How to use WormTracer

(1) Data required

WormTracer only requires a "folder containing binarized worm video images".

Put a binarized worm video in a single folder as individual image files for each frame in a format that can be read by openCV, such as png, bmp, jpeg and tiff. The folder name is arbitrary (you will specify it later in the "detaset\_path" in the code). The image file names should include numbers at the end in a chronological order; you can save the images in this format for example by selecting “save as”>”image sequence” in ImageJ.

(2) File configuration

Prepare the functions.py file and folder of images. Feel free to place them wherever you can specify the paths in the WormTracer code.

(3) Prepare a GPU-enabled environment, such as Google Collaboratory. Although you can run WormTracer on CPU, if the number of images is large, the execution time will be very long. The following python libraries must be installed and made available.

Matplotlib, cv2 (opencv-python), torch, scipy, scikit-image

(4) Parameter setting

Set the variety of parameters in the code that you are going to run, WT17\_5.ipynb or wt17\_5.py. The image folder path (dataset\_path) and image file extension (extension) must be set to match the data files, and functions\_path must be set to the path where functions.py is located. Other parameters can be changed as necessary. Both “start\_T” and “end\_T” should be set to 0 if all images in the data folder are to be used.

(5) Execution and saving the results

Run WT17\_5.ipynb or wt17\_5.py. When the code is executed, two folders containing the results are created in “detaset\_path”; a CSV file with the coordinates of the centerline is saved in "results" folder, and image files depicting the centerline on the worm image are saved in "full\_line\_image" folder. The execution time depends on the size of the images, for example 4200 120x120 px images take about 2 hours on a standard GPU. If “SaveProgress” is set to True, the optimization process can also be saved as image files (in progress\_image folder) in “detaset\_path”.

(5) Check the results and rerun.

If the results are not as expected, try changing the parameters.

If the worm is moving too fast (recorded in low video fps) and the centerline is not keeping up with the moving worm, decrease “continuity\_loss\_weight”. If the centerline is squishy and bent, increase “smoothness\_loss\_weight”.