Dendritic localization of mRNA in Drosophila Mushroom Body Output Neurons

USER MANUAL

The pipeline was designed for the rapid processing of co-localization single-molecule microscopy images. The software requires MatLab, DIPImage, and CUDA, as well as a computer that meets the requirements as outlined in the next section.

SYSTEM REQUIREMENTS

To function properly, the pipeline requires a computer that meets the following specifications:

- A CUDA-enabled graphics card
- Any Intel or AMD x86-64 processor (min. 4 cores recommended)
- 4 GB of RAM or more
- 4-8 GB of space for typical installation

INSTALLATION

All the software packages required for the pipeline support multiple operating systems. As a result, the pipeline should in principle work on Windows, Linux, and macOS. This manual currently includes installation instructions only for Windows. Adventurous users are however encouraged to test the pipeline on their operating system of choice

Windows installation

1. Clone repository or download software from https://github.com/qnano/smFISHlearning
2. Install MatLab using the installation materials and procedures provided by your institution or acquired with the MatLab license.
3. Install DIPimage for MatLab
    a. Download the latest version from https://diplib.org/download_2.9.html#older-releases. To maximize stability, make sure that the version you download has been tested with the MatLab version you use.
    b. Install DIPimage for MatLab (refer to DIPimage User Manual for assistance)
    c. Add the following lines to the MatLab startup file startup.m (usually found in Documents\MATLAB; create file if it does not exist yet):
    run('C:\Program Files\DIPimage\dip_initialize');

Note: if DIPimage was not installed in the default location, the line above needs to be changed. In some DIPimage distributions, the DIPimage folder in Program Files is given a name that includes the version number (e.g. DIPimage 2.9.1).

d. Confirm that DIPimage was installed successfully by executing the command dip image in MatLab. Refer to the *DIPimage User Manual* for assistance with any unresolved errors.

4. Install CUDA

    a. Download the latest version of CUDA from https://developer.nvidia.com/cuda-toolkit

    b. Run the installation as administrator and

        i. Choose *standard* or *express* installation;

        ii. Acknowledge the warnings regarding Visual Studio (not needed to run the image processing pipeline).

    c. Restart the computer

5. Adjust GPU time-out settings in the following way:

    a. Open the application NVIDIA Nsight Monitor

    b. Select *Nsight Monitor options*

    c. Change WDDM TDR Delay → 180 seconds (default:2)

    d. Change WDDM TDR Enabled → FALSE (default: TRUE)

Example data

1. Download the example data from https://figshare.com/articles/dataset/Example_data/13568438 containing:
    a. 'example_set_dendrites.r3d': Four-channel stack of images. The first channel corresponds to the smFISH data and the third channel with the dendrites.
    b. 'example_set_calyx.tif': Single channel stack of images of the calyx.
    c. 'example_set_transcription_focus.tif': Four-channel stack of images. The first channel corresponds to the smFISH data and the third channel with the soma. Around slice z=36, in both somas, the transcription focus can be observed as the relatively bright spot.
2. Create a folder called 'data' in the parent 'smFISHlearning' folder.
3. Put the three downloaded files into the 'data' folder.

DATA PROCESSING AND ANALYSIS

How you can run the code is described below. A more extensive description of processes (detection, segmentation) is provided in the supplementary information. Added comments with/in the different function m-files will further help a user to find his/her way in the code.

Dendritic data analysis

1. Sample data set:' example_set_dendrites.r3d'
2. Fill in the user inputs and run 'detection3D_dendrites.m'
3. Wait until detection is finished.
4. The segmentation of the dendrites must be performed now:
    a. A dip image will pop up, and the command window will tell you to pick the left or right part of the dendrite.
    b. Set the mapping of the figure to the one which applies to your data set.
    c. Guide through the image (by using the keyboard buttons 'n' and 'p') to find the slice where the dendrite covers the largest area
    d. Manually select and close this contour. Then right-click to create a mask.
5. Repeat this for the other side of the dendrite.
6. The properties of the detections within the dendrites can be found in the struct 'results'

Calyx data analysis

1. Sample data set: 'example_set_calyx.tif'
2. Fill in the user inputs and run 'detection3D_calyx.m'
3. The segmentation of the calyx must be performed now:
    a. A dip image will pop up, and the command window will tell you to select the calyx (the 'dark region' in the image).
    b. Set the mapping of the figure to the one which applies to your data set.
    c. Manually select and close this contour. Then right-click to create a mask
4. Repeat this 2 times.
5. The properties (brightness, background, width, and position) of the detections within the calyx can be found in the struct 'results'

Soma and transcription focus data analysis

1. Sample data set: 'example_set_transcription_focus.tif'
2. Fill in the user inputs and run 'transcription_analysis/main.m'
3. Wait until detection is finished.
4. The segmentation of the soma must be performed now.
    a. A dip image will pop up, and the command window will tell you to pick the left or right soma.
    b. Set the mapping of the figure to the one which applies to your data set.
    c. Guide through the image (by using the keyboard buttons 'n' and 'p') to find the slice where the soma covers the largest area
    d. Manually select and close this contour. Then right-click to create a mask.
5. Repeat this for the other soma.

6. Wait for fitting the width of the transcription focus (the highest intensity detection).
7. The properties (brightness, background, width, and position) of the transcription foci can be found in the struct 'trans_site' and of the other spots instruct 'dif_lim'

Supplementary information

Detection & analysis of mRNA molecules

All image analysis is based on the software of (Smith et al., 2010) and (Smith et al., 2015), which can be found at https://github.com/carlassmith/exampleGLRT. The script *Example3D.m* forms the framework of the detection and width and brightness analysis.

The mRNA smFISH channel is extracted and stored as a 3D grayscale image. The software of (Smith et al., 2015) is used to perform the detection of the mRNA molecules. First, the generalized likelihood ratio test (GLRT) based detection on the smFISH data is executed by using the *LLRMap3D.m* function. The inputs of the *LLRMap3D* function are σ, the minimum pixel size of a detection cluster equal to zero, and the other parameters as default. Thereafter, the pre-filter fit parameters are set to default, except the probability of false detections is set to 0.05.

After the detection, the intensity, background, width, and subpixel position of the detected mRNA spots are estimated. This is done using a maximum likelihood estimation (MLE), as proposed by (Smith et al., 2010). For each detection, a region of interest (ROI) is defined, as a 2D box in the x-y plane with a size of $2\times3\sigma+1$, with the center at the rounded pixel coordinates of the detection. For the detection within the ROI, the MLE is performed on the Gaussian PSF to estimate the x and y position, the number of photons *I*, the number of background photons *bg,* and the width of the 2D Gaussian, σ_x and σ_y. This is done by executing *fitBoxCenters.m*, where the parallel GPU implementation *gpuGaussMLEv3.m* of the MLE from (Smith et al., 2010) is called. If the estimated center of the detection falls outside the fitting box (ROI), the detection is removed. Otherwise, the detections are not filtered based on the outcome of the MLE. This implies the *paramsFilterFits* is set to default, except *MaxPFAValue*, which is set to 1.

Segmentation dendrites

The dendritic membrane is labeled and can be segmented based on the intensity level. First, the maximum contour of each dendrite is selected. This is done by setting the image intensity to a logarithmic scale and then search for the z-slice in which the dendrite covers the largest area. The slice with the maximum contour of the dendrite is manually selected, using *roipoly*. All the intensity values within this contour, for each z-slice, are being used for further analysis. This is necessary to remove (high intensity) image artefacts from the analysis. Secondly, the edges of the image are enhanced by using a difference of Gaussians (DoG) filter, where the Gaussians have a width of 1 and 5 pixels. Finally, the enhanced image *A* is thresholded to obtain the mask *M* as

$$M = A > mean(A) + std(A), \quad (1)$$

where *mean*() and *std*() are the operators for the sample mean and sample standard deviation of the image intensity values.

Segmentation calyx

The calyx is segmented slightly differently than the dendrites. For the image analysis of the mRNA inside the Calyx, only the smFISH channel is used. In Fig. 1 a 'dark hole' can be seen, surrounded by densely packed dendrites. The mRNA of interest are the ones that are inside the calyx ('dark hole') and not the ones in the surrounding dendrites.

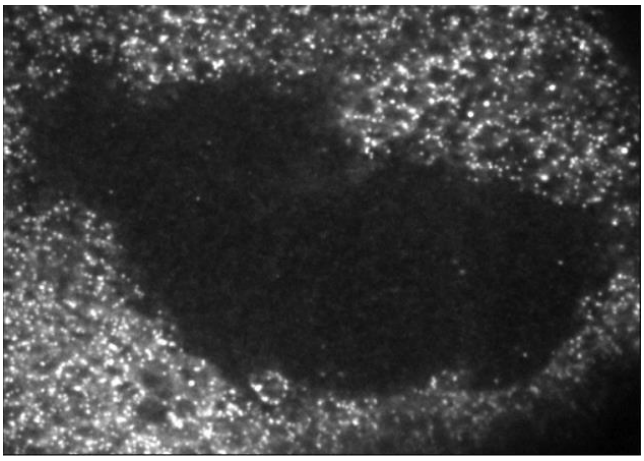

Figure 1: Example z-slice of the calyx, where the middle dark region represents the calyx.

The segmentation is done manually, without any thresholding. It is observed that the shape of the calyx barely changes in three z-slices, resulting in the following method. First, the center slice (in z) of the calyx is selected, suppose this is z=25. Then the shape of the calyx is selected using *roipoly*. This shape is extruded in both directions for three slices, such that the total shape covers 7 slices (z=22-28). This is repeated for the slice, which is 7 slices below the center slice (z=15-21), and for the one which is 7 slices above the center slice (z=29-35). This results in a final mask of 21 z-slices.

Segmentation soma

The whole volume of the soma cannot be segmented based on the intensity, but only the edge of the volume, which requires a different method of segmenting. To segment the soma, the first step is enhancing the edges by using a DoG filter, where the Gaussians have a width of 1 and 5 pixels. Secondly, the maximum contour for each soma is manually selected, by searching through the z-slices, which gives the extrema in the x and y-direction. Third, the soma is cropped manually in the z-direction. At this point, image A only contains the extrema of the soma in x, y, and z-direction. Then every z-plane is thresholded according to eq. 1, giving a binary image of the membrane. The mask for the whole nucleus is then created by selecting the convex hull of the binary image (in every z-plane), using *convhull*. Finally, the convex set is selected to be the mask of the nucleus.

Transcription focus analysis

from all the detections found, the detection with the highest intensity is selected as the transcription site. It is important to realize that the parameters of the detections/Gaussian blobs are estimated within a box with a size of 2*3σ+1. Considering that at the transcription foci multiple molecules are present, the initial σ used to detect the single mRNA molecules might not be a good estimate of σ for the transcription foci. So, a different initial σ must be chosen, s.t. the box size is appropriate to estimate the parameters of the transcription site. The authors of (Smith et al., 2010) have validated that the accuracy of the position estimation, for the small deviations on 2*3σ +1 in the box size, is

not deteriorating heavily. Therefore, the found position can be used to analyze the transcription site further. Using the found and validated position, a 2D Gaussian is fitted on the spot corresponding to the transcription site. This is done by using the MATLAB 2019b non-linear least-squares routine *lsqcurvefit*, using the trust region-reflective algorithm, which gives an estimation of the σ of the transcription focus. The final parameters of the transcription site are obtained by using the same detection and MLE procedure as for the other spots, but with estimated σ, s.t. the box size is appropriate. This makes a comparison of the single spots and the transcription site feasible.

Bibliography

[1] Smith, C. S., Joseph, N., Rieger, B., & Lidke, K. A. (2010). Fast, single-molecule localization that achieves theoretically minimum uncertainty. *Nature methods*, *7*(5), 373-375

[2] Smith, C. S., Stallinga, S., Lidke, K. A., Rieger, B., & Grunwald, D. (2015). Probability-based particle detection that enables threshold-free and robust in vivo single-molecule tracking. *Molecular biology of the cell*, *26*(22), 4057-4062.