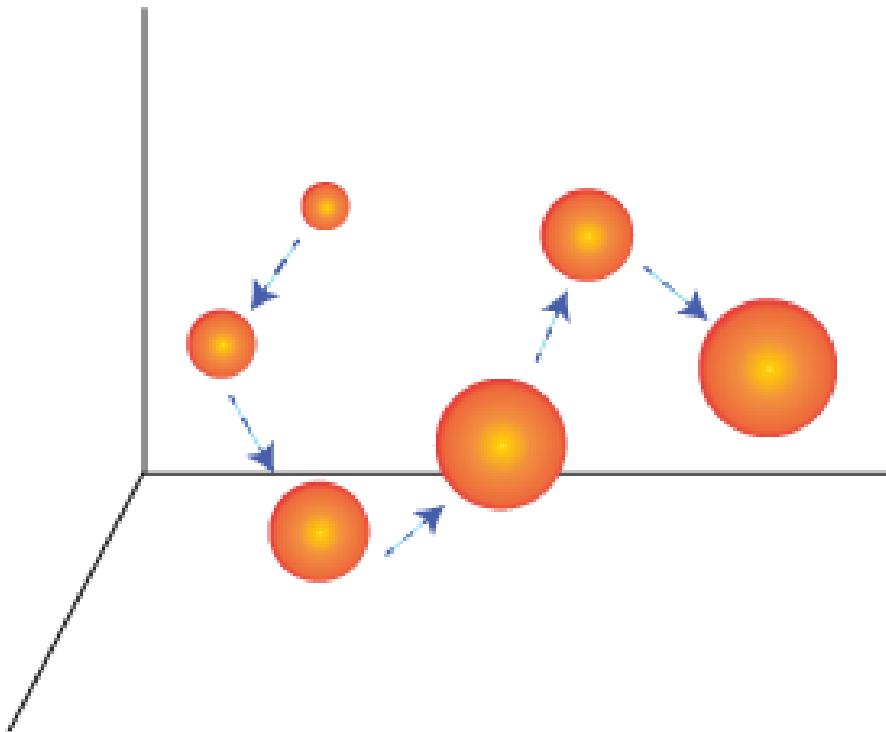


sptPALM Analysis Guide



Adam Hines

A simplified particle localisation and tracking solution.

Contents

1	Introduction	4
1.1	Overview	4
1.2	Spot detection	4
1.3	Particle linking	5
1.4	Mean squared displacement and diffusion coefficients	7
1.5	Comparison of SPA to MetaMorph/PALMTracer	9
2	Using SPA	11
2.1	Installing and setting up SPA	11
2.2	Preparation of image sequences	15
2.3	File directory structure	17
2.4	Setting analysis parameters	17
2.5	Setting threshold values	17
2.6	Running analysis and analysing results	17

1 Introduction

Thank you for downloading and trying sptPALM Analysis (SPA). SPA is a semi-automated single particle tracking tool utilising a graphical user interface (GUI) that interfaces with the ImageJ plugin TrackMate. Users are able to automatically load image sequences of photoactivatable localisation microscopy (PALM) and utilise a built in tracking algorithm for single particles (spt). The semi-automated component refers to the fact that users are required to convert image sequences to the correct format and determine threshold values for spot detection themselves. This is advantageous as auto-detecting threshold values can often lead to over or under-estimation of the spot detection threshold.

SPA was developed by Adam Hines at the Queensland Brain Institute. If utilising this tool in your publications, please cite (*reference*) as well as the original author of Track-Mate (*reference*).

1.1 Overview

This chapter will provide an overview on how the analysis script operates and what algorithms the program uses in order to reliably detect and track the mobility of particles. It will begin by briefly describing the spot detection component of the analysis and the tracking algorithm utilised, however for further information please refer to the source documentation of TrackMate ([TrackMate Algorithms](#)).

1.2 Spot detection

The chosen method for spot detection was the Lapacian of Gaussian (LoG) fitting algorithm. This algorithm takes low resolution spots and fits a LoG over pixels in an x,y plane. The threshold specified will determine which detected spots are appropriate and which are not, and this should be done by the user with their own eyes as this avoids having too many innappropriate or too few appropriate spots. A visualisation of how the spot detection algorithm works is presented in Figure 1.

There are also two settings that are enabled during the spot detection component of SPA. Median filtering and subpixel localisation are enabled to remove the generation of 'jagged' particle linked tracks as well as to improve the acuity of spot detection.

There are other methods for spot detection available in TrackMate, and it should be encouraged to explore the best option for your needs. Embedded into the software however is the LoG detection. SPA is an open-sourced program, so editing the source code will be your only option in order to hardwire a difference detection algorithm - and the same goes for the tracking method (covered in the next section).

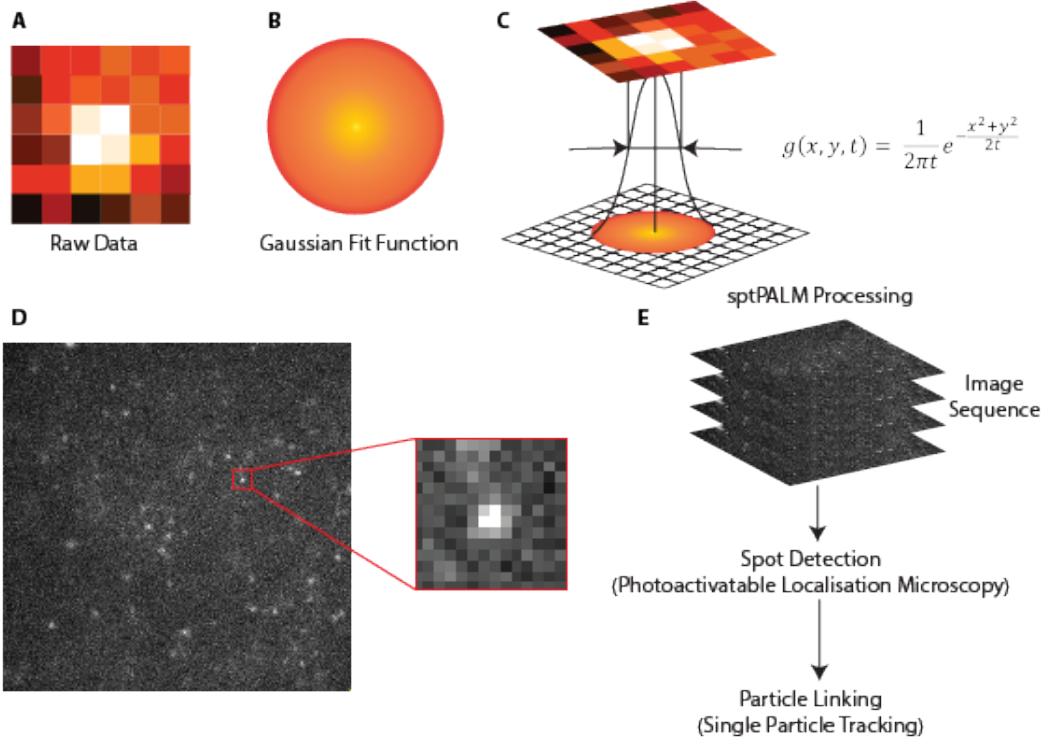


Figure 1: Visualisation of particle detection in SPA. **A** Schematic of typical raw data observed in photoactivatable light microscopy (PALM) experiments, consisting of low resolution pixels with a bright centre with decreasing intensity from the centre. **B** Example of a Gaussian fit of the raw data in **A**. **C** The Gaussian fit function transposing the low resolution raw data into a high-resolution spot. Formula shown if the Lapacian of Gaussian (LoG) spot detection used in SPA analysis. **D** Representative raw data from a PALM experiment highlighting a single spot. **E** Schematic of the workflow used in SPA. Image sequences have parallel threading in spot detection and particle linking in subsequent tracking stages of analysis.

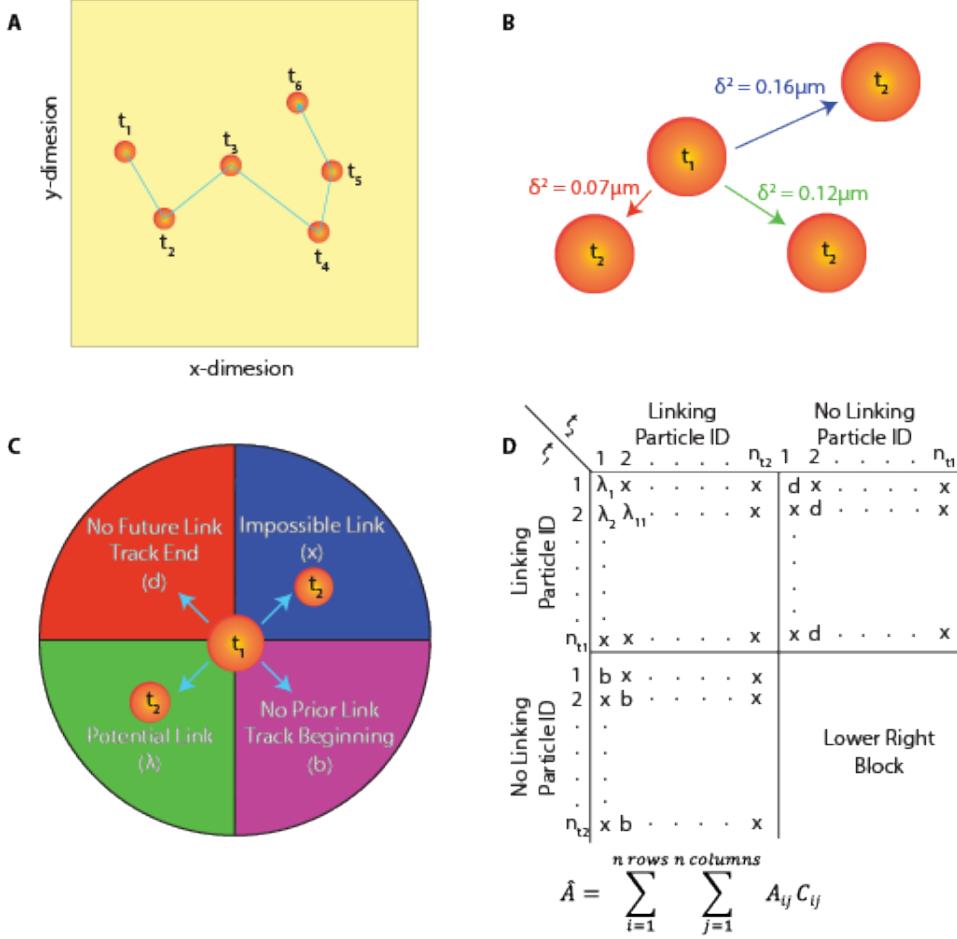
1.3 Particle linking

Single particle tracking has remained a problem for researchers for a very long time. Reliable tracking of multiple particles over a long time frame can cause issues such as linking particles together that are not the same, incorrectly linking a particle that has newly appeared, or even incorrectly ending a track. Figure 2 **A** shows an example of what a tracked particle looks like in an x,y plane. SPA uses the particle linking algorithm known as a linear assignment problem (LAP) cost matrix in order to evaluate and ultimately minimise the cost in favour of the most likely linked particles between frame t_1 and t_2 . The cost values in this context is simply the squared displacement (δ^2) between two detected particles. Each particle in frame n_1 has a potential link associated

to every particle in frame t_2 (Figure 2 **B**), meaning that for each particle in t_1 there are several cost values associated to that particle and the LAP algorithm attempts to find the lowest sum cost for every single particle between frames t_1 and t_2 .

What about particles that have no prior or future link? At a certain point during a recording, particles will bleach and new particles will appear, so how are these dealt with? A separate matrix is established where the previous cost value of a particle linking, (δ^2), is multiplied by a factor of 1.05. If there is a cost that is lower than this value, the particle will have a potential link. If the multiplied cost factor satisfies the lowest sum calculation then that particle, depending on context, will either have no future linkages or is defined as a newly appearing particle.

Particles that are at disparate ends of an x,y plane are easily dealt with by simply setting a maximum cutoff that two particles can physically link. If the (δ^2) exceeds this value, then the link is impossible and cannot satisfy the lowest cost sum. A summary of the potential outcomes of a particle linking between two frames is shown in Figure 2 **C**.



1.4 Mean squared displacement and diffusion coefficients

The main metric utilised in single particle tracking is the mean squared displacement (MSD), which describes the distance a particle moves from its initial displacement in relation to time. In association with the MSD, the diffusion coefficient is another metric that describes the diffusivity of molecules in space. The diffusion coefficient is empirically derived from the MSD and combined describe the mobility of tracked particles (Figure 3 **A** and **B**). The diffusion coefficient values can be represented on a curve as a relative frequency of a whole, taking a logarithmic conversion and counting the frequency in bins

of -5 to 1.

Useful quantifications of the MSD and diffusion coefficients are the area under the curve and the mobile to immobile ratio, defined as the ratio between binned log diffusion coefficients above -1.6 (related to a speed of X) and below -1.6 (Figure 3 **C** and **D**). These two metrics allow for simple statistical analysis.

SPA automatically calculates the MSD and diffusion coefficients thanks to a piece of code from *Reference*. Area under the curve and mobile to immobile ratios are not currently calculated within SPA.

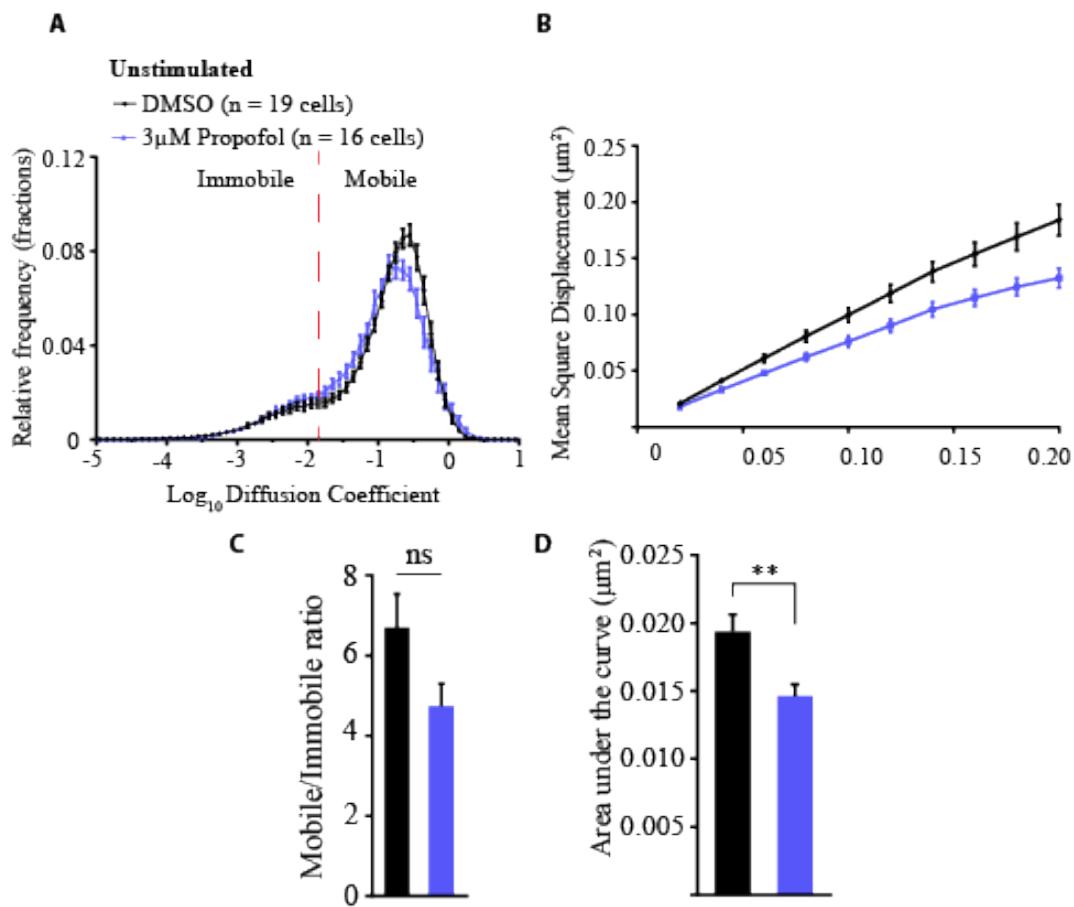
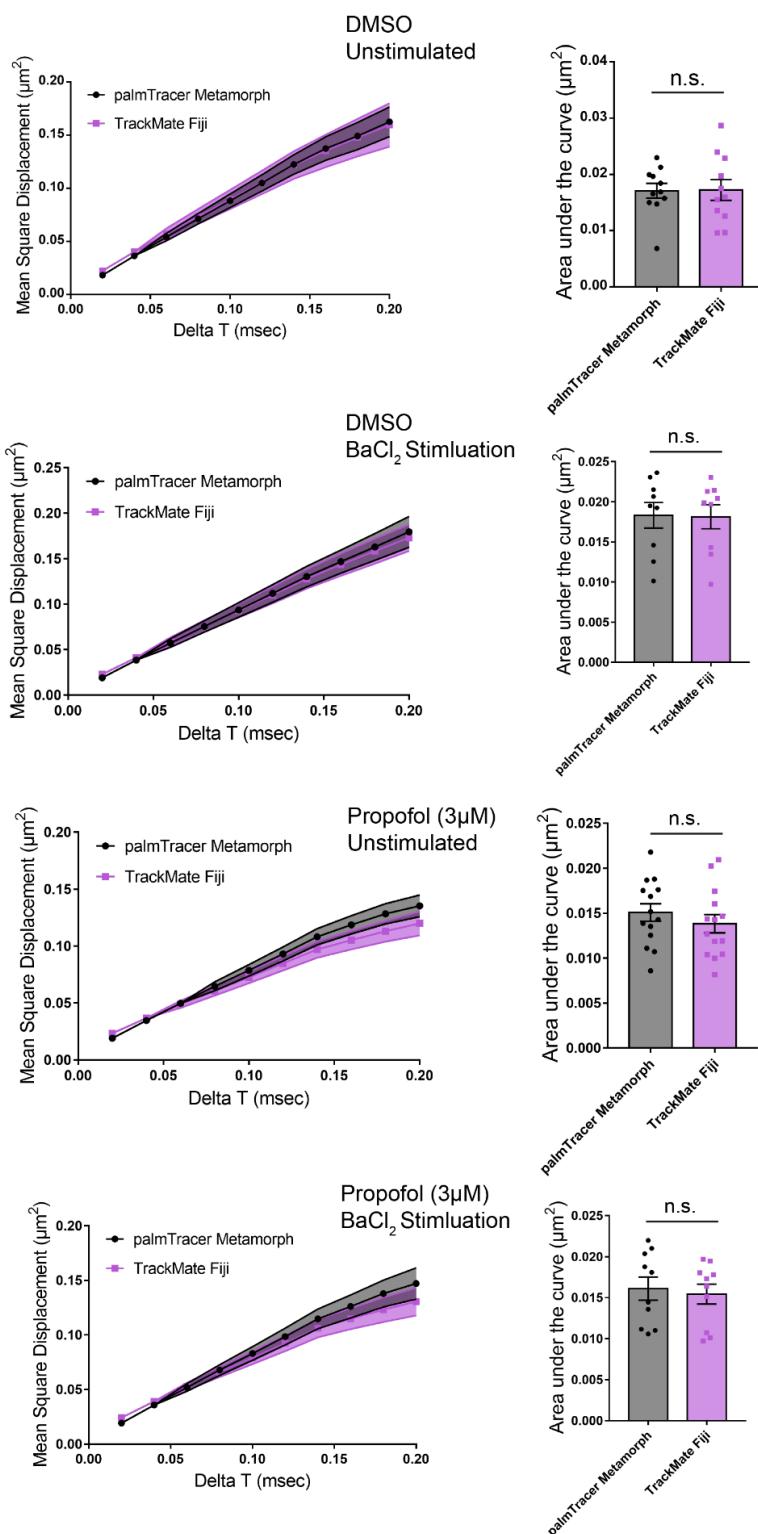


Figure 3: Example of the metrics of single particle tracking experiments. **A** Relative frequency of diffusion coefficients in Munc18-1 mEos2 tracked particles transfected in PC12 cells stimulated with BaCl_2 in the presence of dimethyl sulfoxide (DMSO) or 3 μM propofol. **B** Mean square displacement of mEos2 particles tracked over time. **C** Mobile to immobile ratio calculated from the diffusion coefficients in **A** and area under the curve calculated from **B**.

1.5 Comparison of SPA to MetaMorph/PALMTracer

How does SPA stack up against other localisation and tracking software? Given that TrackMate is a completely open-sourced and free plugin that also happens to run in an open-source and free software, Fiji, how would it fare against MetaMorph - a several thousand dollar piece of software that is often unaccessible to a lot of researchers. MetaMorph makes use of a plugin called PALMTracer, developed by John-Baptiste (*Reference*), in order to localise and track particles. PALMTracer is not an automated workflow and requires the user to manually process each image sequence themselves, in comparison to SPA which allows users to click a single button and walk away.

We can clearly see in Figure 4 that there is no significant difference between analysis performed on MetaMorph vs SPA/TrackMate.



2 Using SPA

This section will explore how to setup, run, and look at the output of SPA. In order to run SPA, you will require MATLAB 2017b or higher as well as some considerable computing power. SPA will run on most modern computers, however older machines may struggle to allocate appropriate computing power for analysis, particularly for large image sequences (1gb). There is a copy of Fiji included with the download and is stored locally in the analysis folders.

2.1 Installing and setting up SPA

To begin, obtain the latest version of SPA through GitHub. Create a GitHub account and download the code and store it somewhere local on your computer (e.g. the MATLAB directory). Access the SPA GitHub repository [here](#), click "Clone or download", and then "Download ZIP" (Figure 5).

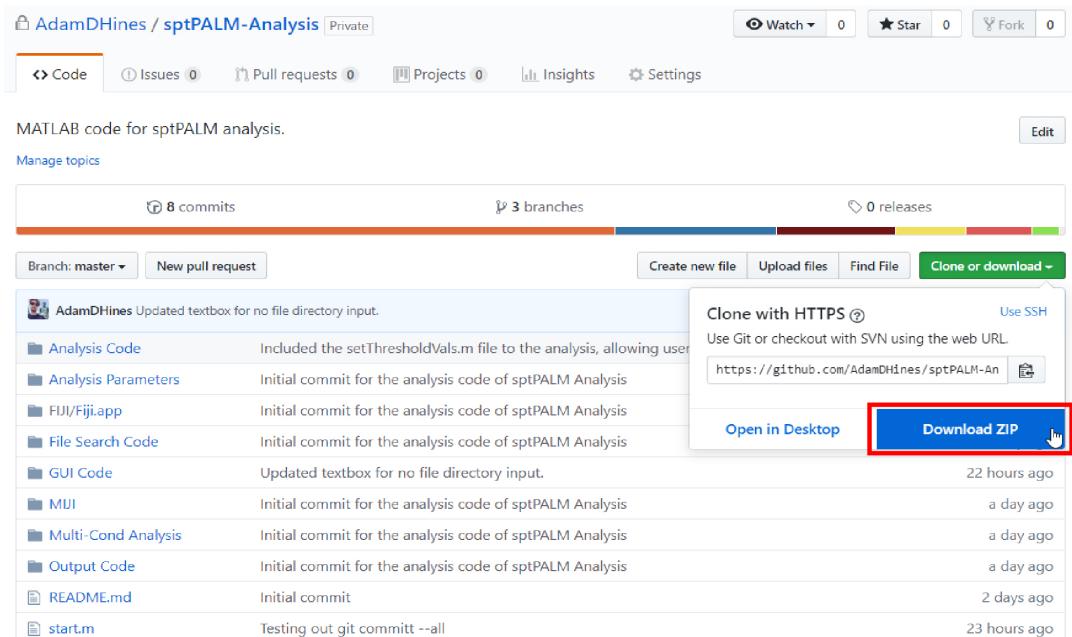


Figure 4: Accessing GitHub to download the latest version of SPA.

Once the ZIP folder has been downloaded, create a folder such as "sptPALM" in a local directory (e.g. the MATLAB folder in Documents) and open MATLAB. Once open, you must set the current directory of MATLAB to the folder that you created that contains all of the code and the "start.m" file. To do so, click the folder icon with the green arrow just above the Current Folder panel on the left hand side (Figure 6).



Figure 5: Changing the current working directory to the SPA analysis folder.

Once this is complete, your Current Folder in MATLAB will look something like this (Figure 6).

Now we can go ahead and run SPA. In the command window of MATLAB, type "start" and hit enter. This will initialise the *start.m* script and launch the graphical user interface. When a fresh install of SPA is performed, you will need to set a default directory to search for files. This can be done by simply pressing "Set directory" in the File directory panel up the top or by going File - Set File Directory (Figure 7). Once this is done, the default directory will be saved and you won't have to set it again.

Congratulations, SPA is now setup on your machine. The next few sections will detail how data files should be prepared and organised in order for the analysis to work.

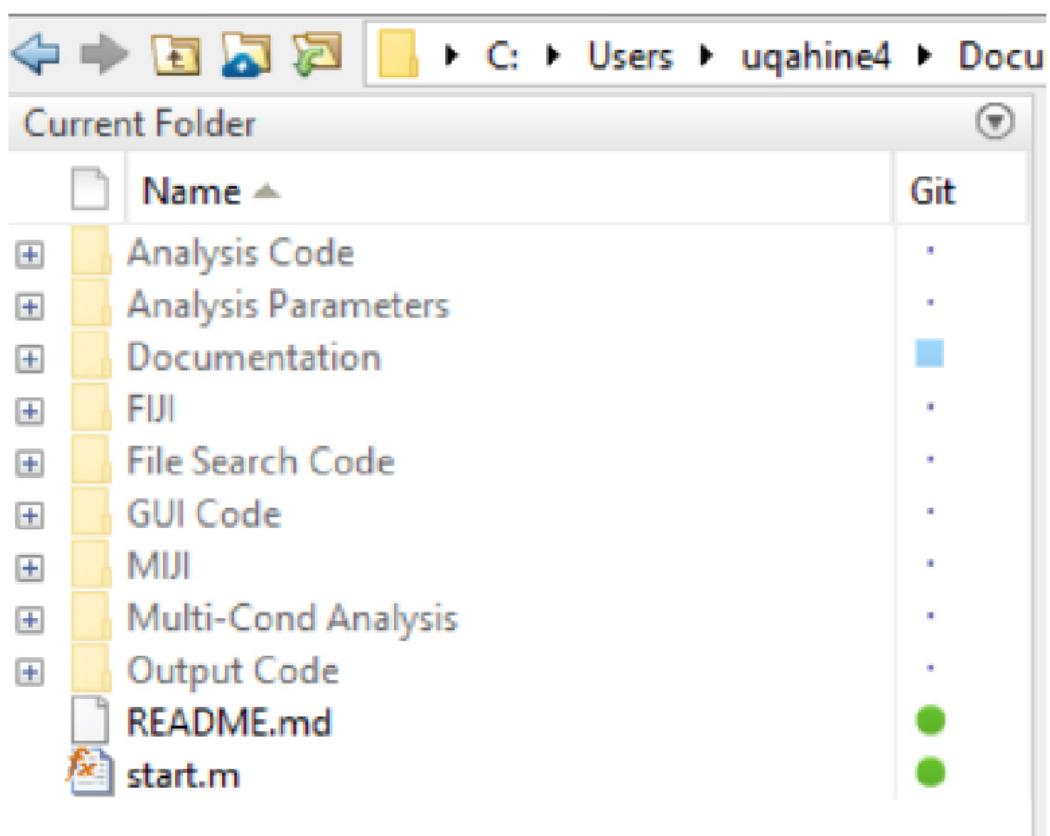


Figure 6: Current directory with SPA code.

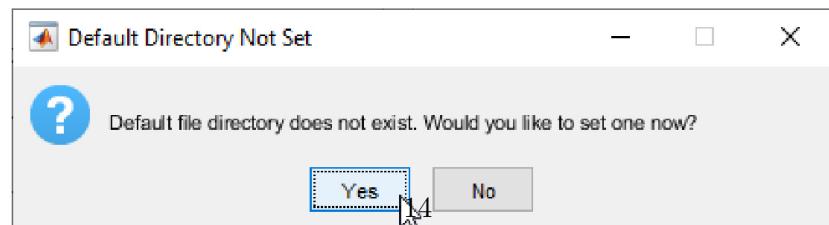
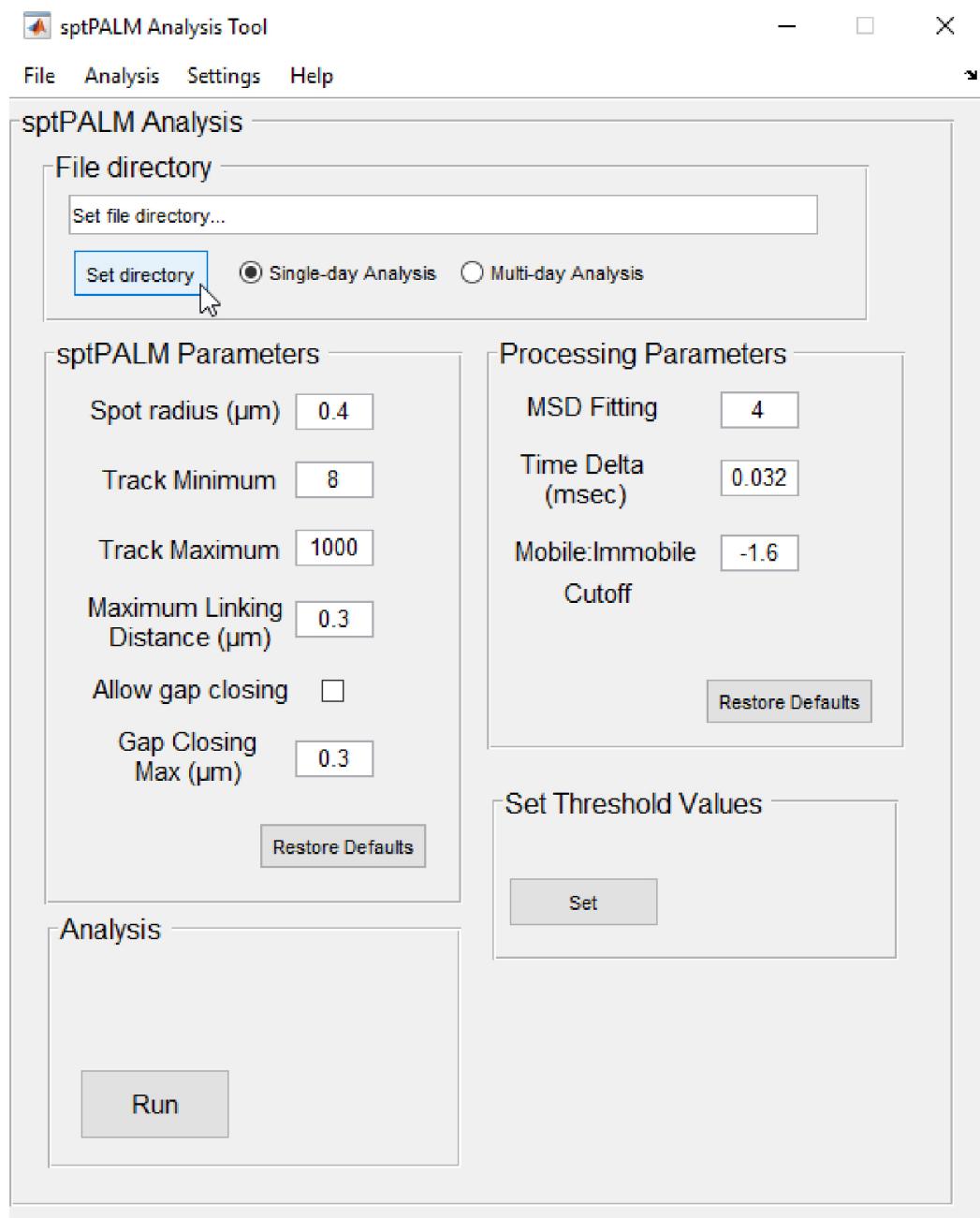


Figure 7: Setting the default working directory in SPA.

2.2 Preparation of image sequences

The image sequences that are used in SPA are .tif stacks. Any other file will not be recognised by SPA and won't be able to be utilised in analysis. Therefore, it is necessary to convert any image sequences to .tif using ImageJ. It is highly recommended that when exporting image sequences from the imaging software of your choice, that you save it in the native format in order to retain metadata about the recordings and then convert to .tif afterwards. What we will also do during this time is establish our threshold values for the spot detection component of the analysis.

To begin, open your image sequence in ImageJ. Check the sequence that the properties are correct, i.e. that the calibration settings for pixel height and width have been set and that the sequence is in terms of t and not z (analysis will stall if the t and z are swapped). Also ensure that the unit of length is "micron". Change the settings if necessary and hit 'OK' (Figure 8). In general, if you're loading the sequence from the source file containing the metadata ImageJ will automatically alter the properties to match.

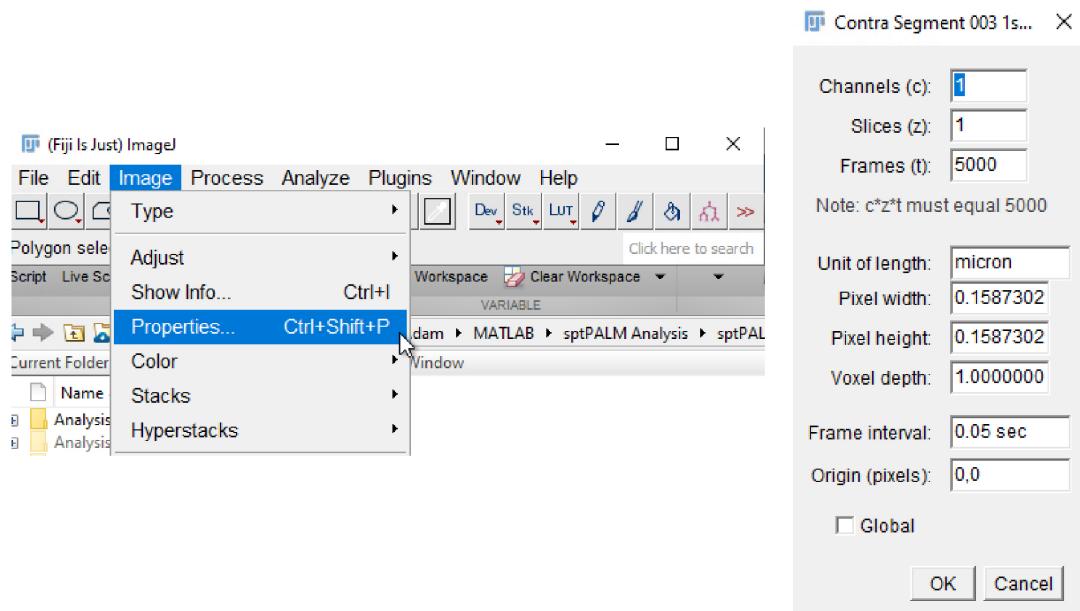


Figure 8: Checking the image sequence properties for calibration and correct z and t orientation.

We now want to determine the threshold value, as well as the spot diameter, for the localisation component of the analysis. Open TrackMate by clicking Plugins → Tracking → TrackMate. The TrackMate GUI will open and simply click next until you reach the 'LoG Detector' settings panel. Here, determine the appropriate spot diameter

and threshold value for the recording. **Important** Ensure that both median filtering and subpixel localisation are selected. Use the preview button to view how the settings localise spots. Make note of the diameter and the threshold value for each individual movie, we will tell SPA when we come to analyse the files what these are.

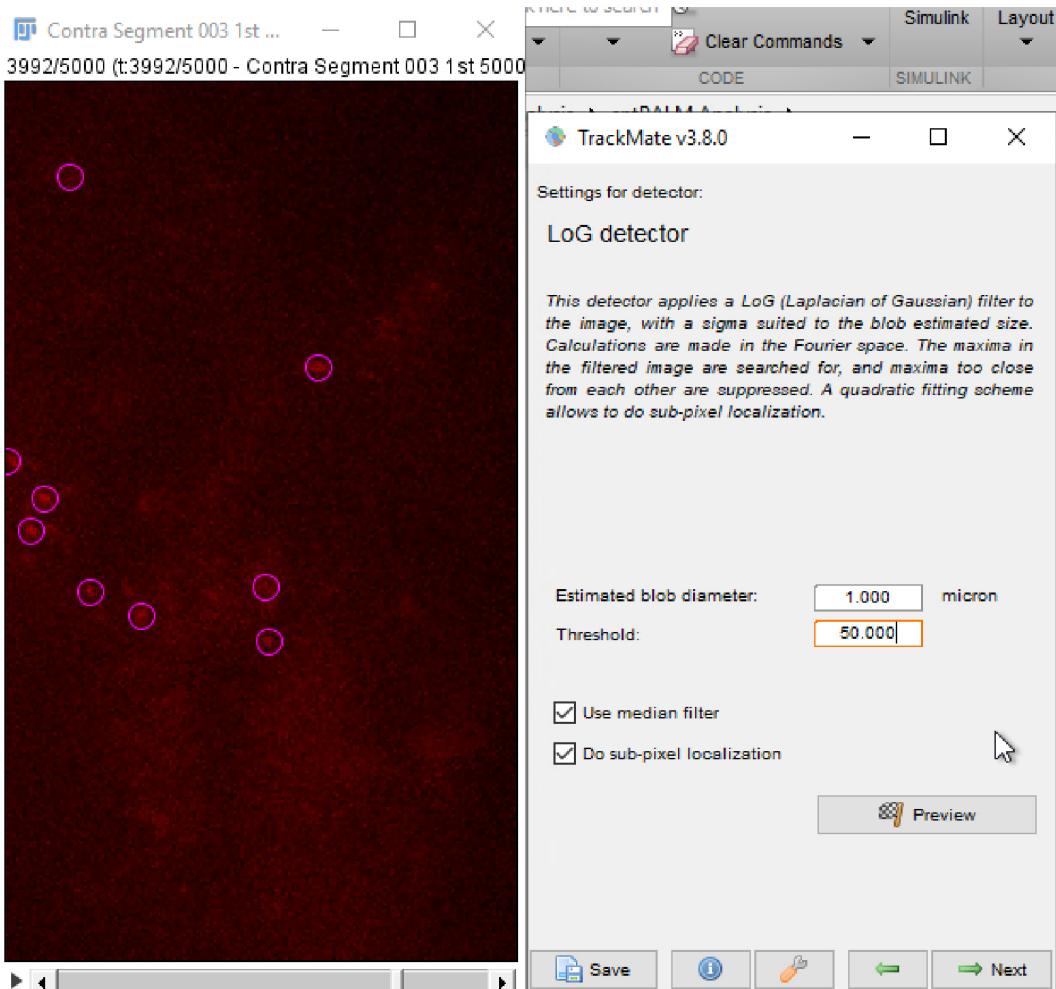


Figure 9: Setting the spot diameter and threshold value for image sequences.

Once that's done, all we need to do now is save the sequence as .tif by going to File → Save as... → Tiff... The next section will show you how to organise your data structures such that the analysis script can search for and find the files for analysis.

2.3 File directory structure

The file structure for the image sequences is very crucial and must be followed exactly. If it is not follow correctly, then the analysis script will not load or find any sequences to analyse and will just throw an error. In short, the sequences are all organised into individual folders inside of a parent folder. An example of this would be a parent folder of *Sx1a-mEos2 28-03-2019* and within this folder are several sub-folders *Recording 1*, *recording 2*, *recording 3*, etc.. The sub-folders are where the original recording (the one containing meta-data) and the .tif converted file should be stored. Adjusting to this file structure from the beginning from saving the initial recordings will save a lot of time in the future.

Let's have a look at what this file structure is. From left to right, we have the main data folder where we can see the folder *20190320* - the folder that contains the sub-folders with our recordings in them. In the middle, we see all of the sub-folders of individual recordings, and on the right we have the contents of the sub-folder. Notice the raw *.czi* file and the converted *.tif* that has been made. SPA will search through all of the folders and try to find a *.tif* file, and eliminate everything else. This is why it's so important to convert all of the image sequences into *.tif* before continuing. If a *.tif* is missing from the folder, the analysis will still run but skip over that experiment.

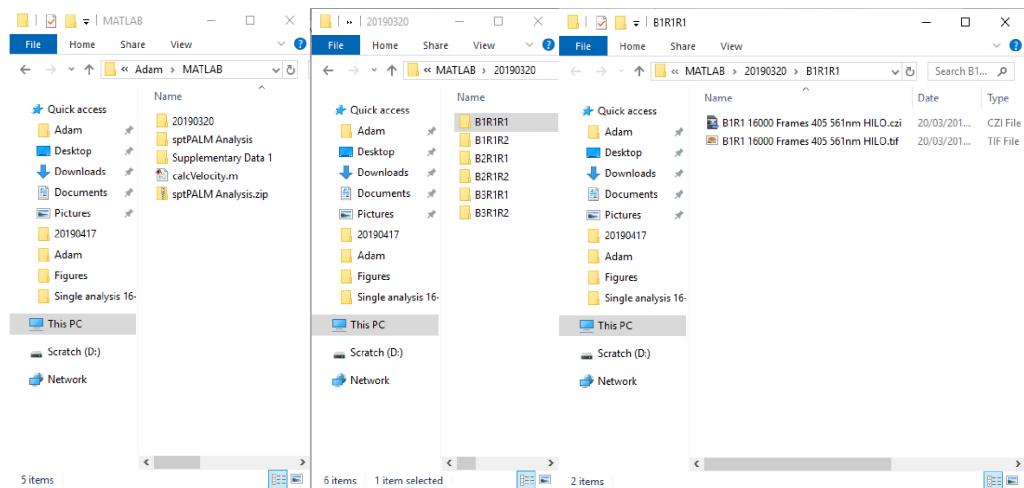


Figure 10: Folder directory structure for the analysis of files.

SPA will output all of the analysis into the same parent folder as the analysis, so getting used to this structure early on will be of great benefit to using the software. The next section will focus on how to set the analysis parameters (but not how to derive them to tailor your experiments).

- 2.4 Setting analysis parameters
- 2.5 Setting threshold values
- 2.6 Running analysis and analysing results