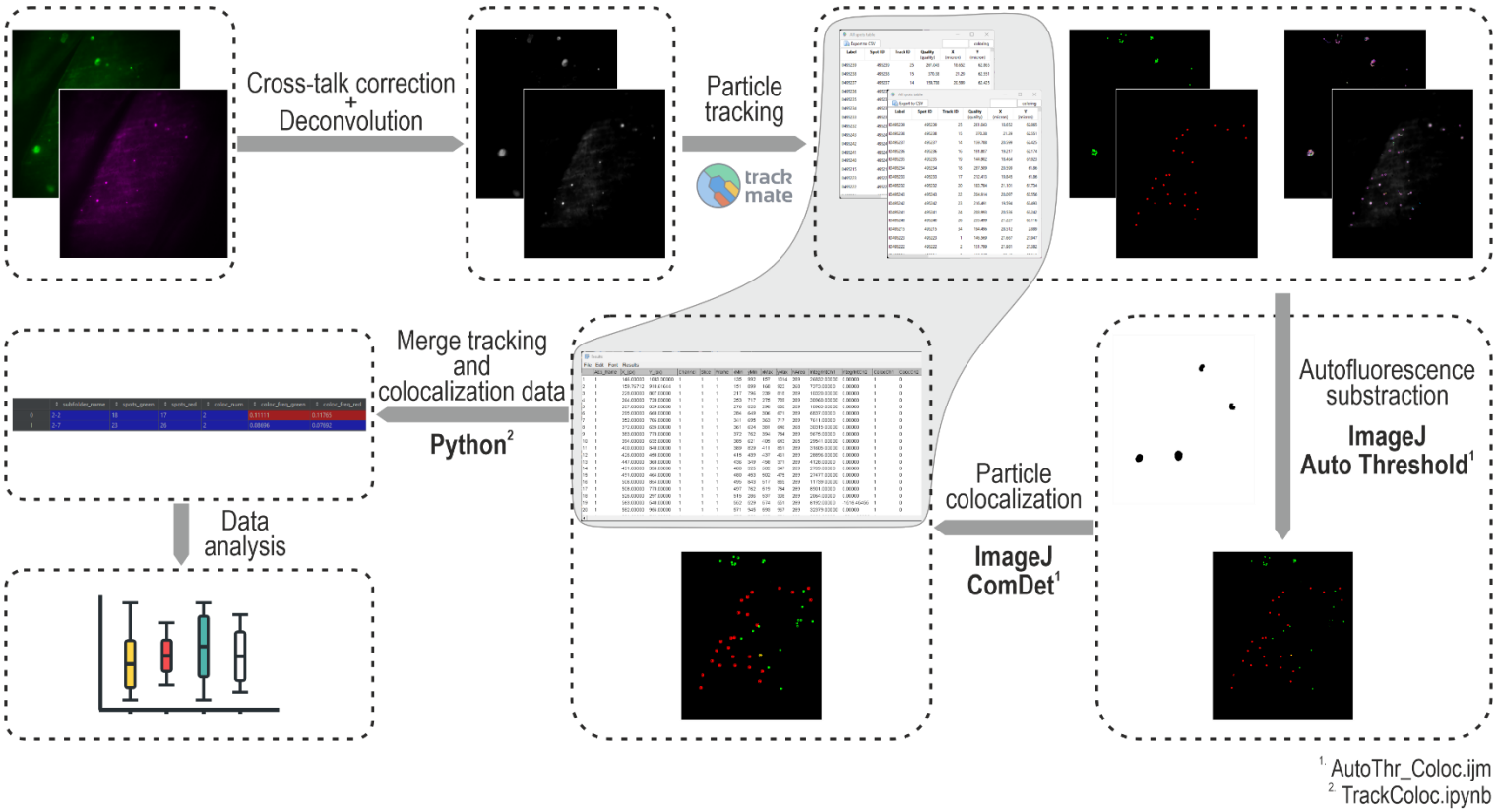
# WORKFLOW – TIME SERIES COLOCALIZATION




This workflow determines the object-based colocalization frequency in time series images. Once the images are acquired, per-image channel crosstalk correction and deconvolution are performed following particle tracking using the ImageJ plugin TrackMate. Using ImageJ macro language, particles that were detected within autofluorescence regions are discarded, and filtered label images are further used for particle colocalization via the ImageJ plugin ComDet. Tracking data generated in TrackMate and detected colocalization information are then merged and filtered to count unique colocalization events over time using a custom Python script.

## REQUIREMENTS

This workflow has been tested on Windows.

Crosstalk correction and deconvolution have been performed using Huygens Professional for Win64 systems with Huygens compute engine 22.10.0p3 64b. Other open-source software can be used for this purpose.

Particle tracking and colocalization analysis were performed using the ImageJ plugins TrackMate version 7.10.2 and ComDet version 0.5.5, respectively.

The ImageJ distribution Fiji version 2.10.0 was used for macro and plugin installation and execution.

For Python script running, the following versions were used:

- Python v3.10.9
- Python packages:
    - os v0.1.14
    - pandas v1.5.2
    - tkinter v8.6.12

## WORKFLOW STEPS

1. Crosstalk correction and deconvolution are performed individually for every image using the software Huygens Professional built-in wizards, using the guidelines offered in the following resources:
    a. Deconvolution using HuygensPro, Image Processing How-To Guides, Copyright © 2020 Scientific Computing Facility, MPI-CBG
    b. https://mpicbg-scicomp.github.io/ipf_howtoguides/guides/Huygens_Deconvolution.html
    c. Huygens Professional User Manual 22.10
2. Particle tracking is also performed individually for every image. Spot detection and tracking algorithms used vary on the imaging setup. For more information see the TrackMate documentation and tutorials (https://imagej.net/plugins/trackmate/#documentation-and-tutorials).
   In order to perform subsequent steps, the required particle diameters are set to 0.5 microns for the green channel (ch00) and 0.7microns for the red channel (ch01). Other parameters are set at the discretion of the user.
   Note: for spot detection, TrackMate offers the possibility of applying a median filter for salt and pepper noise. An alternative form of noise reduction that has proven useful is a Gaussian filter prior to spot detection.
3. Autofluorescence subtraction and colocalization. A custom ImageJ macro (AutoThr_Coloc.ijm) is used for autofluorescence detection in the desired channel (currently set to ch00). This macro assumes that autofluorescence is defined by bright and large regions, which can be automatically thresholded and subsequently used to filter previously detected points.
   The same script runs object-based colocalization analysis with fixed parameters on label images generated by TrackMate, and optimized for the particle size defined in step 2.
   Note: the macro can be also run to generate control images to determine if the amount of colocalization found is higher than random coincidence. For this, the green channel (designated as ch00) is rotated 90 degrees to the right and the red channel 90 degrees to the left.
4. Tables obtained after spot detection, tracking and colocalization are then used as input in a python script (TrackColoc.ipynb) in order to merge data based on spot centroid localization and track persistence. To improve robustness analysis and consider only stable colocalization events, only particles that persist three or more frames are taken into account. Colocalization frequency is determined for all the samples in a folder following a predefined structure, stored and exported in a csv table.

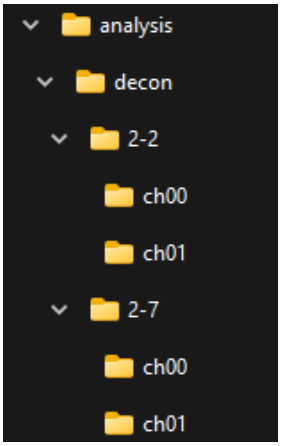
## EXAMPLE DATA AND STEP-BY-STEP GUIDE

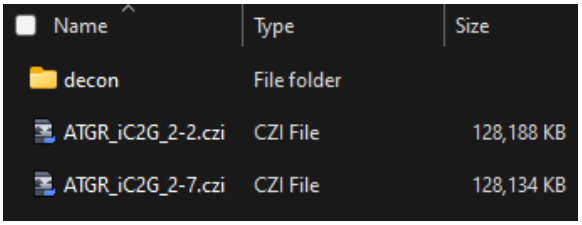
The example data consists of a two-sample dataset of *Arabidopsis thaliana* hypocotyl epidermal cells imaged using the Zeiss ELYRA PS.1 imaging system with the alpha Plan-Apochromat objective 100x Oil DIC M27 Elyra (NA 1.46, WD 0.11 mm). Collected images are 16-bit.

Crosstalk-corrected and deconvolved images are included in the dataset in folders designated ch00 and ch01, as subfolders of sample folders.

Example data structure is depicted below:

> analysis
> ATGR_iC2G_2-2.czi
> ATGR_iC2G_2-2.czi
> > decon
> > > 2-2
> > > > ch00
> > > > ch01
> > > 2-7
> > > > ch00
> > > > ch01



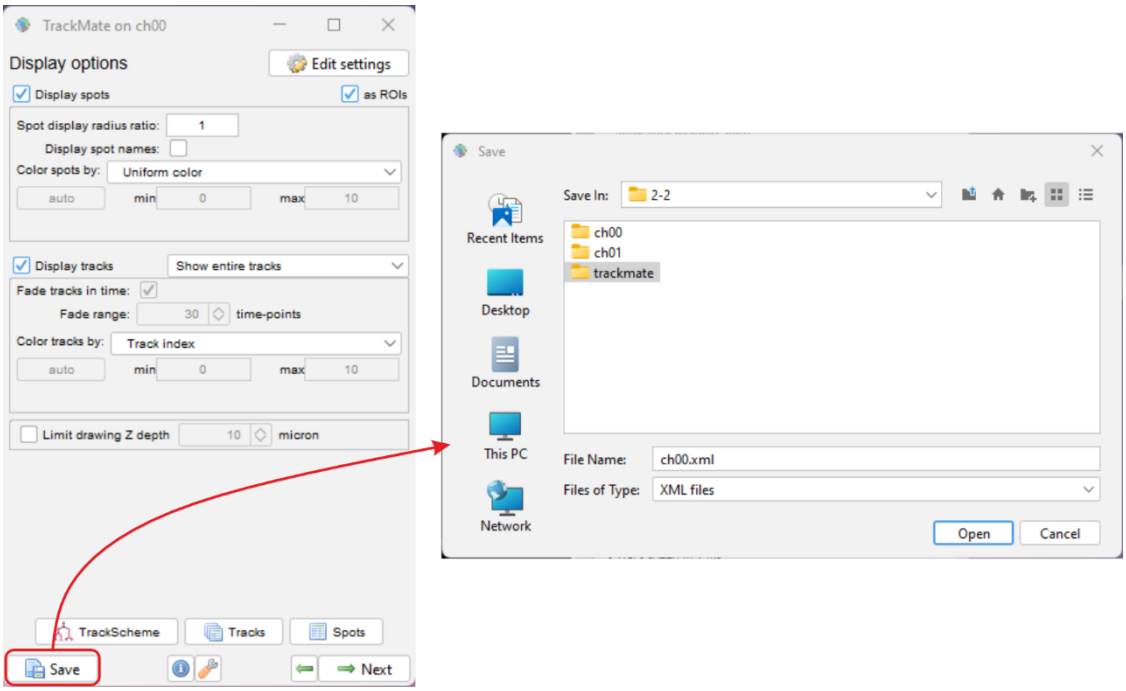




Each image series corresponding to individual sample channels (e.g. analysis/decon/2-2/ch00) was run for spot detection and tracking in ImageJ TrackMate plugin, using the following parameters, with no spot or track quality filter.

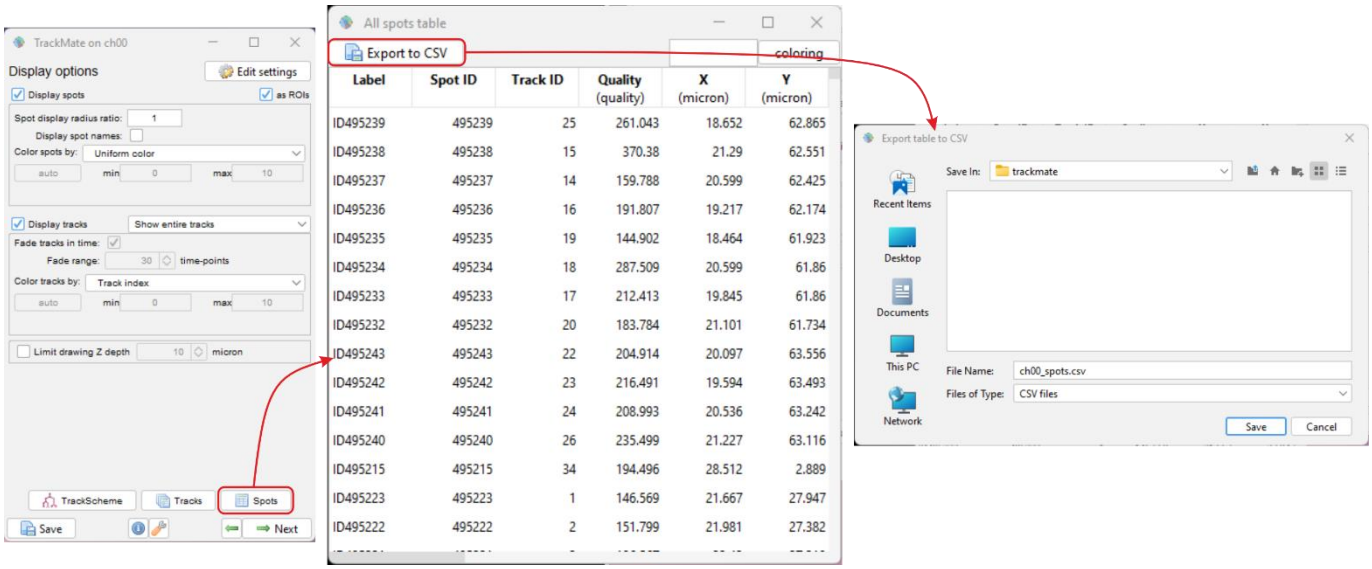
| Subfolder name | Spot Detector | Spot radius (micron) | Quality threshold | Median filter? | Subpixel localization? | Tracker | Max. linking distance (micron) | Max. gap closing distance (micron) | Max. frame gap | Pre Gaussian filter? | Gaussian filter radius |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 2-2/ch00 | DoG | 0.25 | 100 | FALSE | FALSE | Simple LAP tracker | 2.5 | 1.5 | 1 | FALSE | NA |
| 2-2/ch01 | DoG | 0.35 | 800 | FALSE | FALSE | Simple LAP tracker | 2.5 | 1.5 | 1 | TRUE | 2 |
| 2-7/ch00 | DoG | 0.25 | 60 | FALSE | FALSE | Simple LAP tracker | 2.5 | 1.5 | 1 | FALSE | NA |
| 2-7/ch01 | DoG | 0.35 | 230 | FALSE | FALSE | Simple LAP tracker | 2.5 | 1.5 | 1 | TRUE | 2 |

The following outputs need to be saved for further processing:

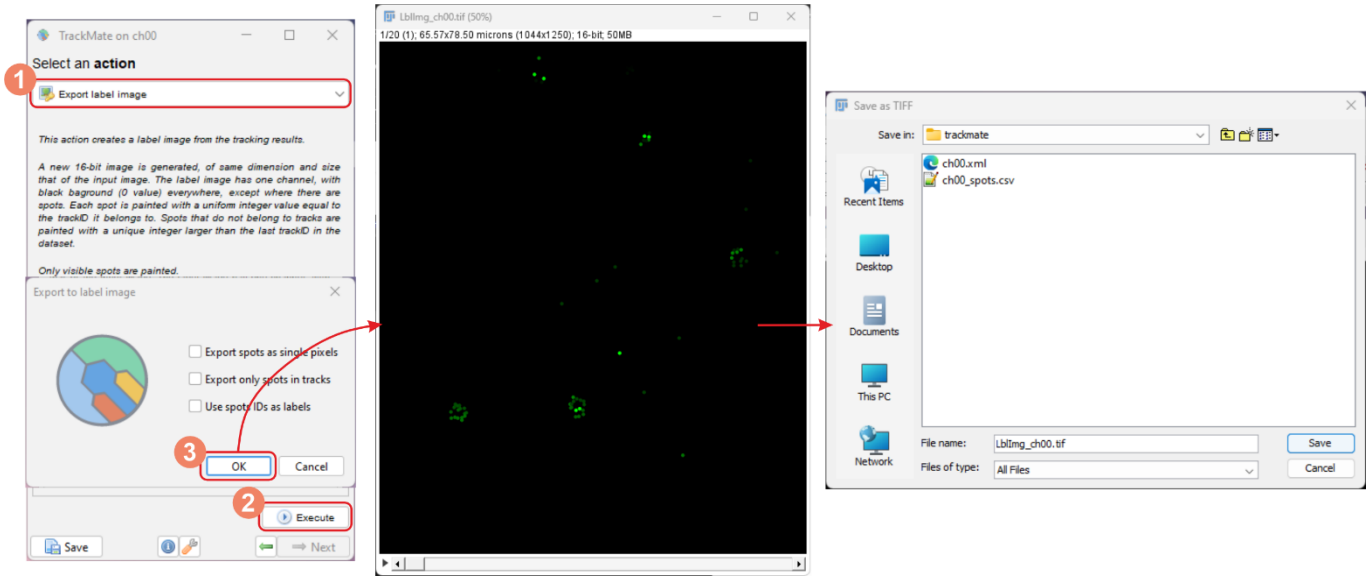
1. Trackmate log files: ch00.xml and ch01.xml




2. Trackmate spots tables: ch00_spots.csv and ch01_spots.csv



3. Trackmate label images: LblImg_ch00.tif and LblImg_ch01.tif

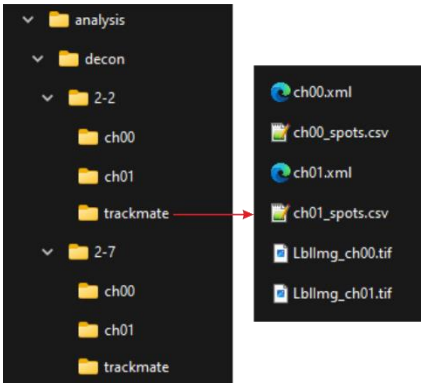



Following TrackMate spot tracking evaluation, we will proceed to detect autofluorescence regions (in this case, visible in the green channel – or ch00).

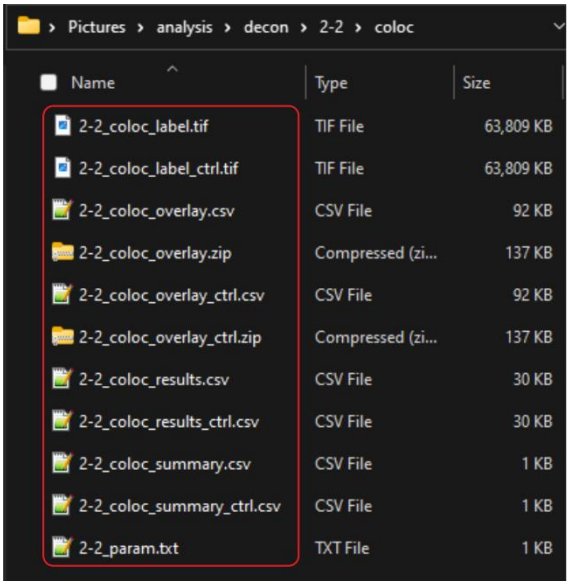
The ImageJ macro script AutoThr_Coloc.ijm can be executed after installation in ImageJ (Plugins>Macros>Install…), or from the ImageJ script editor (if parameters need to be adjusted – e.g. ComDet colocalization value modification or AutoThreshold method selection).

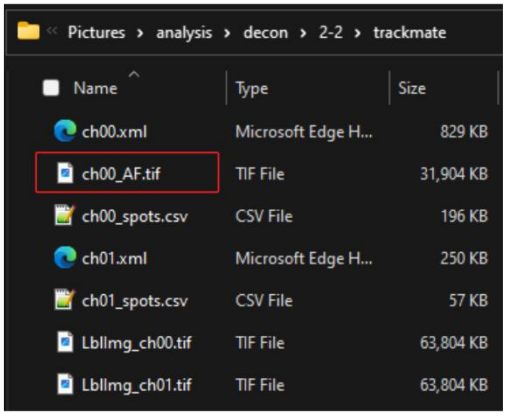
At this point, the required file structure is as follows:

> analysis
ATGR_iC2G_2-2.czi
ATGR_iC2G_2-2.czi
> decon
> 2-2
> ch00
> ch01
> trackmate
ch00.xml
ch01.xml
ch00_spots.csv
ch01_spots.csv
LblImg_ch00.tif
LblImg_ch01.tif
> 2-7
> ch00
> ch01
> trackmate
ch00.xml
ch01.xml
ch00_spots.csv
ch01_spots.csv
LblImg_ch00.tif
LblImg_ch01.tif

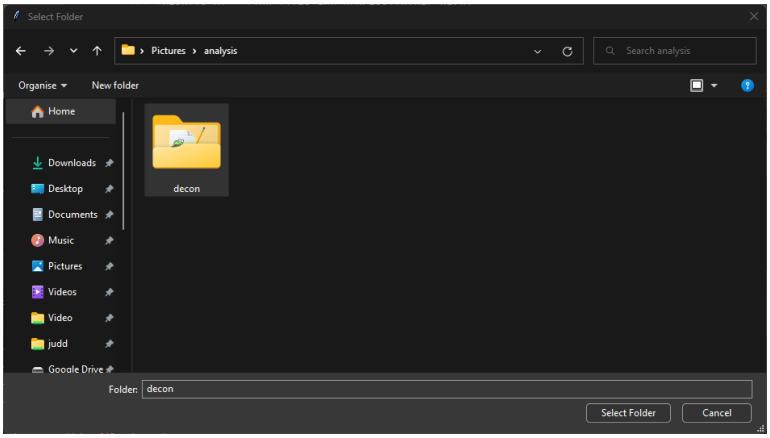


After running the macro, colocalization outputs will be included in each sample folder, under the newly created decon/sample/coloc folde, as depicted in the below image.




Lastly, the python script TrackColoc.ipynb will be used to merge the obtained tracking and colocalization data and calculate colocalization frequency for all the samples included in the selected parent folder. The script is stored in a jupyter notebook. Once the script is running, it will prompt the user to select the folder where all sample subfolders are contained.




After the script is run, a results file (results_df.csv) containing the filtered number of spots for each channel and the corresponding reciprocal colocalization frequencies for each of the analyzed time series is created in the selected folder. This results file can be used for further data evaluation.

| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 1 | Column1 | subfolder_name | spots_green | spots_red | coloc_num | coloc_freq_green | coloc_freq_red | |
| 2 | 0 | 2-2 | 18 | 17 | 2 | 0.111111111 | 0.117647059 | |
| 3 | 1 | 2-7 | 23 | 26 | 2 | 0.086956522 | 0.076923077 | |
| 4 | | | | | | | | |
| 5 | | | | | | | | |