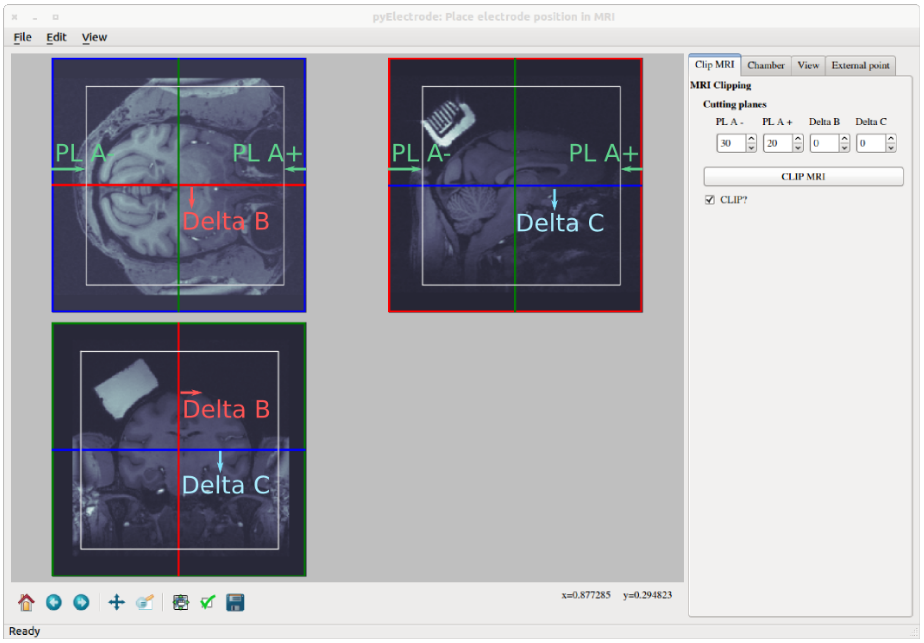
PL A+ Increase to clip more on the right side of view A and B (see PL A+ arrows in Fig. 3).

The upper and lower borders in view A are automatically computed to ensure a cubic MRI. Two controls have been added to translate the cube and center it as desired:

Delta B Translate the cube downward (positive value) or upward (negative value) in view A and thus rightward (positive value) or leftward (negative value) in view C (see Delta B arrows in Fig. 3).

Delta C Translate the cube downward (positive value) or upward (negative value) in views B and C (see Delta C arrows in Fig. 3).

[2] Programming note: this method is unconventional but is used because of a bug in the window management of pyQT that can make the traditional save window unstable and crash the program.



MRI Clipping

Cutting planes

PL A -	PL A +	Delta B	Delta C
30	20	0	0

☑ CLIP?

x=0.877285 y=0.294823

Figure 3: Clipping MRI tab and associated views.

Once the borders of the MRI has been correctly defined, press the “CLIP MRI” button. **Caution: this changes the internal data in pyEletrode (not the original DICOM files) and can not be undone. If you already have defined a coordinate system, you will most likely have to redefine it.**.

Once this is done, uncheck the “Allow Clipping” checkbox.

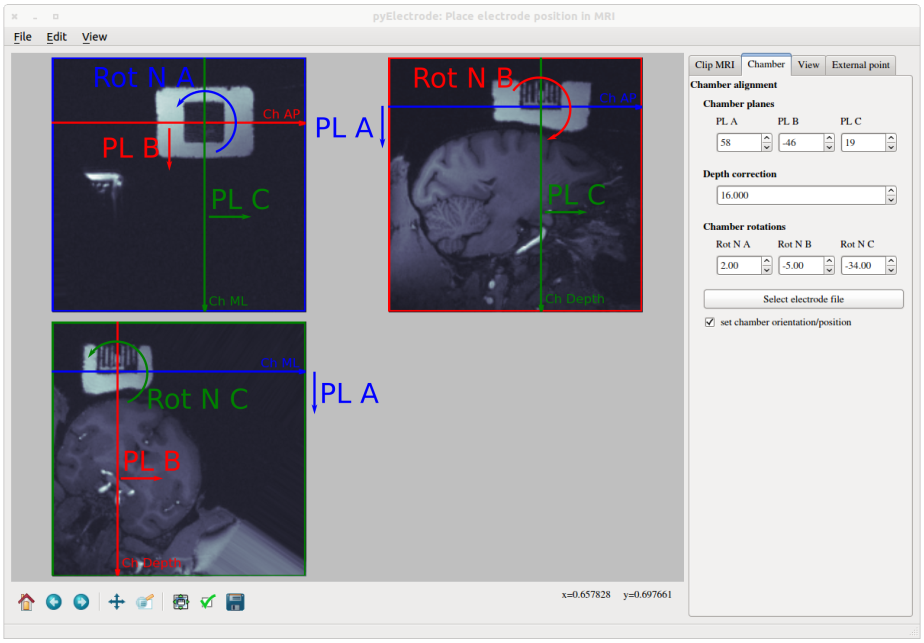
4.2 Chamber

The second tab, Chamber, is used to define the position and the orientation of the chamber in the MRI as well as the specification of a file containing electrode coordinates. Figure 4 shows the layout associated with this tab.

4.2.1 Chamber alignment

The different controls to orient the chamber are accessible once the “set chamber position/orientation” checkbox is checked.

To align the chamber, you must know the location of the origin and the orientation of the coordinate system of the chamber. Then, you must center the intersection of the three views (represented by the green, red and blue lines in each view) with the origin of the chamber using:



Chamber alignment

Chamber planes

PL A	PL B	PL C
58	-46	19

Depth correction

16.000

Chamber rotations

Rot N A	Rot N B	Rot N C
2.00	-5.00	-34.00

☑ set chamber orientation/position

Figure 4: Chamber tab and associated views.

PL A Select an upward (positive value) or a downward (negative value) slice in the MRI for view A in views B and C (see PL A blue arrows in Fig. 4).

PL B Select a downward (positive value) or an upward (negative value) slice in the MRI for view B in view A and a rightward (positive value) or leftward (negative value) slice for view B in view C (see PL B red arrows in Fig. 4).

PL C Select a rightward (positive value) or a leftward (negative value) slice in the MRI for view C in views A and B (see PL C green arrows in Fig. 4).

Then you have to align the chamber coordinates with the coordinate system shown in Fig. 4. You rotate the MRI around the defined center of the chamber to align the axis of the coordinate system to the known orientation of your chamber axes using:

Rot N A Rotate the MRI anticlockwise (positive value) or clockwise (negative value) around a normal of view A passing through the origin of the chamber (see Rot N A blue arrow in Fig. 4).

Rot N B Rotate the MRI anticlockwise (positive value) or clockwise (negative value) around a normal of view B passing through the origin of the chamber (see Rot N B red arrow in Fig. 4).

Rot N C Rotate the MRI anticlockwise (positive value) or clockwise (negative value) around a normal of view C passing through the origin of the chamber (see Rot N C green arrow in Fig. 4).

A correct orientation and localisation of the chamber coordinate system is critical for a proper use of pyElectrode. Thus, to maximize the utility of this method, it is critical to align MR images on the orientation of the recording chamber. Because every computation is based on the correct alignment of the chamber by the user, an error will be propagated to the position of the electrode in the MRI. For example, an error of 1 deg in the orientation of the chamber can affect the precision of the estimate of the electrode position up to $\approx$0.35 mm for a 20 mm deep penetration. To prevent this type of error, we recommend the user to follow the following two steps. First, the square reference grid must be placed inside the chamber accurately. Second, the MR images must be aligned on individual grid holes accurately. This can be done by observing a movie across adjacent MR images; even a slight tilt can be detected as a wave of grid hole images.

Finally, because it could be difficult to position the center of the chamber at the exact depth corresponding to the exit point of the guide tube, you can set a bias (represented in millimeters) in the spinbox "depth correction" to account for the difference in depth between the specified origin and the actual exit point of the guide tube. This will not modify the center of the chamber, it will just correct the depth values that you enter to specify an electrode position.

4.2.2 Electrode coordinates file

Once the chamber has been correctly located and oriented (see previous section), you can select a text file containing the coordinates of the electrode tip kept during experiments by clicking on the "Select electrode file" button. Then you select a text file containing six columns with columns corresponding to:

1. Classification 1: an integer used for the first classifier. This number could be used later to show only a subset of the recorded positions (see section 4.3.2 for details). For example, 1 could represents dopamine neurons or neurons from a particular penetration while 9999 would represents neurons with no particular feature.

2. Classification 2: an integer used for the second classifier. This number could be used later to show only a subset of the recorded positions (see section 4.3.2 for details). For example, 1 could represents dopamine neurons or neurons from a particular penetration while 9999 would represents neurons with no particular feature.

3. ML coordinate: a number representing the medio-lateral coordinate in millimeters of the electrode position in the grid system.

4. AP coordinate: a number representing the anterio-posterior coordinate in millimeters of the electrode position in the grid system.

-1	1	0	0	10000	1
1	99	0	0	10200	2
2	1	4	0	17530	3
3	1	0	-3	15550	4

Table 1: Specifications for the text file containing the position and the classification of the recorded neurons. The first column represents the classification number 1. The second column represents the classification number 2. The third column represents the medio-lateral chamber coordinate of the electrode [mm]. The fourth column represents the anterio-posterior chamber coordinate of the electrode [mm]. The fifth column represents the depth of the electrode [μm]. The sixth column represents the number of the neuron.

5. Depth coordinate: a number representing the electrode depth in micrometers.

6. A integer for the neuron (optional)

An example of file structure is presented in table 1.

This type of text file can be easily generated from a large variety of software (e.g. Excell, LibreOffice Calc, Scilab, Matlab, etc.). For example, if your data are saved in Matlab, you can create a matrix with the correct structure. Then, to generate a correct file from your Matlab matrix, use the following command:

```
save(FileName,’MatrixVariableName’,’-ascii’)
```

FileName is a string representing the name of the file and its path. The name of the matrix to save must be written as a string. Once this is done, you can read the corresponding file in pyElectrode.

If a pyElectrode file is saved after loading a matrix of electrode positions, those positions will be saved in the pyElectrode file.

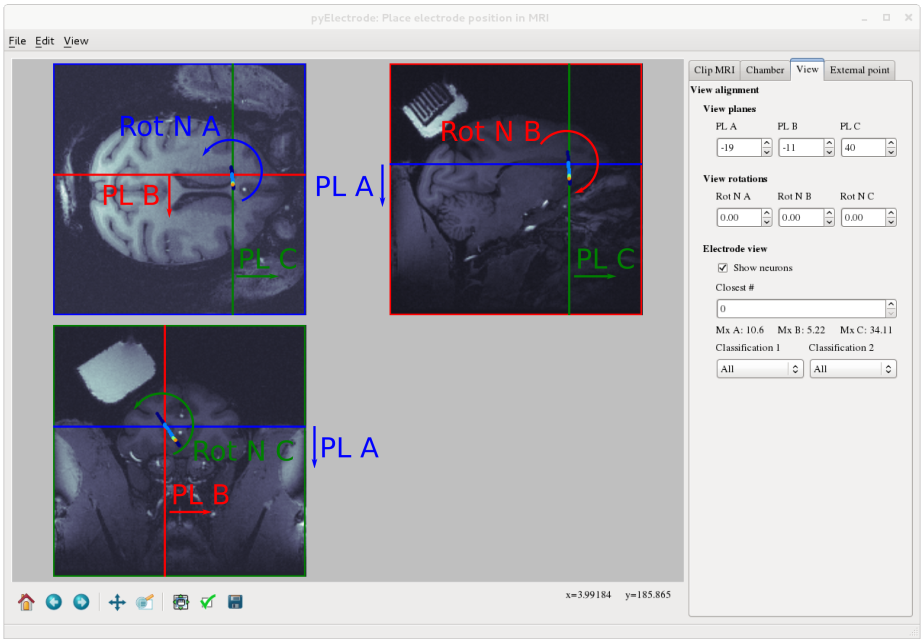
4.3 View

The third tab, View, is used to navigate in the MRI to show slices at a user selected position and orientation. It is also used to select which category of neurons will be shown. Finally, you can select whether all the neurons or only a closer subset of them must be shown. Figure 5 shows the layout associated with this tab.

4.3.1 View alignment

Controls to orient and center the view are available as the ones used in “chamber alignment”, section 4.2.1. For translations:

PL A Select an upward (positive value) or a downward (negative value) slice in the MRI for view A in views B and C (see PL A blue arrows in Fig. 5).



View alignment

View planes

PL A: -19
PL B: -11
PL C: 40

View rotations

Rot N A: 0.00
Rot N B: 0.00
Rot N C: 0.00

Electrode view

☑ Show neurons

Closest #: 0

Mx A: 10.6 Mx B: 5.22 Mx C: 34.11

Classification 1: All
Classification 2: All

x=3.99184 y=185.865

Figure 5: View tab and associated views.

PL B Select a downward (positive value) or an upward (negative value) slice in the MRI for view B in view A and a rightward (positive value) or leftward (negative value) slice for view B in view C (see PL B red arrows in Fig. 5).

PL C Select a rightward (positive value) or a leftward (negative value) slice in the MRI for view C in views A and B (see PL C green arrows in Fig. 5).

and for the orientation:

Rot N A Rotate the MRI anticlockwise (positive value) or clockwise (negative value) around a normal of view A passing through the intersection of the green and red lines (see Rot N A blue arrow in Fig. 5).

Rot N B Rotate the MRI anticlockwise (positive value) or clockwise (negative value) around a normal of view B passing through the intersection of the blue and the green lines (see Rot N B red arrow in Fig. 5).

Rot N C Rotate the MRI anticlockwise (positive value) or clockwise (negative value) around a normal of view C passing through the intersection of the blue and the red lines (see Rot N C green arrow in Fig. 5).

4.3.2 Electrode positions

Once a file with electrode positions has been loaded, the "show neurons" checkbox and the "Classification 1" and "Classification 2" list boxes are enabled. If the "show neurons" checkbox is checked, **the projection** of the position of each neuron present in the loaded text file will be shown on each of the views. Thus, to correctly visualize position of the neuron in the MRI, you have to translate the views such that they intersect at the location of the neuron you want to see. The color of each neuron is automatically determined by the "classification 2" parameter (see section 4.2.2 for a description of the neuron's parameters). "classification 1" parameters does not change the color of the neurons.

The other items in the electrode view are:

Closest # Determines the number of neurons to plot on each view. When equal to zero, all the neurons are plotted. When the number is bigger than zero, N>0, the program computes the absolute distance between each neuron and each view plane. Then, it orders them and shows on each view the N closest points to the current view. **Thus each view does not necessarily plot the same neurons**.

Mx A: xx.xx Displays the maximum distance between the current position of the A slice and the displayed neurons in millimeters.

Mx B: xx.xx Displays the maximum distance between the current position of the B slice and the displayed neurons in millimeters.

Mx C: xx.xx Displays the maximum distance between the current position of the C slice and the displayed neurons in millimeters.

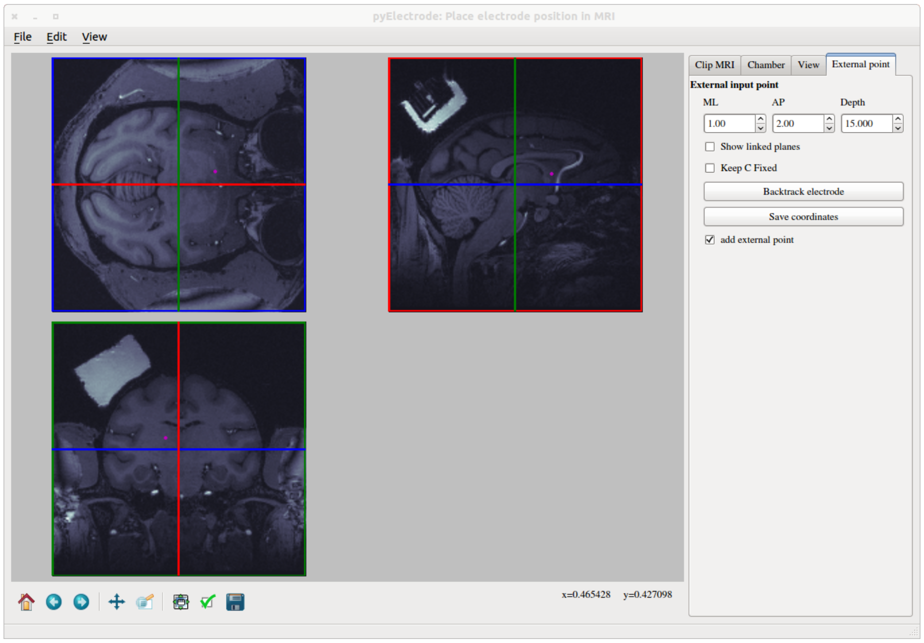
Classification 1 Select the subset of neurons with the corresponding "classification 1" parameter in the electrode file (see section 4.2.2 for details on the electrode file). When set to "All", no subset is chosen as a function of the "classification 1" parameter.

Classification 2 Select the subset of neurons with the corresponding "classification 2" parameter in the electrode file (see section 4.2.2 for details on the electrode file). When set to "All", no subset is chosen as a function of the "classification 2" parameter.

The program will plot the subset of neurons corresponding to the intersection between the subset defined by the "classification 1" parameter and the subset defined by the "classification 2" parameter. Thus if both are set to zero, all the neurons in the file will be plotted.

4.4 External point

The last tab, external point, is used to plot a point corresponding to particular chamber coordinates. The "external point" tab and its associated views is shown in Fig. 6.



External input point

ML 1.00 AP 2.00 Depth 15.000

☐ Show linked planes

☐ Keep C Fixed

☑ add external point

x=0.465428 y=0.427098

Figure 6: External point tab and associated views.

The three spinboxes are used to define the coordinate of a hypothetical electrode penetration in chamber coordinates as defined in section 4.2.1:

ML Medio-lateral chamber coordinate of the electrode.

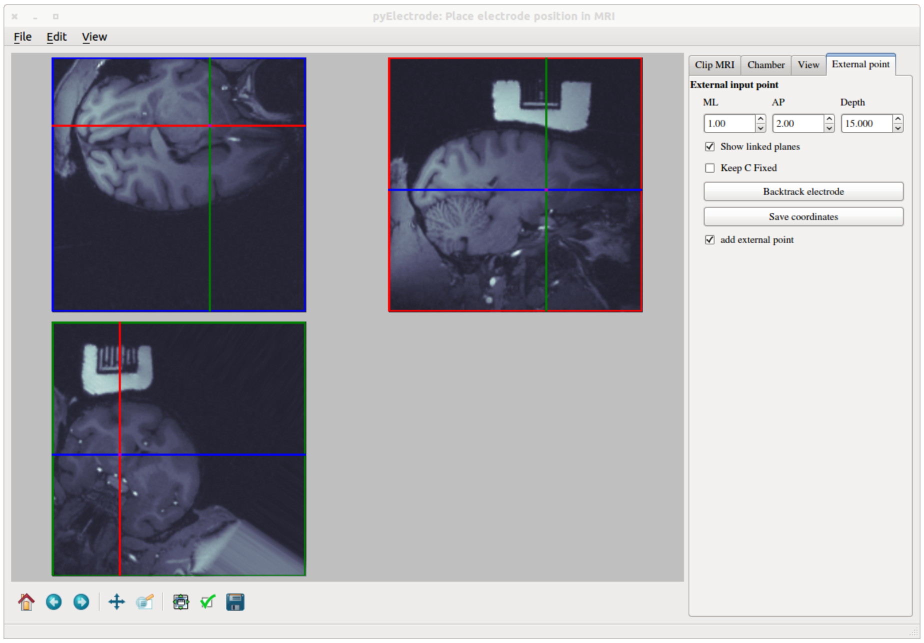
AP Anterio-posterior chamber coordinate of the electrode.

Depth Depth of the electrode penetration.

The projection of the position of the electrode tip corresponding to those coordinates will be plotted as a purple dot in the three views. Therefore, the purple point in the three views in Fig. 6 represents the projection of an electrode position with chamber coordinates equal to 1 mm medio-lateral, 2 mm anterio-posterior and 15 mm depth.

4.4.1 Show linked planes

If the "Show linked planes" checkbox is checked, the three views are changed (translated and rotated) to intersect at the position of the electrode tip specified by the external point coordinates defined in the previous section. The orientation of the views corresponds to the chamber orientation. Thus, as shown in Fig. 7, view A represents a slice that goes through the electrode tip and is oriented as if you were looking through the chamber.



External input point

ML	AP	Depth
1.00	2.00	15.000

☑ Show linked planes

☐ Keep C Fixed

☑ add external point

Figure 7: Linked planes views.

4.4.2 Keep C fixed

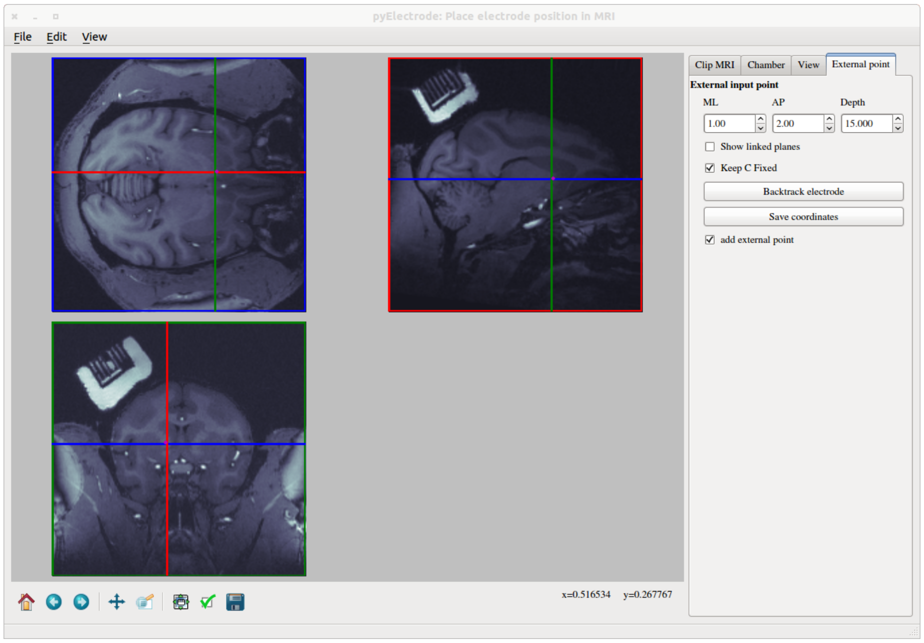
If the "Keep C fixed" checkbox is checked, the three views still intersect at the position of the electrode tip specified by the external point coordinates defined in section 4.4. However, the orientation of the views corresponds to the orientation defined in the view tab. This can be done to force a specific orientation. For example, Fig. 8 show the three views with an orientation that represents a coronal view in View C. This view makes it easier to relate what is plotted to a stereotaxic atlas.

4.4.3 Backtrack electrode

If you want to target a specific neural area, you can place the intersection of the three views using the view tab and then click the "Backtrack electrode" button. The program computes the chamber coordinates needed to reach this position and loads those values in the ML/AP/Depth spinboxes of the "External input point" tab.

4.4.4 Save coordinates

After you have already selected a filename and a path for saving the data (see section 2), if you want to save the chamber coordinates of the external point for



External input point

ML	AP	Depth
1.00	2.00	15.000

□ Show linked planes

☑ Keep C Fixed

☑ add external point

x=0.516534 y=0.267767

Figure 8: Keep C fixed views.

latter, you can click on the “Save coordinates” button. A window will open in which you can set the values for the two classification parameters as well as a comment attached to the current point. Then, the coordinate values as well as the classification parameters will be saved in YOURFILENAME_ExtElect.txt and the comments will be saved in YOURFILENAME_ExtElectLog.txt at the specified path.