Instruction manual for GCaMP6f transient signal extraction from image sequences

Tiantian Guo, Zichao Bian, Guoan Zheng, and Bin Feng

Department of Biomedical Engineering, University to Connecticut, Storrs, CT, 06269

Abstract: This protocol is developed to monitor the DRG neural activities by extracting the GCaMP6f transients from an image sequence. The DRG neural activities are recorded at a resolution of individual action potentials using a regular upright fluorescence microscope. However, both electrical and mechanical stimulation of DRG could introduce unwanted sample movement during the recording process. Although these movements are not obvious via visual inspection, the recorded time-lapse images will contain pixel-level displacements. To extract the accurate neuron activity signals, we first align the captured images using a template-matching approach via an ImageJ plugin in this protocol. We then extract the transient signal via morphological processing and marker-based watershed segmentation. This instruction provides a protocol for characterizing colorectal afferent functions in a mouse ex vivo preparation with distal colorectum, pelvic nerve, L5-S1 DRG, and dorsal roots in continuity.

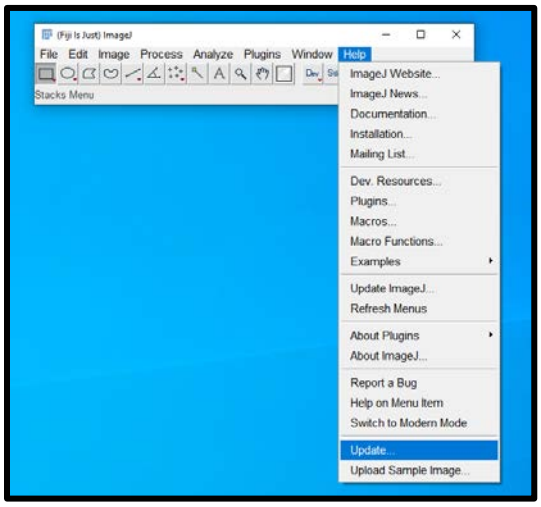
1. Software installation

We use two software for data processing in this protocol: Fiji and MATLAB. Fiji is a distribution of ImageJ that bundles many popular and useful ImageJ plugins for scientific image analysis[2,3]. MATLAB is a programming platform designed specifically for engineers and scientists. It can directly express matrix and array mathematics[4].

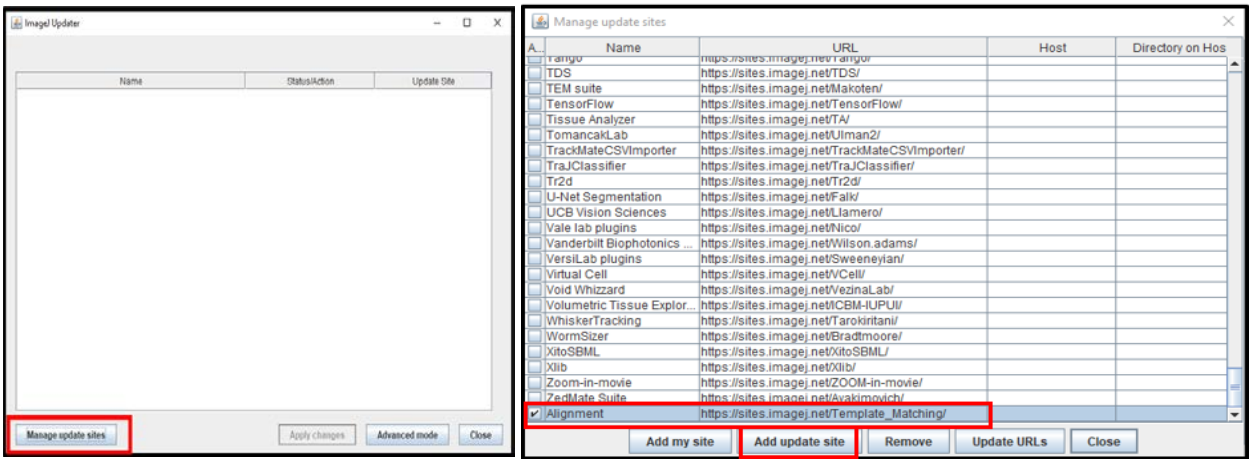
Software version requirements: Fiji (1.44 or later version), and MATLAB (R2016b or later version) with the Image Processing Toolbox. Fiji can be downloaded from the following links: https://fiji.sc/ or https://imagej.nih.gov/ij/download.html. MATLAB and the Image Processing Toolbox can be downloaded and installed following these two links: https://www.mathworks.com/downloads/, and https://www.mathworks.com/products/image.html.

For Fiji user, the 'Template Matching and Slice Alignment' plugin need to be installed using the following steps:

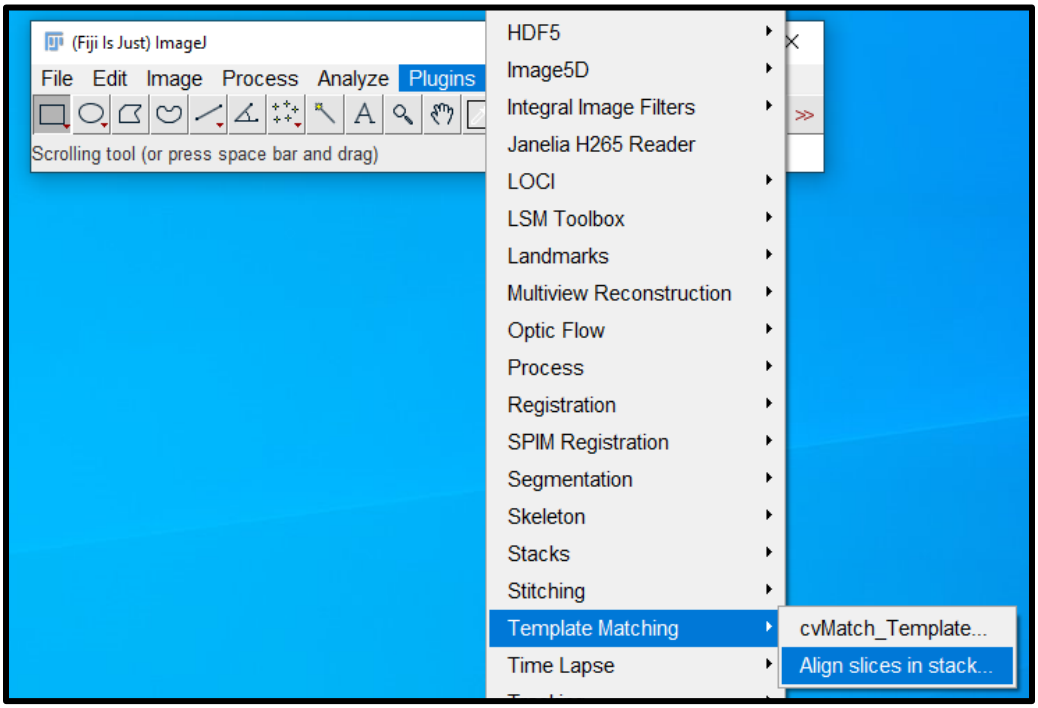
1) Open Fiji, select the built-in update manager (Fiji's Menu > Help > Update...), and update to the latest Fiji version.



2) Add the following URL via ‘Manage update sites’ > ‘Add update site’ (http://sites.imagej.net/Template_Matching/) listed below:



3) After applying the changes and restarting Fiji, the plugin will appear under the ‘Plugins’ menu (Plugins > Template Matching) shown below:

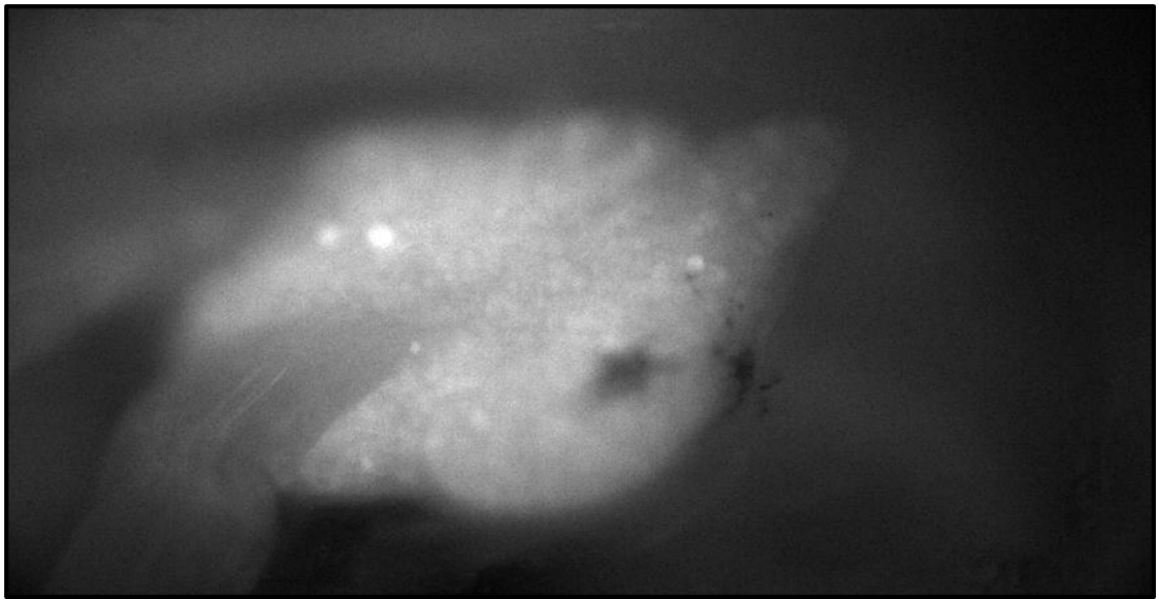


If the users do not want to install Fiji, they can also install plugins using ImageJ following this link: https://sites.google.com/site/qingzongtseng/template-matching-ij-plugin#install

2. Data acquisition

We employed an upright microscope platform (BX51WI, Olympus, Waltham, MA) to capture one whole DRG with a water immersion 10X objective (UMPLFLN 10XW, 0.3 NA) using a regular halogen epi-illumination light source and a regular filter cube for sample illumination. A high-

speed ultra-low noise sCMOS camera (Zyla-4.2P, 82% quantum efficiency, Andor Technology, South Windsor, CT) was used to capture high-resolution images (1920 x 1920, 2 x 2 bin) at 100 frames per second, which provided a spatial resolution of 1.6 pixels/µm sufficient to resolve individual DRG neurons. This system allowed the recording of Ca^{2+} transients to resolve individual action potentials (APs) in multiple GCaMP6f - expressing DRG neurons simultaneously. These high-resolution images were saved as a tiff file. A whole DRG can be recorded with a spatial resolution of the GCaMP6f signals at individual DRG neurons as shown below[1].



3. Data processing

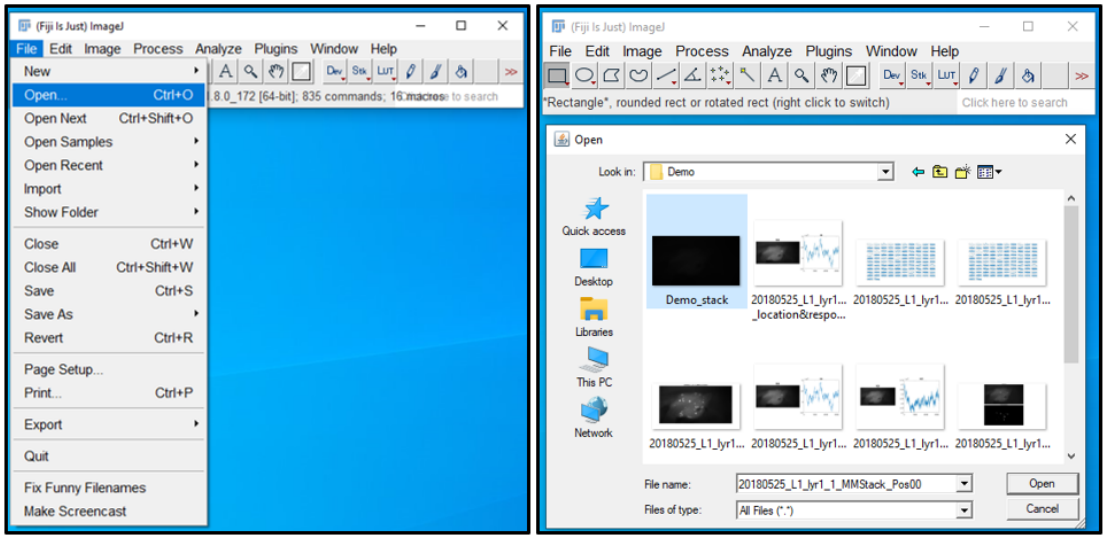
In section 3.1, we perform image alignment using an ImageJ plugin called 'Template Matching and Slice Alignment'. The positional shifts in-between different frames can be corrected in this process. In section 3.2, we perform morphological processing and marker-based watershed segmentation to determine the boundary of the neurons. We then extract the GCaMP6f signals in the form of pixel intensity (0–255) from individual DRG neurons for further data analysis.

3.1. Image alignment using the 'Template Matching and Slice Alignment' plugin

'Align_slices in stack' is one function of the 'Template Matching and Slice Alignment' plugin in Fuji. It utilizes the template matching approach to perform image alignment based on a selected region of interest (ROI). This function tries to find the same ROI in every frame and translate each frame so that the ROI can be at the same position throughout the whole stack[5-7]. This plugin was programmed and shared on Github by Dr. Qingzong Tseng. The detailed description of this plugin can be found on the website: https://sites.google.com/site/qingzongtseng/template-matching-ij-plugin

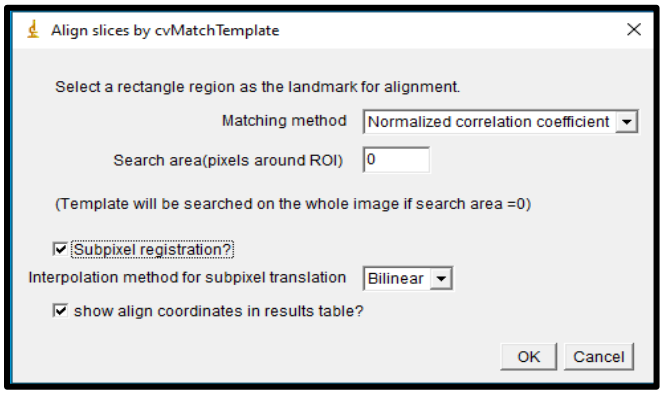
The following steps are used to perform image alignment in this protocol:

1) Open a captured image stack 'Demo_stack.tif' in Fiji (or ImageJ).

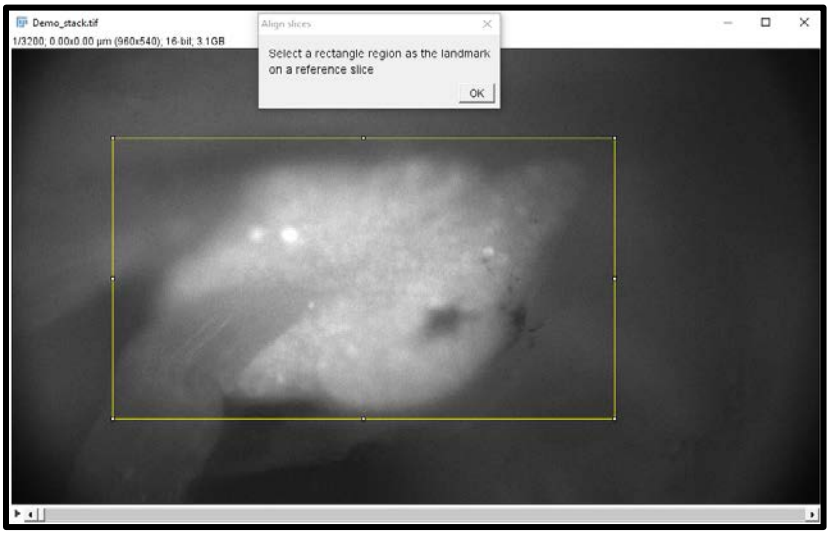


2) Run Plugins > Template Matching > Align slices in stack. A setting window will appear.

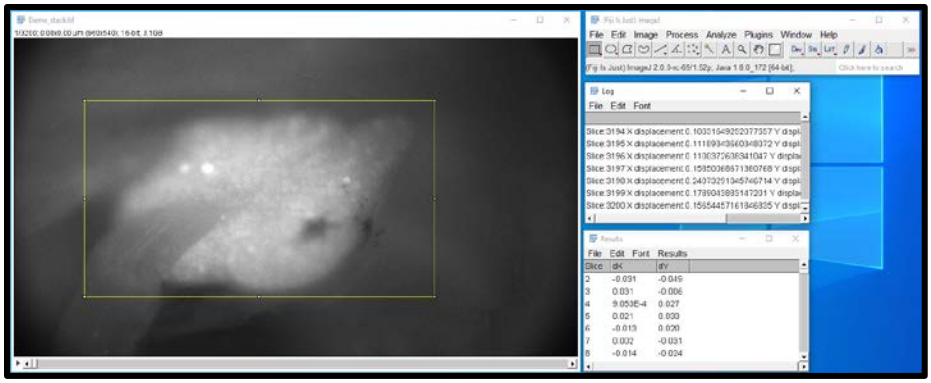
3) We can keep the settings shown below. With the selection of 'Subpixel registration', subpixel image translation can be performed by ImageJ's native translation function with bilinear or bicubic interpolation method. In our experience, both interpolation methods perform well.



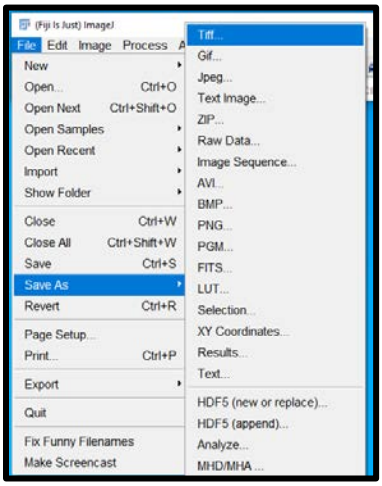
4) Click ‘OK’ and a dialog will appear, asking you to select a square ROI for alignment. After creating the square ROI on the image, click ‘OK’.



5) After the alignment process, all frames in the stack will be aligned and the displacements for each frame are printed out in both the log window and results table.



6) Save the aligned stack as a new tiff file for later uses.

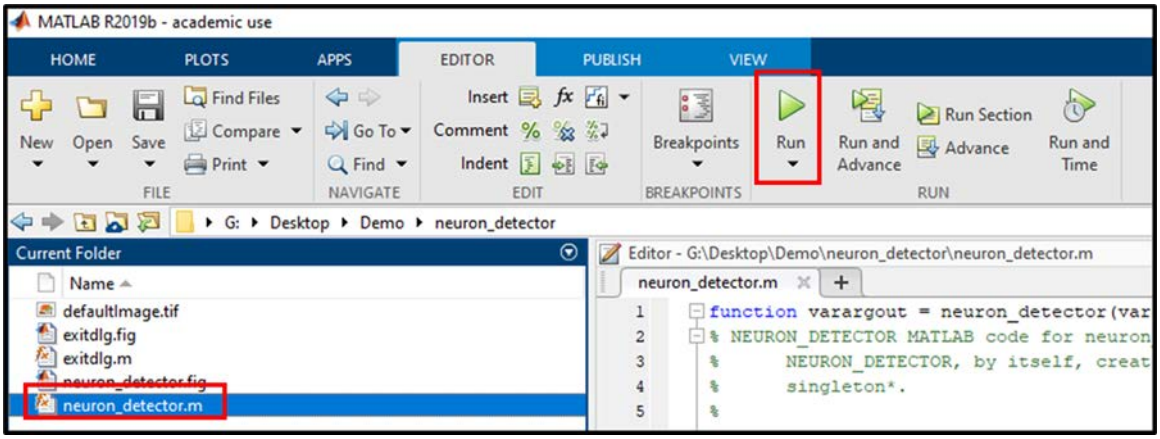


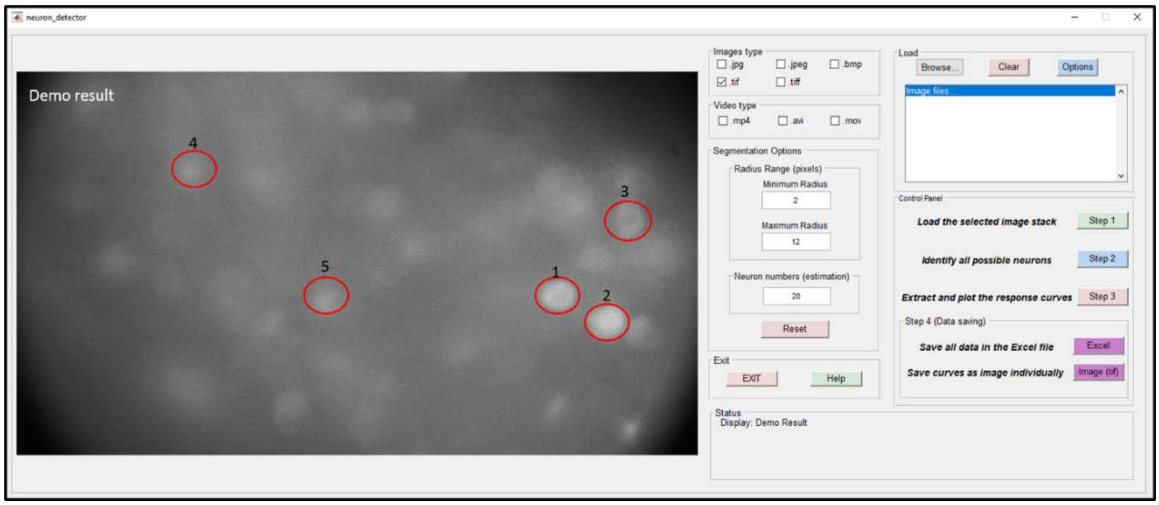
3.2. Extracting GCaMP6f transient signal using a MATLAB GUI

We have developed a MATLAB graphical user interface (GUI) to further extract the GCaMP6f transient signal from aligned image stack. The first step is to load the image stack based on the user's selection. In the second step, the morphological operations are applied to the image stack. Regions with DRG signals can be identified by marker-based watershed segmentation. The third step is to extract all signals in the form of pixel intensity. Even though the result includes some noise signals, it provides the action potentials information about the whole DRG sample under stimulus. The last step is to display and save all response curves based on the generated results.

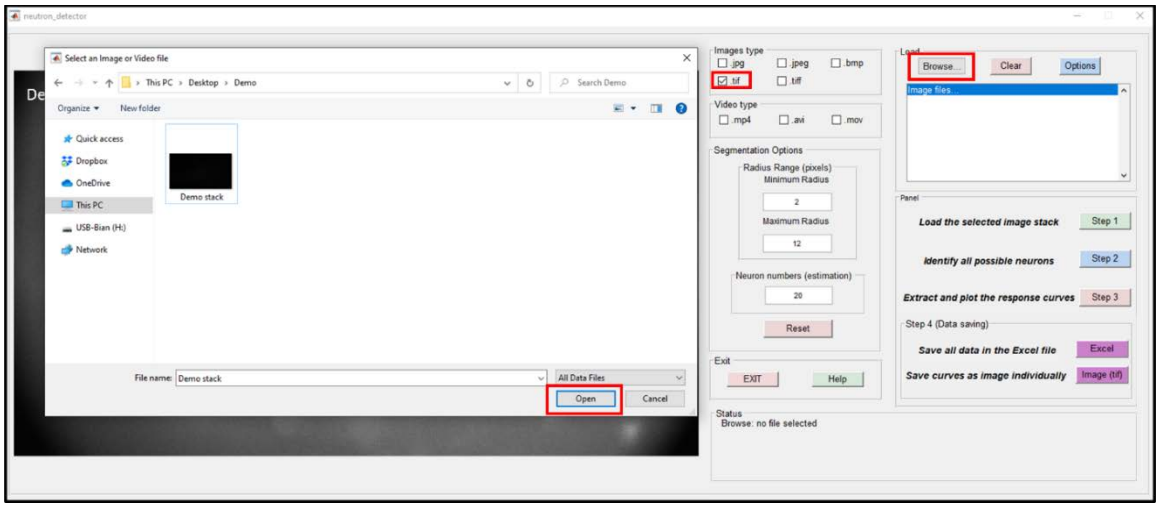
The operations of this MATLAB GUI can be shown as follows:

1) Open the file 'neuron_detector.m' in MATLAB. Click the green 'Run Section' icon. The GUI will appear like below.

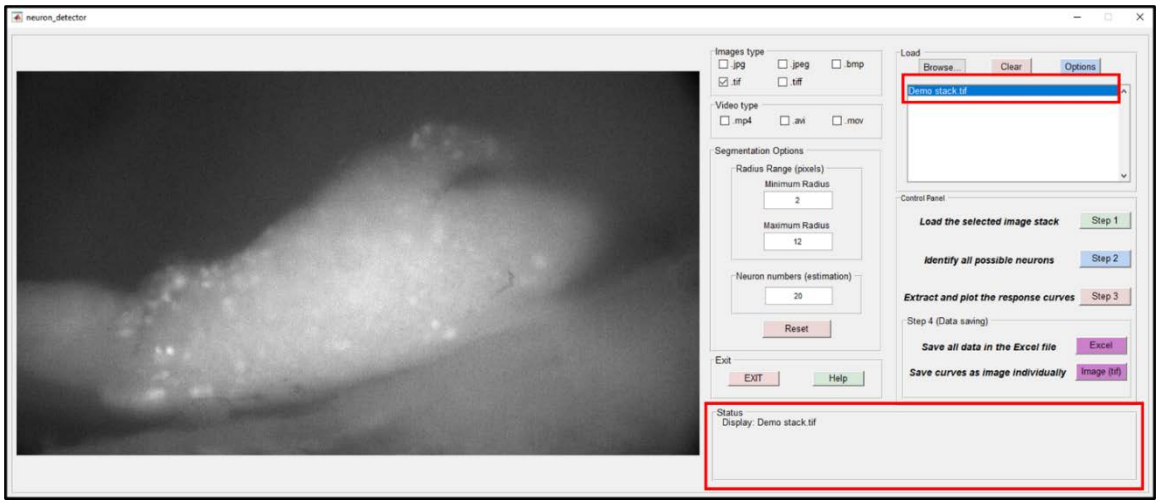




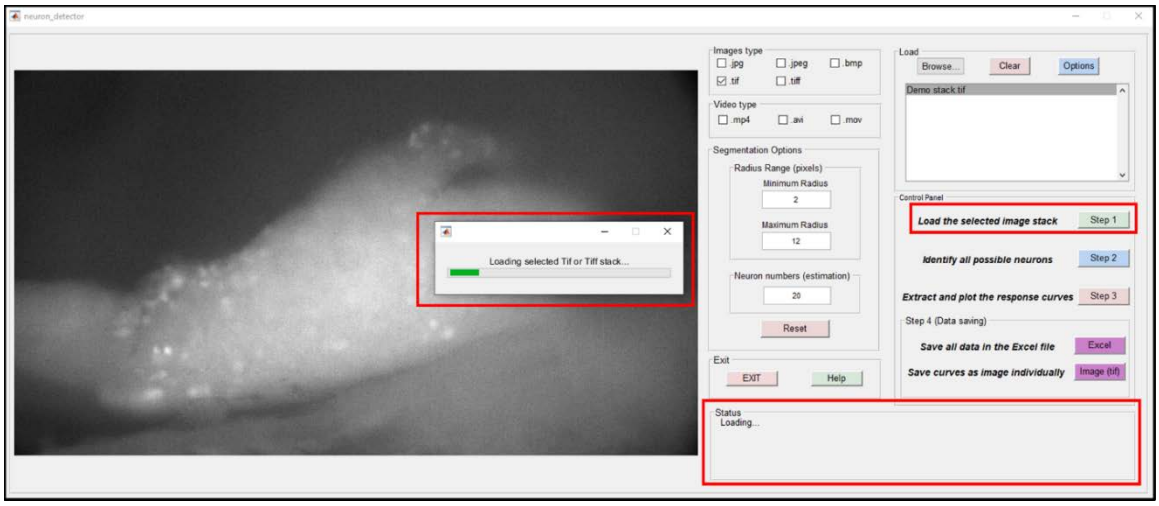
2) After opening the GUI, the first step of operations is to load the correct image stack into the MATLAB workspace. Our GUI is able to recognize multiple image formats and video formats. Here, we use the 'Demo_stack.tif' file as an example. First, select '.tif' checkbox. Second, click the 'Browse…' button, a pop-up window will appear. Select the 'Demo_stack.tif' and click the 'Open' button.



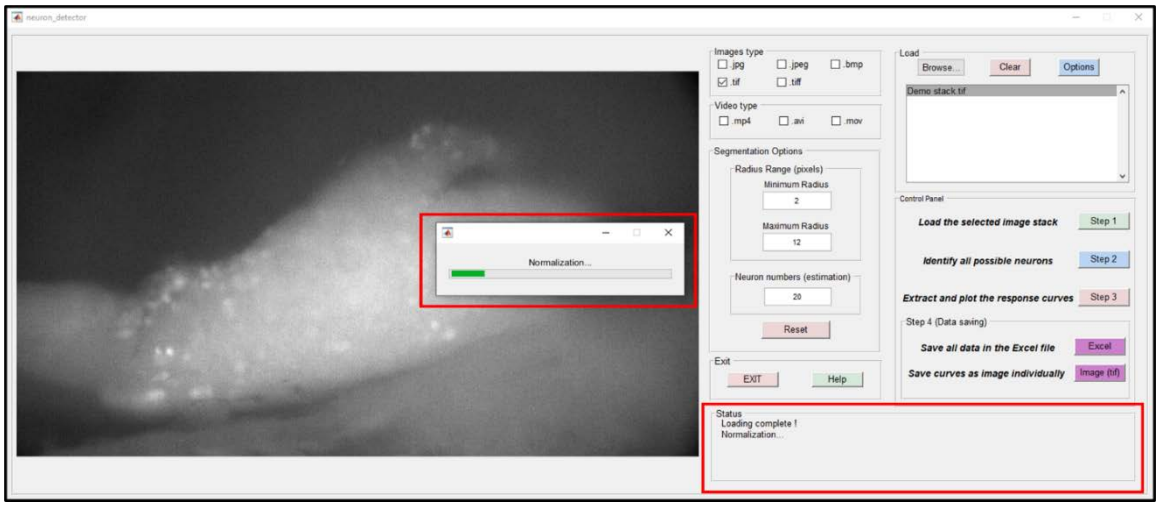
All tif files under the same file path will be listed in the GUI. In this example, the 'Demo_stack.tif' is the only tif file under the selected path. Select and left click the file's name. The first frame of the stack will appear in the GUI and the 'Status' panel will also list the file's name.



After selecting the tif file, click the 'Step 1' button. It will load the stack into the MATLAB workspace. The process bar will appear and the 'Status' plane will change to 'Loading…' as shown below.



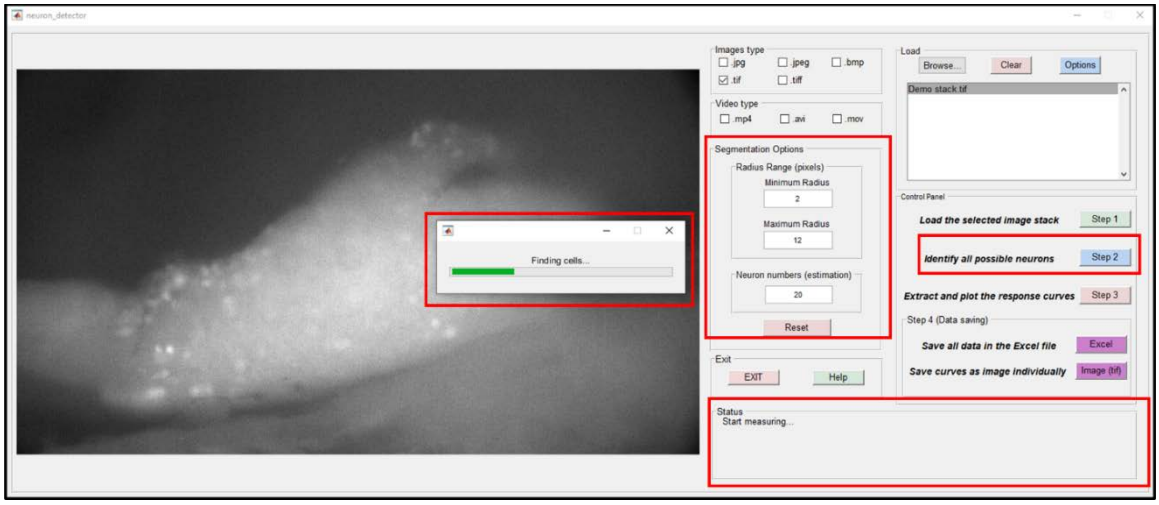
After the ‘Loading’ process, a ‘normalization’ operation will be applied to the image stack. Each frame will be divided by its mean intensity value to remove the background fluctuation. Following the normalization operation, a variation map will be generated to show the intensity variations of the image stack.



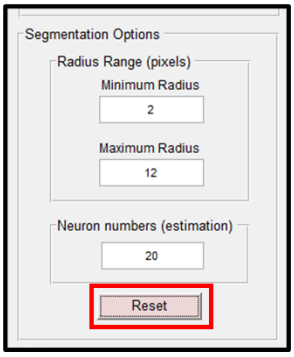
When Step 1 is finished, the ‘Status’ plane will appear as follows:

3) The second step of the operation is to identify all possible active neurons. They can be identified by adjusting the binarized threshold of the variation map. The marker-based watershed segmentation will label the boundary of the neurons. We use the parameters ‘Minimum Radius’ and ‘Maximum Radius’ to represent the estimated minimal and maximal radius of DRG neurons (in pixels). The identified regions outside the radius region will be excluded from the final result. The parameter ‘Neuron numbers’ represents the estimated number of positive neurons in the field

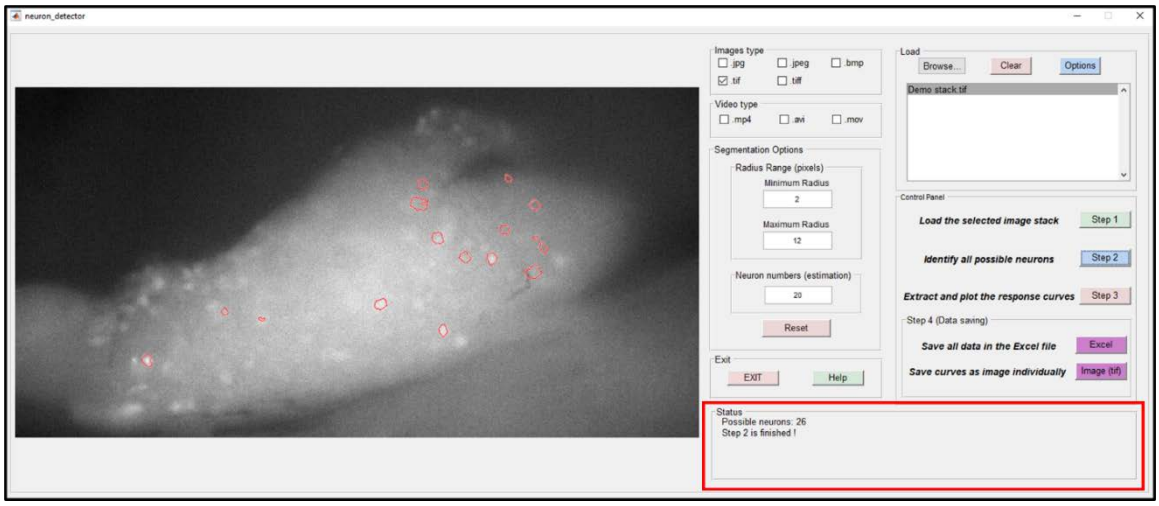
of view. The default setting is based on our microscope setup: Olympus BX51 with a water immersion 10X 0.3 NA objective lens. These three parameters can be adjusted based on the magnification factor of the microscope setup and specimens under testing. Click the 'Step 2' button. The process bar will appear and the 'Status' plane will change to 'Start measuring…', as shown below.



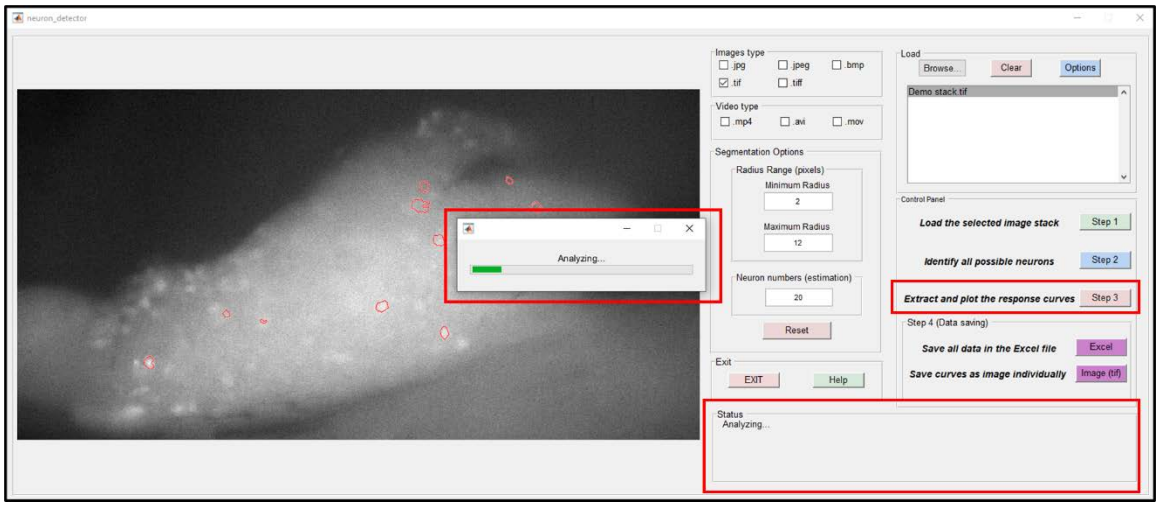
The 'Reset' button can adjust three parameters back to our default settings.



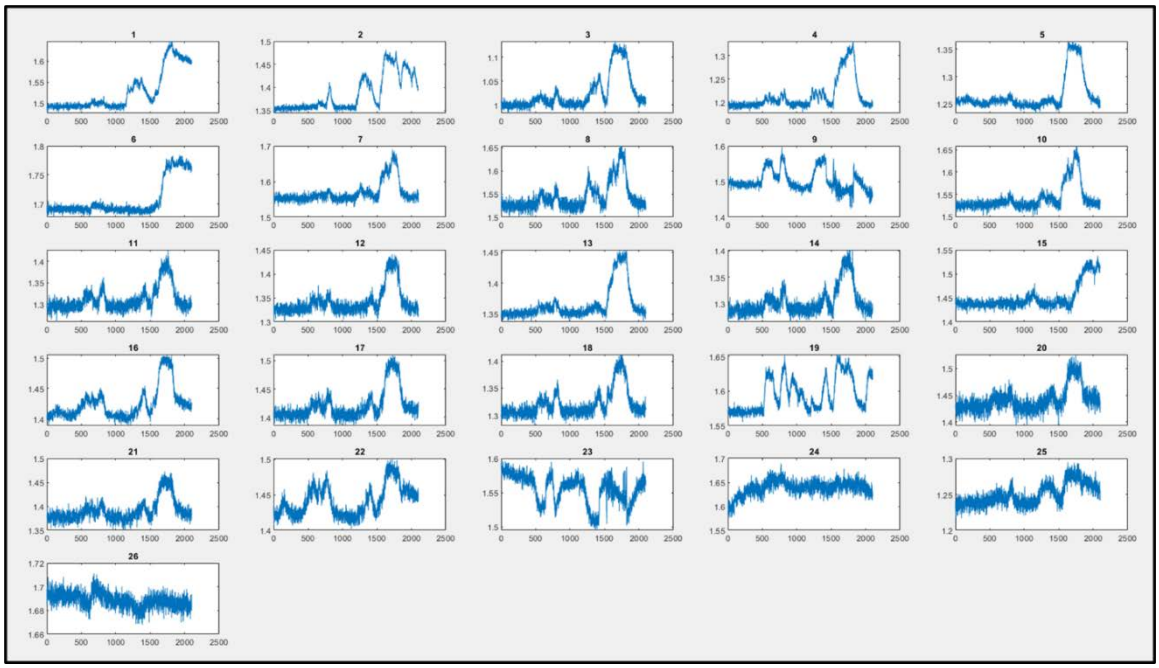
When Step 2 is finished, the image with labeled neurons will show in the GUI. The 'Status' panel will also list the number of possible neurons, and states 'Step 2 is finished!'.



4) Step 3 of the operation is to extract and plot the response curves from the identified neurons. The average normalized pixel intensity of the labeled neuron is calculated to represent its response over time. Click the ‘Step 3’ button, the process bar and the ‘Status’ plane will appear as shown below.

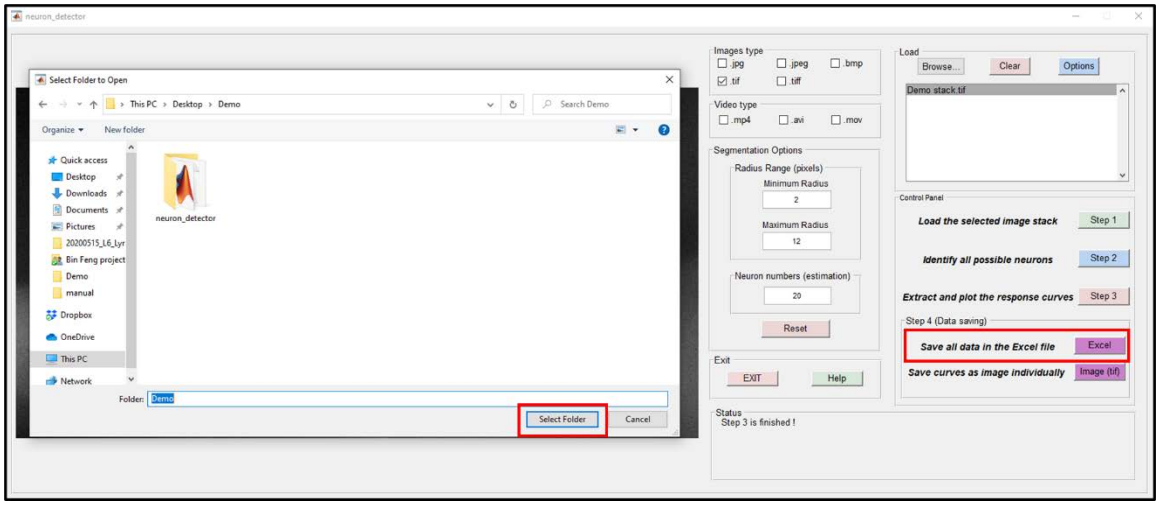


When Step 3 is finished, a full-screen figure with all response curves will appear as the figure shown below. The curves are ranked by their amplitude of variation. Some negative neurons are detected in this process. A better ranking strategy will be included in a later version of the protocol for removing negative responses.

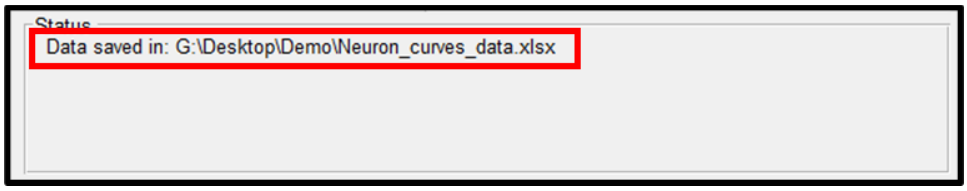


The ‘Status’ plane will change to ‘Step 3 is finished!’, as shown below.

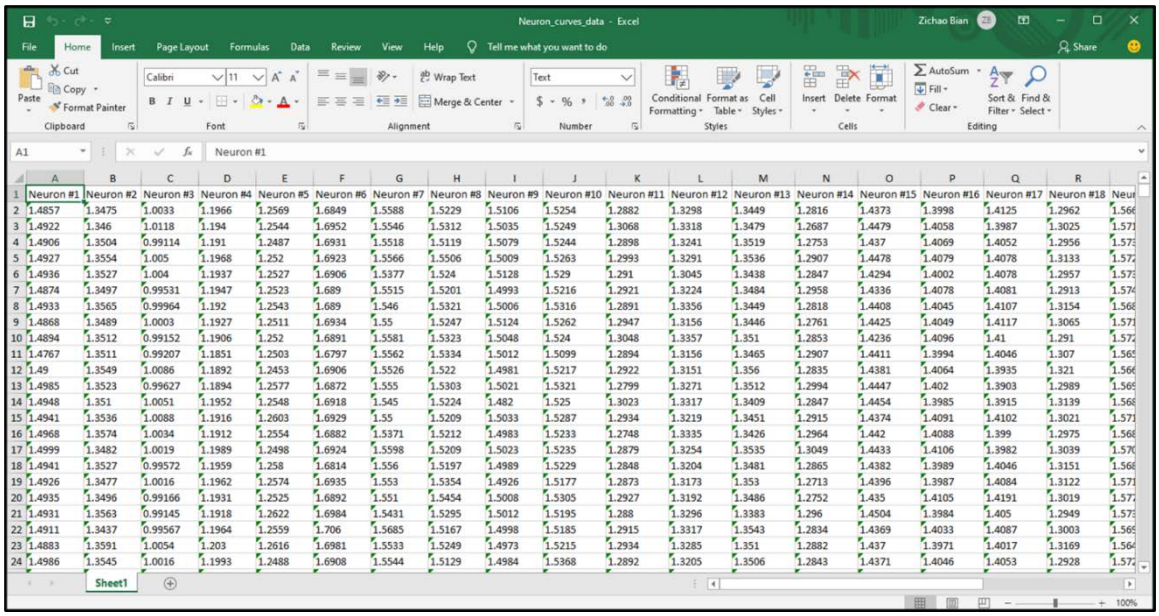
5) Step 4 of the operation is to save the response curve data. There are two options shown below. For the first option, user can save all data in an Excel file: click the ‘Excel’ button -> select the desired saving folder and click the ‘Select Folder’ button.



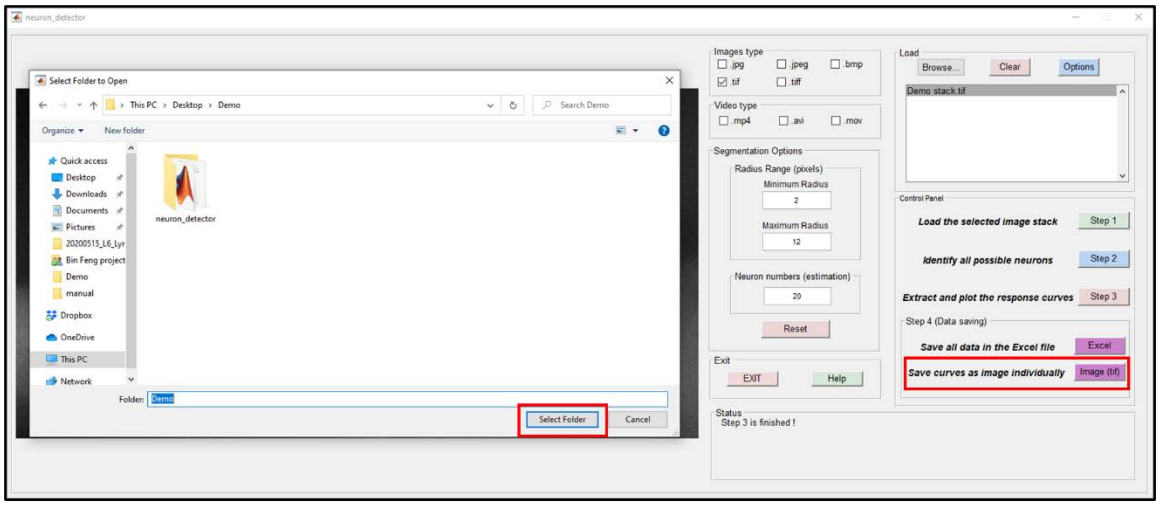
When the saving process is done, the ‘Status’ plane will list the file path as shown below. A figure with all response curves (same as the Step 3 result) will also be saved in the same folder.



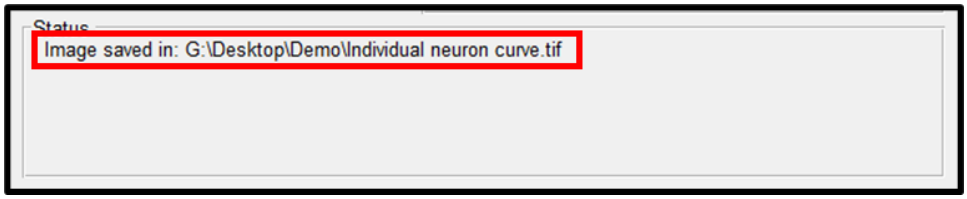
In this example, the ‘Neuron_curves_data .xlsx’ Excel file will contain all neurons’ response curves:



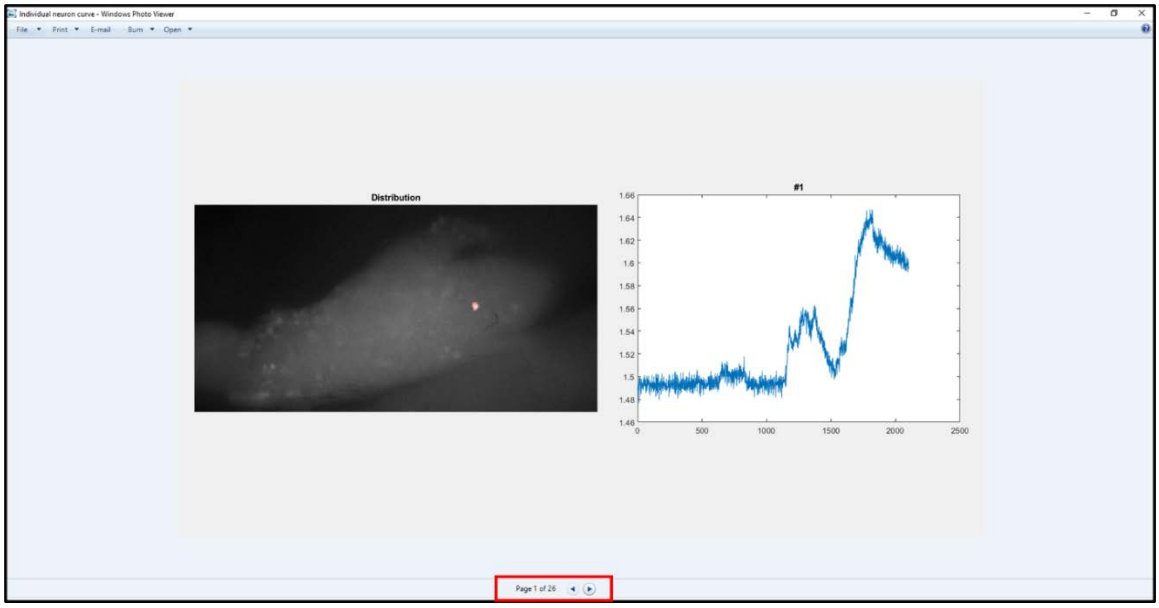
For the second option, user can save every curve with its corresponding neuron’s location as a tif image file. All the tif files will be merged into one tif stack. Click the ‘Image (tif)’ button, the dialog window will appear. Select the desired saving folder and click the ‘Select Folder’ button.



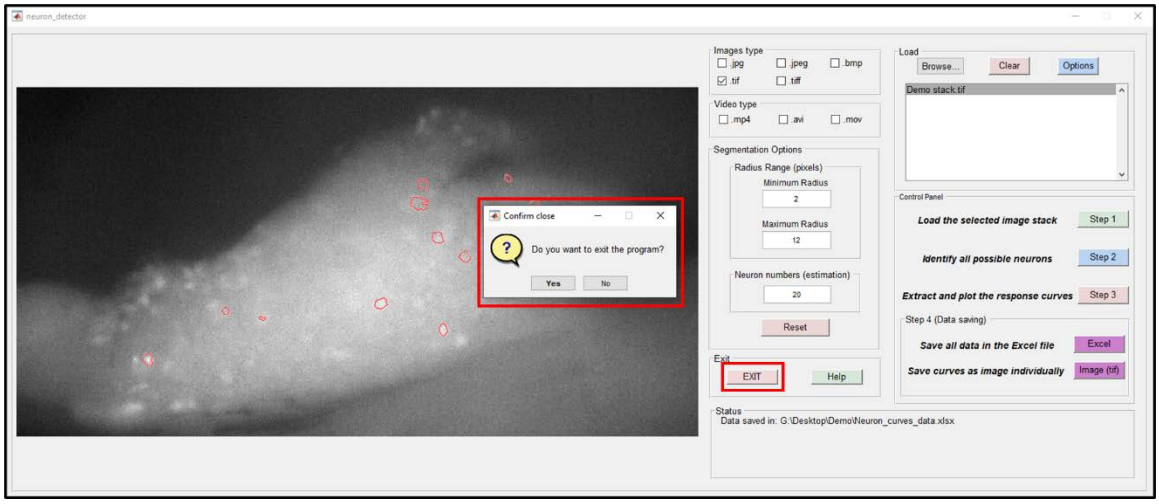
When the saving process is done, the ‘Status’ plane will list the file path as shown below.



The ‘Individual neuron curve.tif’ stack contains all neurons’ curves and the corresponding locations. The frame number of this stack equals the possible neurons’ number as shown below.



6) When all operations are finished, user can click the ‘EXIT’ button and select ‘Yes’ to exit the MATLAB GUI.



Reference:

1. Guo, Tiantian, et al. "Optical recording reveals topological distribution of functionally classified colorectal afferent neurons in intact lumbosacral DRG." *Physiological reports* 7.9 (2019): e14097.

2. Schindelin, Johannes, et al. "Fiji: an open-source platform for biological-image analysis." Nature methods 9.7 (2012): 676-682.

3. Schneider, Caroline A., Wayne S. Rasband, and Kevin W. Eliceiri. "NIH Image to ImageJ: 25 years of image analysis." *Nature methods* 9.7 (2012): 671-675.

4. “What Is MATLAB?” MATLAB & Simulink, 2020, www.mathworks.com/discovery/what-is-matlab.html.

5. Tseng, Qingzong, et al. "A new micropatterning method of soft substrates reveals that different tumorigenic signals can promote or reduce cell contraction levels." *Lab on a chip* 11.13 (2011): 2231-2240.

6. Tseng, Qingzong, et al. "Spatial organization of the extracellular matrix regulates cell–cell junction positioning." *Proceedings of the National Academy of Sciences* 109.5 (2012): 1506-1511.

7. Tseng, Qingzong. "Study of multicellular architecture with controlled microenvironment." *Université Grenoble Alpes* (2011).