jCA software

INTRODUCTION

jCA software (**Fig.1**) is designed for the initial processing of fluorescent microscopy images. It defines the cell borders, tracks cells along the image series, and calculates fluorescence dynamics of each individual cell. The software uses a TIFF-file with multiple frames as an input and produces a text file with time-series as an output. It allows for a variety of adjustments resulting in better quality image processing. jCA mainly uses java classes implemented in ImageJ.

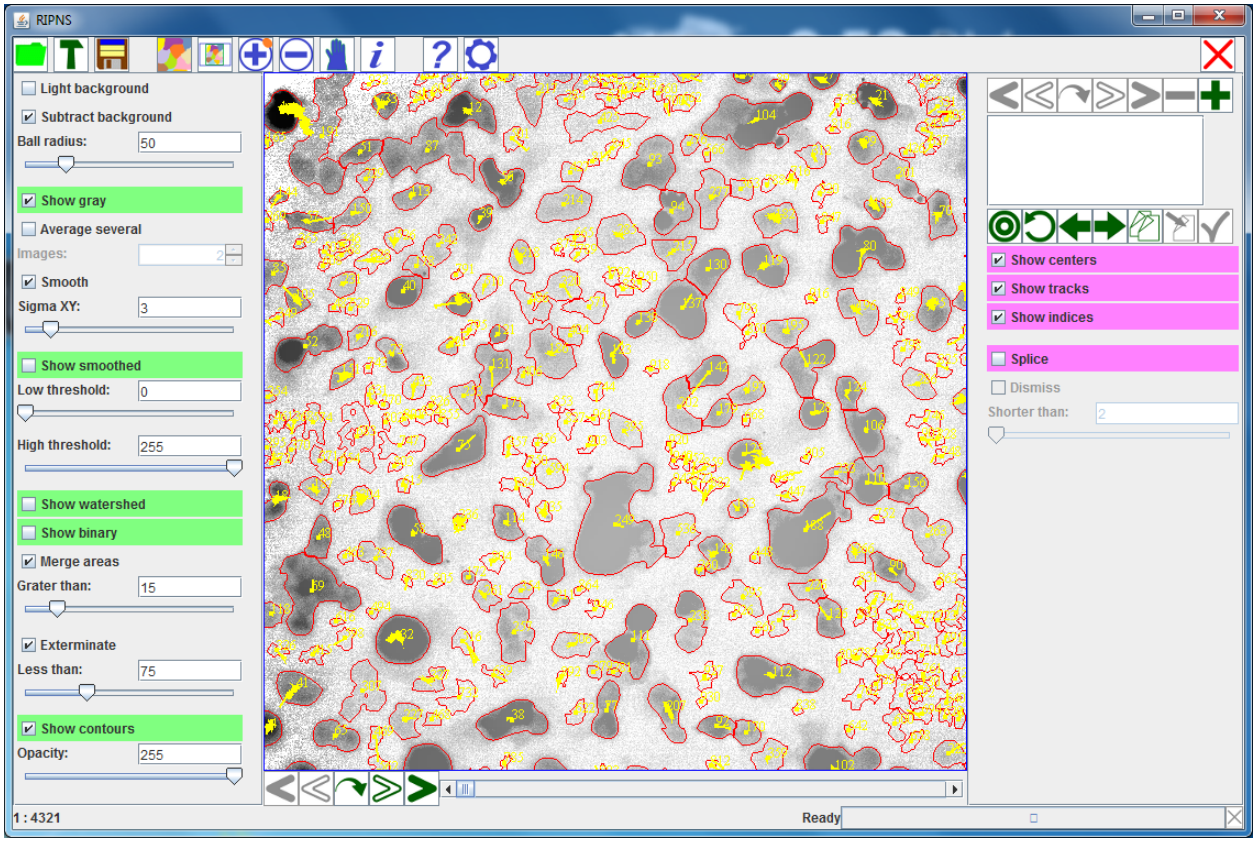


Fig.1. Overview of the jCA software.

HARDWARE and SOFTWARE REQUIREMENTS

1. MacOS or Windows computer
2. 8 Gb memory (recommended 16 or more)
3. Java virtual machine (download and install from https://java.com/en/download/)

INSTALLATION GUIDE

1. Download jCA.zip file from https://github.com/tonza17/jCA to the folder of choice
2. Unpack the jCA.zip file in the same directory
3. Click on jCA.jar file. If this doesn't work:

Windows:

4. Open Terminal
5. Go to the folder with installed jCA
6. Type "java jCA" in the command line

MacOS:

4. Right click on the jCA.jar and chose "Open With->Jar Launcher (default)"

GETTING STARTED

1. Click the  button to open your imaging data. Data can be one- or two-colors. If you have two colors, the segmentation will be performed on the first color, while fluorescence intensity will be calculated from the second color.
2. Select the 'show contours' check box to see the contours.
3. To achieve better cell segmentation you can adjust the following: background subtraction, smoothing, thresholding and other parameters (see below for a detailed description of available options). The buttons on the bottom allow you to navigate the data from frame to frame or to jump to a specific frame number.
4. Click the  button to start segmentation on the entire movie. This may take some time, depending on the duration of the movie. If you want to increase the amount of memory used by the Java Runtime Environment, you should use (and modify) the provided bat file.
5. After the segmentation process is complete you can view the results using a set of checkboxes on the right-hand side. You can see the ROI number, its trajectory, and the center of mass. You will notice that the index of a given cell may change from frame to frame. These can be spliced later using the 'Splice' checkbox.

6. To view responses from individual cells select the  button and then click on the cell of interest. The fluorescence trace will appear in a separate window along with traces showing the overall distance moved by the cell and area. If the area shows sudden drops or rises, it means that the algorithm segmented only part of the cell. All three traces can be saved in a separate text file and analyzed using other software (e.g. IgorPro or Matlab).

7. To save the entire dataset, click the  button. In this case, the data is saved as a tree and you will need to splice it yourself using any programming language. We provide a set of Igor Pro functions that can be used for this purpose.

POSSIBLE ADJUSTMENTS

1. Check the 'Light background' checkbox if you have dark cells on a light background and uncheck otherwise.

2. Check 'Subtract background' and adjust 'Ball radius' to subtract the background using the rolling ball algorithm.

3. Check the 'Show gray' checkbox to see the grayscale picture after subtracting the background.

4. Check the 'Average several' checkbox and choose the number of images you want to be used while averaging if you want to define your cells on an averaged image.

5. Check the 'Smooth' checkbox and choose 'Sigma XY' to use the Gaussian filter.

6. Check the 'Show smoothed' checkbox to inspect the smoothed picture visually.

7. Adjust 'Low threshold' and 'High threshold' to get rid of small artificial fragments.

8. Check the 'Show watershed' checkbox to see the result of the watershed algorithm.

9. Check the 'Show binary' checkbox if you want to see the binarization result.

10. Check the ‘Merge areas’ checkbox and choose the ‘Greater than’ value if you want to merge areas which have common boundary (in percentages of the whole boundary length).

11. Check the ‘Exterminate‘ checkbox and choose the ‘Less than‘ value to exterminate small artificial fragments.

12. Check the ‘Show contours’ checkbox if you want to see the contours created by the algorithm.

13. Adjust ‘Opacity’ if needed.

14. On the right side of the main window, there is a small organizer allowing you to use different adjustments (made on the left side) for different frames. This feature interpolates values between every two consequent savings you make.

15. Check the ‘Splice’ checkbox to connect track fragments into long tracks.

16. Check the ‘Dismiss’ checkbox and choose the ‘Shorter than’ value to exterminate too short tracks.

Output format

A CSV-file containing the tracks. The file has 7 (for non-spliced tracks) or 8 (for spliced tracks) columns.

n – number (for spliced tracks only)

t – time (in frames)

i – index

p – parent index

a – area

v – value (luminance)

x – x-coordinate of the cell’s center

y – y-coordinate of the cell’s center

EXPORTING THE RESULTING DATA TO OTHER PROGRAMS

There are two ways to export the result of jCA analysis to other programs. The first way exports data from individual cells. The other way performs bulk-export of all the data and requires additional processing in Igor Pro, Matlab, Python or an alternative scripting language.

I. Export of individual traces

1. Click on the info button  .
2. Click on the cell of interest.
3. Window with the average fluorescence trace, area, and total distance from the start will appear (see below for more details).
4. Click on the save button  . All information will be saved in ASCII format.

5. Import individual traces to the statistical program of interest. To import the data in excel select File->Import and tick the "Text file" radio box. To import the data in Igor Pro select "Data->Load waves->Load delimited text".

II. Bulk-export of the data

1. Select the "splice" checkbox on the right side.
2. Press the "Save" button in the top left corner  . The data is saved in ASCII format.

SUBSEQUENT DATA EXTRACTION AND ANALYSIS USING IGO PRO

The following section describes how the data can be analysed using Igor Pro. Similar functions can be developed for other programming languages (e.g. Matlab or Python). All Igor Pro functions required for the following procedures can be found in AnalyseCellDynamics.ipf file. Chose File->OpenFile->Procedure from the drop-down menu to load the functions.

1. Import the data as described above.
2. Load AnalyseCalciumDynamics procedure file.
3. Call AnalyseCalciumDynamics(numtraces, numframes, minlength, framesPerSecond) in the command line. The resulting procedure will perform all calculations and store the data in 3D wave. Parameters: numtraces – number of found cells in the movie, numframes – number of frames in the original movie, minlength – select traces that are bigger than this value, framesPerSecond – number of frames per second in the original movie.
4. Call plotTrace (traceNumber) function to draw the fluorescent dynamics of a given cell. For example, if you want to plot the fluorescent dynamics of the 0^{th} cell, type plotTrace(0) in the command line.
5. Call plotPowerSpectrum (traceNumber) function to draw the power spectrum of the fluorescence dynamics of a given cell. NB! **First datapoint of the power spectrum was made Not A Number for presentation purpose.**
6. Call plotWignerTransform (traceNumber, gaus, cutoffvalue) function to draw the power spectrum of the fluorescence dynamics of a given cell. Parameters: gaus – Gaussian width of the Wigner Transform; cutoffvalue – do not show values above cutoffvalue (for presentation purpose). The latter parameter may also be manually changed if you right-click on the wigner transform image and chose "Modify Image Appearance".
7. Chose Data->SaveWave->Save Delimited Text to export data in ASCII format to be used in other programs.

NB! Igor Pro v.6 (or less) has memory limit of 2 GB. Do not use datasets which exceed this to avoid memory overload. If your data file is too large, redo the analysis on the subset of the movie. To do this, open the original movie in ImageJ, select subset of data and select Image->Duplicate from the drop-down menu.

DETAILED EXPLANATION OF EACH FUNCRTIONAL BUTTON

Top pad:

 Opens imaging data in tiff format

 Starts segmentation

 Saves the result of experimental analysis (see above for the data format)

 Zoom out with maximal vertical stretch

 Zoom out with maximal horizontal stretch

 Zoom in, zoom out, and move image

 Information about cell dynamics. If this button is clicked (yellow spot is visible on the bottom right corner of the button), then clicking on individual cell leads to appearance of the following window:

Top trace shows the dynamics of the calculated cell area. Sudden changes in area mean that the cell was artificially united with another one or split between two. The middle trace shows the dynamic of fluorescence. Bottom trace shows the distance (in pixels) the cell has moved from the location at the

time 0. To save this data (both images and the numerical data in ASCII format), press the bottom left button.

In some cases, individual cells are split into two regions of interest (e.g. due to cell division). In this case the ROIs are assigned new numbers. The numbers in square brackets on top show which individual ROIs were used to draw this trace. In order to use different sets of ROIs, use navigation panel in the top right corner. Note that most of the splits into new ROIs are not due to cell division but noise leading to an artificial split. If you are interested in getting more detailed information about a cell that underwent mitosis, then it is worth doing this manually. We will work on improving the segmentation algorithms in future versions of the jCA in order to decrease the number of artificial splits.

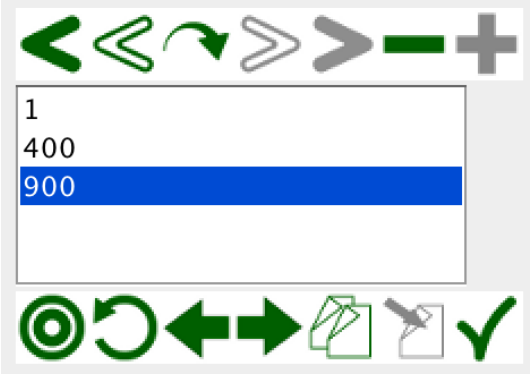
 Shows area (region of interest) with a certain index

 Settings of the program (mainly font colour)

Bottom pad:

 Moving to the: first frame, previous frame, specific frame, next frame and the last frame respectively

Top right pad:



This pad allows for specific conditions to be assigned to and interpolated between individual frames.

 Add and remove a condition. Pressing this button will lead the software to set the condition for this particular frame. When several conditions are saved for different frames, the software then interpolates the condition between the frames. For example, if the threshold is 255 for frame 1 and 155 for frame 100, then the software will automatically adjust the threshold from frame 1 to 155 (frame 2 will have threshold 254, frame 3 – 253 etc.).

 **These buttons allow navigation between frames in which specific conditions have been set.**

 **Copy, paste, and commit to the selected set of conditions.**

 **Left: set default conditions for this frame. Right: set the previously chosen condition for this frame (reverse the change).**

DETAILS, THE SOURCE CODES AND LICENSES OF THE PROCESSING ALGORITHMS

Stage 1: Background removal

The rolling ball algorithm (Sternberg, 1983) is used for background subtraction (**Fig.2**).

Java source code: http://imagej.nih.gov/ij/plugins/download/Rolling_Ball_Background.java

More details on the implementation: http://imagej.nih.gov/ij/plugins/rolling-ball.html

License: ImageJ and its Java source code are freely available and in the public domain. No license is required. https://imagej.nih.gov/ij/features.html

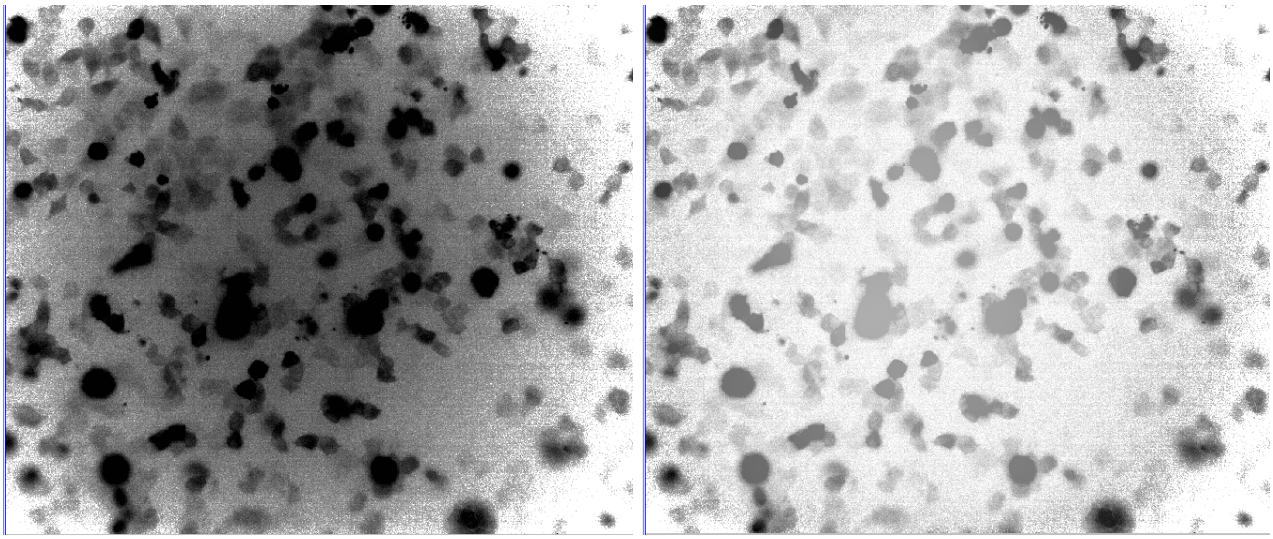


Fig.2. Fluorescent image of cells before and after background subtraction using rolling ball.

Stage 2: Smoothing.

Standard Gaussian blur filter is used for image smoothing.

Java source code:
https://github.com/fiji/imageware/blob/master/src/main/java/imageware/ByteProcess.java

License: BIG License http://fiji.sc/BIG_License

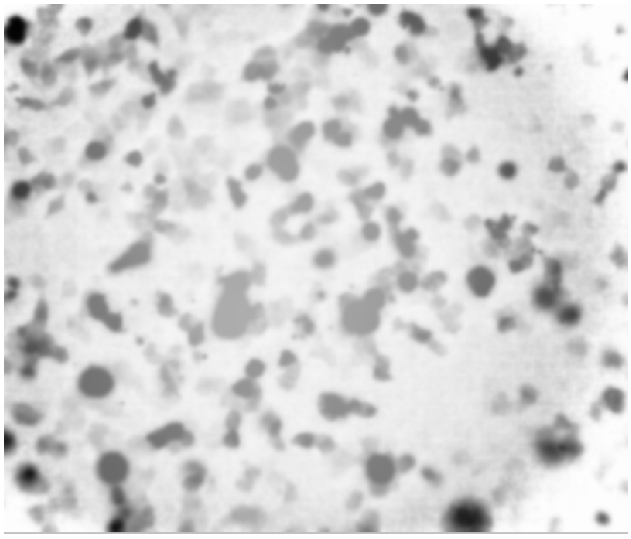


Fig. 3. Image processed by Gaussian blurring.

Stage 3: Thresholding and watershed transform

The Watershed Transform is utilized to segment the image. The software uses the modified java source code file WatershedTransform2D.java: http://ijpb.github.io/MorphoLibJ/ src\main\java\inra\ijpb\watershed. The conceptual ideas have been described on http://cmm.ensmp.fr/~beucher/wtshed.html

License: GNU LESSER GENERAL PUBLIC LICENSE Version 3, 29 June 2007 https://github.com/ijpb/MorphoLibJ/blob/master/LICENSE

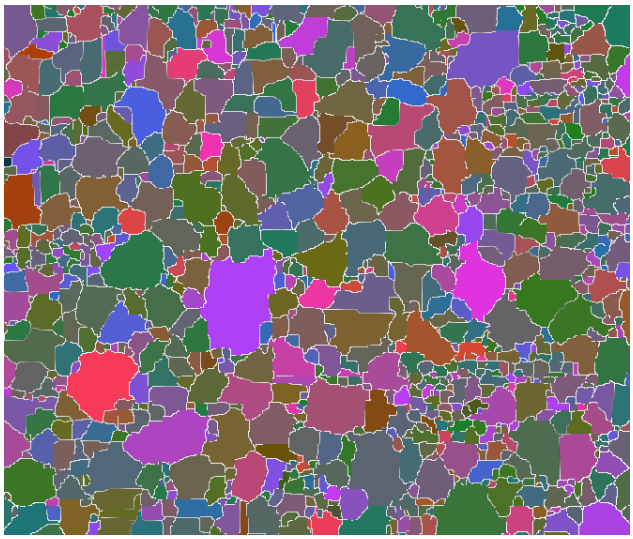


Fig. 4. Image segmentation using watershed algorithm.

Stage 4. Binarization

The watersheds are binarized (separately) using Otsu's method.

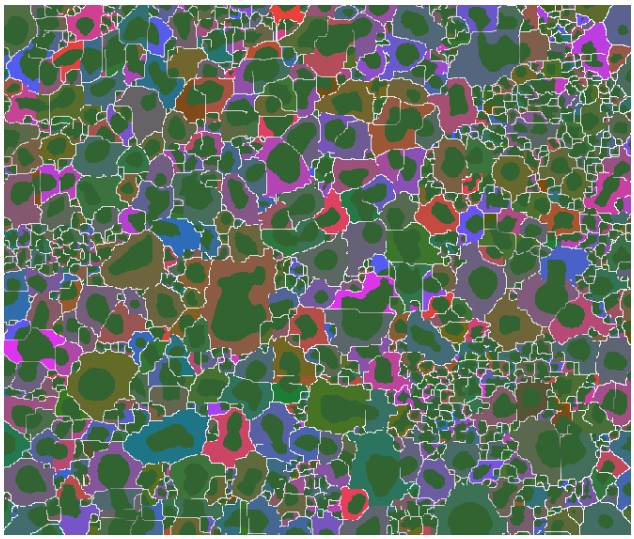
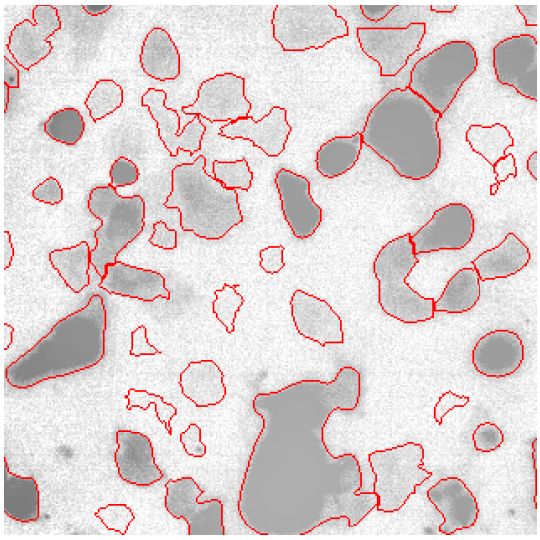


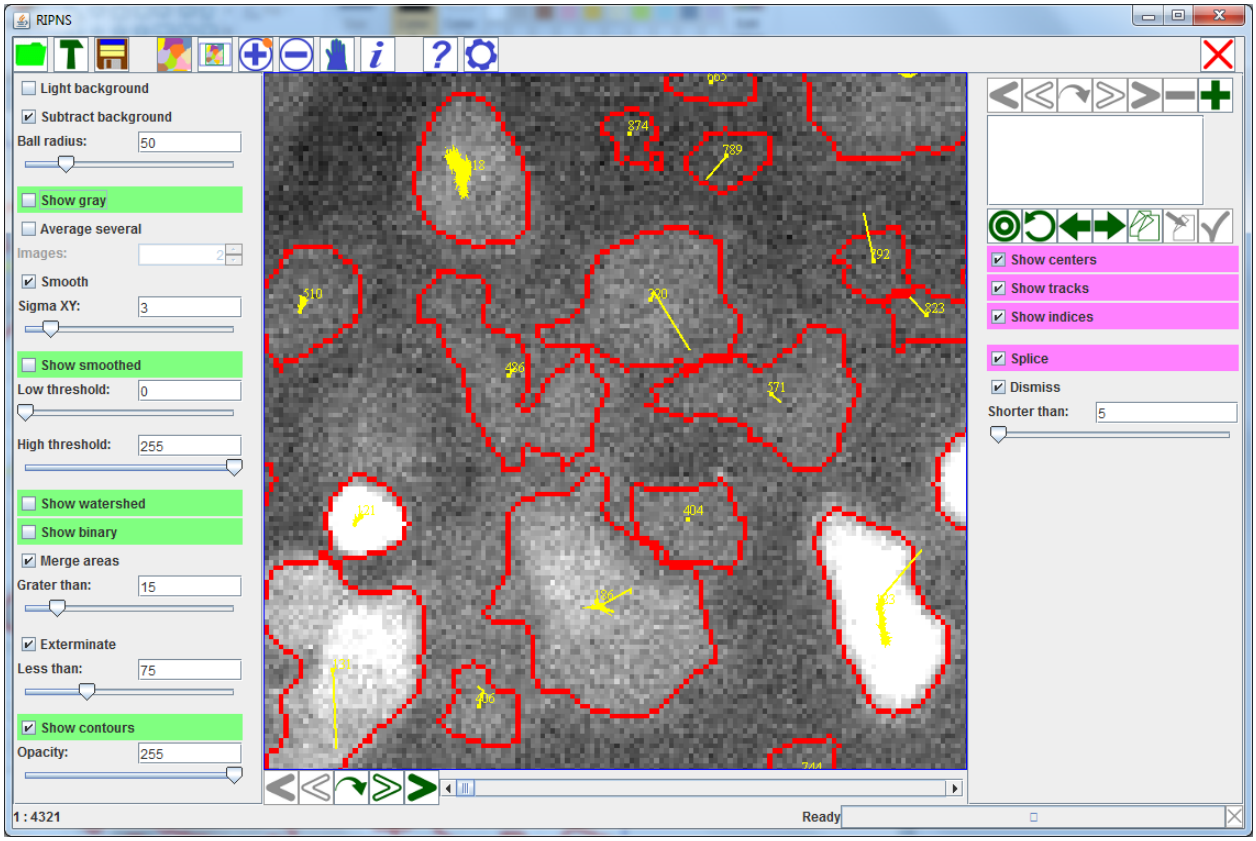
Fig.5. Image binarization using Otsu thresholding.

Stage 5. Final analysis

5.1. Merging of ROIs belonging to the same cell:



5.2. Splicing tracks belonging to the same cell and removing the very short tracks



Other libraries and licenses:

1. ij library for reading TIFF-files. https://imagej.nih.gov/ij/

License: ImageJ and its Java source code are freely available and in the public domain. https://imagej.nih.gov/ij/features.html