



UNLOC user's manual

UNsupervised particle LOCalization algorithm

UNLOC authors:

Nicolas BERTAUX (developer), Sébastien MAILFERT (co-developer), Didier MARGUET

Jérôme TOUVIER (Plugin developer), Lamia BENYOUSSEF, Asma RABAoui, Roxane FABRE, Marie-Claire BLACHE

Contents

1	New releases	3
1.1	Modifications of the Matlab UNLOC code	3
1.1.1	2018 July - UNLOC v1.1	3
1.1.2	2023 June - UNLOC v1.2	3
1.1.3	2024 April - UNLOC v2.4	3
1.1.4	2024 April - UNLOC v2.51	3
1.2	Modifications of the UNLOC plugin	4
1.2.1	2018 July - Plugin version 1.0	4
1.2.2	2022 November - Plugin version 2.1	4
1.2.3	2023 March - Plugin version 2.2	5
1.2.4	2023 May - Plugin version 2.3	5
1.2.5	2024 April - Plugin version 2.4	5
1.2.6	2024 April - Plugin version 2.51	5
1.2.7	2024 July - Plugin version 2.52	5
1.3	Modifications of the user manual	6
1.3.1	2018 July - User manual version 1.0	6
1.3.2	2022 November - User manual version 2.1	6
1.3.3	2023 March - User manual version 2.2	6
1.3.4	2023 June - User manual version 2.3	6
1.3.5	2024 April - User manual version 2.4	7
1.3.6	2024 April - User manual version 2.51	7
1.3.7	2024 May - User manual version 2.52	7
1.4	Known issues	8
2	About UNLOC	9
2.1	What is UNLOC?	9
2.2	UNLOC modules	10
2.3	Download and installation	10

2.4	Compatibility and system requirements	10
2.5	Evaluation of the performance of UNLOC on synthetic data	11
2.6	Help with UNLOC	11
2.7	UNLOC updating and bug reporting	12
3	Quick start	13
3.1	Data loading	13
3.1.1	Tiff, nd2 and raw files	14
3.1.2	Multiple files or stacks	14
3.2	Analysis	14
3.3	Reconnection	15
3.4	Rendering	15
3.5	Batch	16
3.6	Help	17
4	Detailed description of UNLOC	18
4.1	Analysis	18
4.1.1	For beginners	19
4.1.2	For experts	19
4.2	Reconnection	23
4.3	Rendering	24
4.3.1	For beginners	24
4.3.2	For experts	25
4.4	Batch	28
4.4.1	Module specifications	28
4.4.2	How does it work?	29
5	PSF size: r_0	30
5.1	How to determine r_0 ?	30
5.1.1	How to choose between TIRF vs. non TIRF options?	30
5.1.2	Determine r_0 for your optical setup	31
5.2	How to display r_0 in the <code>Analysis.csv</code> output file?	32
5.2.1	Check if r_0 is written in the <code>.csv</code> output file	32
5.2.2	What should I do if r_0 is not written in the output file?	33
6	Appendix	35
6.1	Supported systems	35
6.2	Advanced installation procedure	35
6.3	Disclaimer	35
7	List of acronyms	36

New releases

1.1 Modifications of the Matlab UNLOC code

1.1.1 2018 July - UNLOC v1.1

Original plugin version published in Biophysical Journal (Mailfert et al., BiophysJ, 2018).

1.1.2 2023 June - UNLOC v1.2

UNLOC was slightly modified to support RAW data from QCM (Quality Control Maps) including the parameter file called "data_info.txt".

1.1.3 2024 April - UNLOC v2.4

UNLOC has been customized for a specific collaboration and has not been distributed.

1.1.4 2024 April - UNLOC v2.51

UNLOC was compiled from the 1.2 version to be compatible with Matlab 2024a release. No modification of the core of UNLOC, only the possibility to read RAW data file from PCO cameras.

1.2 Modifications of the UNLOC plugin

1.2.1 2018 July - Plugin version 1.0

Original plugin version published in Biophysical Journal (Mailfert et al., BiophysJ, 2018).

1.2.2 2022 November - Plugin version 2.1

- Due to a modification of the ProcessBuilder function in the Java 1.8.0_322 version, we had to recompile the plugin with a syntax modification of the call of ProcessBuilder. This has NO impact on the UNLOC results, it is only link to the call of the Matlab executable files on the Windows platforms.
- Point Spread Function (PSF) size: the previous version of the plugin did not write r_0 in the .csv output file. Now, by default, r_0 is automatically written:

	A	B	C	D	E	F	G
67	DATA						
68	imagesize_i	95					
69	imagesize_j	135					
70	nb_part	31258					
71	nb_frames	7840					
72	time	position_i	position_j	std_error_pcalpha	std_bg_nois	r0	
73	1	11.926	22.284	0.031	1396	24.2	1.271039
74	1	80.915	36.988	0.031	1412	25.0	1.274786
75	1	22.786	121.587	0.029	1425	23.5	1.228009
76	1	42.745	121.539	0.031	1393	24.0	1.270653
77	1	62.759	121.647	0.034	1314	25.3	1.237750
78	2	18.528	37.056	0.029	1427	23.9	1.167355
79	2	78.361	60.642	0.031	1371	23.8	1.297869
80	2	23.904	80.611	0.032	1447	26.3	1.283052
81	2	42.522	80.750	0.031	1444	25.6	1.262750

Figure 1.1: r_0 as column in the output file

- Consequently, we added a new histogram on the Rendering tab displaying the distribution of r_0 and corresponding min/max filters:

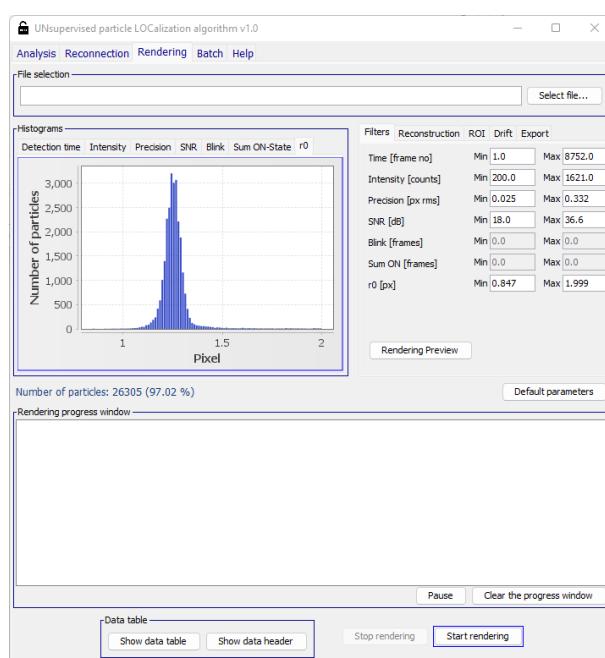


Figure 1.2: Histogram of r_0 in the Rendering tab

1.2.3 2023 March - Plugin version 2.2

- The rendering executable was not correctly compiled in the 2.1 version
- The "Integrated Gaussian" and "Gaussian" rendering views were swapped

If you load a data which has been previously reconnected (see section [4.2](#)), the r_0 histogram will be empty because we cannot evaluate the PSF size of particles that have been grouped over time.

1.2.4 2023 May - Plugin version 2.3

RAW data files coming from PCO cameras are now supported.

1.2.5 2024 April - Plugin version 2.4

This plugin version should not be diffused!

1.2.6 2024 April - Plugin version 2.51

- Updates of the dependencies
- Updates of the Matlab Compiler Runtime version (R2024a)
- RAW data generated by PCO cameras are supported

1.2.7 2024 July - Plugin version 2.52

- Simplification of some plugin windows whose buttons were not being used
- Nikon ND2 files can now be read

1.3 Modifications of the user manual

1.3.1 2018 July - User manual version 1.0

Original user manual version published in Biophysical Journal (Mailfert et al., BiophysJ, 2018).

1.3.2 2022 November - User manual version 2.1

- Addition of a chapter about the release notes (Chapter 1)
- Addition of a chapter about r_0 (Chapter 5)
- Correction of the erroneous formula linking r_0 and FWHM (see section 4.1.2):

- Before correction:

$$r_0 = \frac{\text{FWHM}}{2\sqrt{2 \ln(2)}} \approx 0.59 \times \text{FWHM}.$$

- Corrected formula:

$$r_0 = \frac{\text{FWHM}}{2\sqrt{2 \ln(2)}} \approx 0.42 \times \text{FWHM}.$$

- Correction of the erroneous formula for the conversion of the RMS precision in nm (see section 4.3.2):

- Before correction:

$$\begin{aligned} \text{Precision in nm} &= \frac{\sqrt{2} \times \text{RMS precision} \times \text{Pixel size } (\mu\text{m})}{1000} \\ &= \frac{\sqrt{2} \times 0.18 \times 0.107}{1000} \\ &\approx 27 \text{ nm}. \end{aligned} \tag{1.1}$$

- Corrected formula:

$$\begin{aligned} \text{Precision in nm} &= \sqrt{2} \times \text{RMS precision} \times \text{Pixel size } (\mu\text{m}) \times 1000 \\ &= \sqrt{2} \times 0.18 \times 0.107 \times 1000 \\ &\approx 27 \text{ nm}. \end{aligned} \tag{1.2}$$

1.3.3 2023 March - User manual version 2.2

- Modification of the release chapter to include the plugin version (2.2)

1.3.4 2023 June - User manual version 2.3

- Modification of the release chapter to include the plugin version (2.3)
- Modification of the installation process to include the Matlab Compiler Runtime R2023a
- Modification of the Quick start, Detailed description and Appendix chapters to account for RAW data

1.3.5 2024 April - User manual version 2.4

- Modification of the release chapter to include the plugin version (2.4)
- Modification of the installation process to include the Matlab Compiler Runtime R2024a

1.3.6 2024 April - User manual version 2.51

- Modification of the release chapter to include the plugin version (2.51)

1.3.7 2024 May - User manual version 2.52

- Modification of the release chapter to include the plugin version (2.52)
- Update of screenshots

1.4 Known issues

1. Randomly, the Matlab Compiler Runtime (MCR) is not launched properly and UNLOC displays this message ("Undefined function or variable 'matlabrc'"):

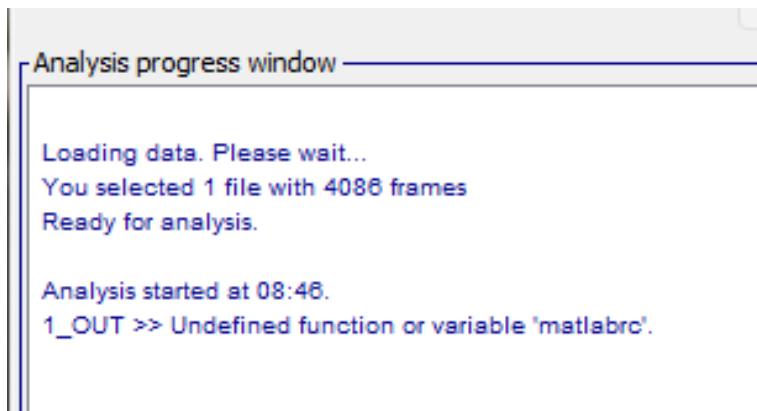


Figure 1.3: Undefined function or variable 'matlabrc' error

This error is known on the Matlab community (<https://fr.mathworks.com/matlabcentral/answers/98050-why-do-i-get-an-error-saying-undefined-function-or-variable-matlabrc-when-executing-a-program-th>).

The simple way to solve this issue is to:

- 1.1. Go to the MCR folder named "mcrCache9.2" located in a folder whose path is "C:\Users\YOURNAME\AppData\Local\Temp\YOURNAME":

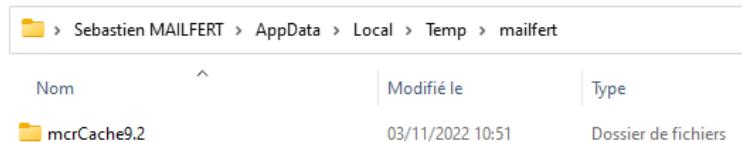


Figure 1.4: "Undefined function or variable 'matlabrc'" error

- 1.2. delete it.
2. Fiji is sometimes unable to read Nikon ND2 files correctly. If you encounter problems with this type of file, please convert them to tiff !!!

About UNLOC

2.1 What is UNLOC?

As an open-source and user-friendly Fiji plugin UNsupervised particle LOCalization (**UNLOC**) is designed for processing Single Molecule Localization Microscopy (**SMLM**) data. UNLOC is available for Windows, Linux and Mac OS (see Chapter 6)

UNLOC generates a list of single-molecule localizations with their respective precisions, allowing the reconstruction of super-resolution images. UNLOC is robust on data with high density of particles/frame, large differences of Signal to Noise Ratio (**SNR**) and variation of spatio-temporal background. UNLOC is well adapted to analyze stacks of images acquired at the same focal plane (2D), with performances close to the Cramér-Rao Bound (**CRB**) (see **Online Methods** and **Supplementary Information**).

As an effective unsupervised algorithm, UNLOC requires only the size of the point spread function (**PSF**) of the optical setup used for the data acquisition or for the generation of synthetic data.

The UNLOC plugin executes standalone compiled codes developed on Matlab/Octave (Fig. 2.1). The Matlab version of UNLOC does not require specific toolbox except the Parallel Computing Toolbox if speeding-up the execution of the analysis module is necessary.

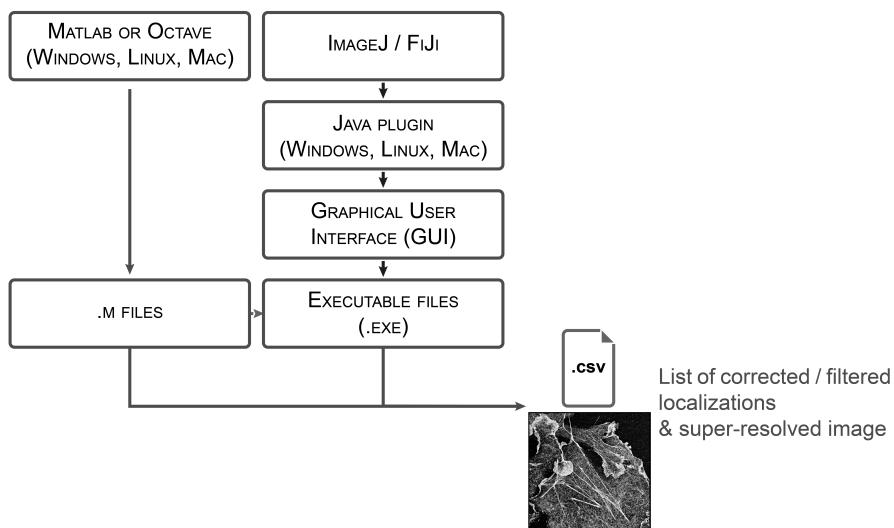


Figure 2.1: Schematic structure of UNLOC. Only the plugin part is developed in this manual

2.2 UNLOC modules

UNLOC is divided into four modules (Fig. 2.2):

Analysis - This module encompasses the detection and estimation steps. It provides a list of coordinates from any SMLM data such as direct STochastic Optical Reconstruction Microscopy (**dSTORM**), STochastic Optical Reconstruction Microscopy (**STORM**), Ground State Depletion microscopy followed by Individual Molecule return (**GSDIM**), Photo-Activated Localization Microscopy (**PALM**), Point accumulation in nanoscale topography (**PAINT**) and similar approaches based on single molecule localization.

Reconnection - It is designed to reconnect particles over time due to the blinking process occurring during data acquisition. This module is actually designed for static data but can easily be replaced by a tracking algorithm such as MTT (Sergé et al., Nat. Meth., 2008) for dynamic observations.

Rendering - This module integrates different optional functions (xy-drift correction, filtering of the list of localizations) before reconstructing a super-resolved image.

Batch - This module allows the analysis of multiple files in a pipeline.

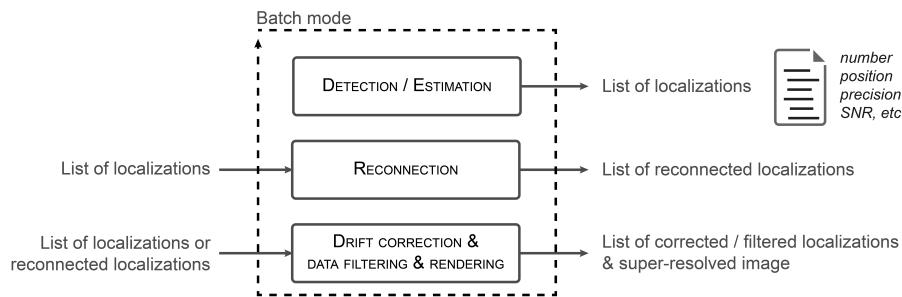


Figure 2.2: UNLOC modules and their respective inputs and outputs

2.3 Download and installation

In order to work with UNLOC, the following **free** software packages need to be installed:

- The **Fiji** open source image processing program. To download its latest version, visit its website here: [Fiji](#) [Schneider et al., Nat. Meth., 2012].

For help with Fiji, please consult its official website. Please check that your Fiji software and Java are up-to-date. In Fiji, the subdirectory "java" must contain a recent Java Development Kit (**JDK**) version (> 1.8).

- The **Matlab Compiler Runtime** (MCR 2024a, version 24.1) which corresponds to your operating system is available [here](#). Install the MCR following the Matlab instructions.

Always keep the default installation folder!

- The **UNLOC plugin** freely available [here](#).
- Fiji: copy/paste the UNLOC plugin ("UNLOC_2.51.jar" and the associated "UNLOC" folder) into the "plugins" folder of Fiji (for example C:\Fiji.app\plugins\). Allow if necessary file execution access to the files containing these patterns UNLOC_detect, UNLOC_reco and UNLOC_visu.
- The Matlab/Octave source code is available on request upon material transfer agreement (MTA).

2.4 Compatibility and system requirements

Disable the multi-thread option in the BIOS to optimize the speed of analysis.

UNLOC has been tested on the following operating systems:

- Windows 7 (SP1) / 8 / 8.1 / 10 / 11.
- OS X Yosemite (10.10) / El Capitan (10.11) / Sierra (10.12).
- Linux Ubuntu (≥ 14.04) / Mint (≥ 17).

In practice, the UNLOC plugin can be used on any computer on which the MCR 2017a, version 9.2, 64-bit is running.

Since the processing of SMLM data used to be computationally demanding, we recommend to use a computer with high amount of RAM (at least 8 GB) and a processor with a high number of cores. For instance, UNLOC has been tested on a DELL Precision T1700, Intel Core i7-4770 CPU at 3.4 GHz, with 64 GB with Windows 10 (64-bit).

The heuristic used in UNLOC for High Density ([HD](#)) data is not compatible with GPU computation. Thus, optimizing the speed of the analysis is obtained by parallel computing.

2.5 Evaluation of the performance of UNLOC on synthetic data

The two evaluation charts described in the **Online Methods** are included in the **Supplementary Software** package:

- **The alphanumeric character chart** reproduces the density-SNR space diagram with the alphanumeric characters resolved for SNR and local density ranging from 20 to 40 dB and 0.1 to 5 particles/ μm^2 , respectively.
- **The inter-particle distance chart** mimics a two-particle canonical scenario at different SNR ratios (from 20 to 30 dB), at variable inter-particle distances ranging from 0.5 to 6 r_0 .

We provide in the **Supplementary Software** package two Windows executable files (`Inter-particle-distance_Chart.exe` and `SNR_Density_Diag_Chart.exe`) allowing the generation of these two types of synthetic data. A standard random generator seed was used to generate strictly identical synthetic image stacks at each run.

It is recommended to use these two charts to test that the UNLOC plugin works on your systems.

2.6 Help with UNLOC

At first, please consult this user's manual to get general principles of the UNLOC modules in the "Quick start guide" chapter 3, and for a more detailed description of the modules, the corresponding "Detailed UNLOC description" chapter 4.

You can also get on-line help at any moment by hovering over items with the mouse (tooltips).

2.7 UNLOC updating and bug reporting

The present version of UNLOC will be updated in the future. Download the latest UNLOC version on [our website](#).

Prior contacting us for possible errors (bugs), please upgrade for the latest version of the software package. We also encourage you to check that your computer follows the compatibility and system requirements defined above.

If malfunction still persists, please email us at our [UNLOC email address](#). We will try to do our best to answer your question.

Quick start

This part covers the basics of UNLOC, in particular, how to launch UNLOC and perform analysis without modifying the default parameters except the ones characteristic of the experimental setup.

Note that the default parameters of UNLOC are set for processing Total Internal Reflection Fluorescence Microscopy ([TIRFM](#)) SMLM images, for high-density (HD) data, low spatio-temporal frequency background variations, a PSF size of 1.25 pixels and automatic camera parameter evaluation.

We recommend to be accustomed to UNLOC by running the first analysis on ones of the synthetic data charts included the **Supplementary Software** package (see section [2.5](#) for details). Thus, you can test the Analysis and Rendering modules except the Reconnection module. In that case, the *Reconnection* module is useless since the particles are not allowed to blink.

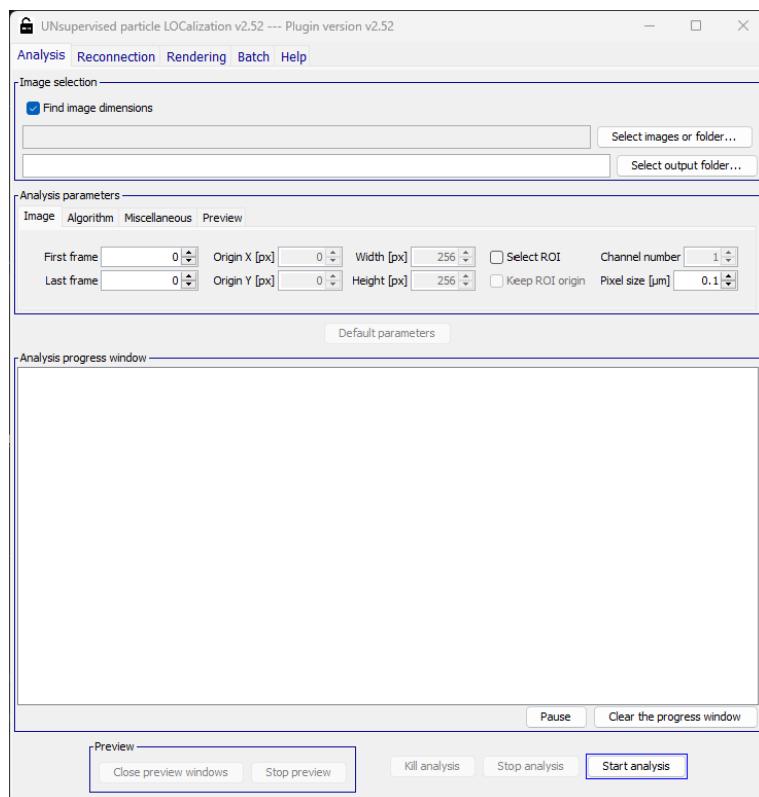


Figure 3.1: General window

3.1 Data loading

1. Start Fiji
2. From the *Plugins* menu, select *UNLOC* to get access to the single window providing the four sequential steps and to the Help tab (see Fig. 3.1):
 - Analysis
 - Reconnection
 - Rendering
 - Batch
 - Help

3.1.1 Tiff, nd2 and raw files

- UNLOC supports the TIFF file format. It can load file ending with .tif, .tiff or .stk extension (case insensitively)
- UNLOC supports the Nikon ND2 file format
- UNLOC supports the PCO raw data file format
- Stack, multiple stack (with same size only) and multiple one frame image are supported.

- From the *Analysis* menu, in the *Image selection* box, click on *Select images of folder...*
- You can drag and drop files or folder directly on the *Analysis* tab
- On the file dialog, locate and highlight the file or folder, and click *Open*.
- By default, the results are saved in an output folder created within the one containing the data. Otherwise, you can click on *Select output folder...* to choose another folder at your best convenience.
- A unique folder whose name is based on the date will be automatically created in this folder (UNLOC_yymmdd_hhmmss_detectionmode).

3.1.2 Multiple files or stacks

In case of multiple stack or one single frame series the name of each file must have exactly the same length. For example, an image series with 11 files must match the following pattern: Cell00, Cell01, ..., Cell10.

Be careful, in this case (and with image series only) the first frame number must be 0 and the last must be 10. In the image selection field, the name of the image sequence will be: 'Cell%.2d.tif'. By selecting the *Find image dimensions* (*Image selection* box), all of these fields will be automatically filled.

3.2 Analysis

This module encompasses the detection and estimation steps. It provides a list of coordinates from any SMLM data.

1. From the *Analysis* menu, select the *Image tab* and set the range of frame to analyze.

2. Then, select the *Algorithm* tab.
3. Set in the *PSF model* box, the size of the PSF of your experimental setup. The value corresponds to the radius expressed in pixels.
4. Click the *Start analysis* to launch the analysis.
5. The *Analysis progress window* details the progression of the work and indicates the time at which the processing ended.
6. An **Analysis** file (.csv) is automatically created and saved. This file includes the information on the parameters used for the analysis and a list of detected particles with: the frame number in which the particle is detected (time), the i and j particle coordinates (position_i and position_j, respectively), the RMS precision (std_error_position_ij), the intensity value (alpha) and the standard deviation on the background (std_bg_noise).

3.3 Reconnection

This module (see section 4.2) is designed to reconnect particles over time due to the potential blinking process occurring during data acquisition. The **Reconnection** is actually designed for static data but can easily be replaced by a tracking algorithm such as Multiple Target Tracing (MTT) (Sergé et al., Nat. Meth., 2008) for dynamic observations.

1. From the *Reconnection* menu, click on *Select file...* to open a file dialog, locate and highlight the file (.csv) and click *Open*. By default, the file dialog opens the folder created in the *Analysis* menu. You can drag and drop a .csv file directly on the *Reconnection* tab.
2. Click *Start reconnection* to launch the **Reconnection**.
3. The *Reconnection progress window* details the progression of the work and indicates the time at which the processing ended.
4. A **Reconnection** file (.csv) is automatically created in a new folder whose name is based on this template: **Reconnect_yymmdd_hhmmss**. This file includes the information on the parameters used for the **reconnection** and a list of trajectories with: the trajectory number (num_traj), the frame at which the trajectory is turned-off (time_end), the first frame of the trajectory (time_start), the i and j averaged value particle coordinates (position_i and position_j), the RMS precision (std_error_position_ij), the number of frames where the particle is emitting signal (sum_on), the number of disappearance of a particle (nb_blink) and the sum of the intensity values (alpha_sum).

3.4 Rendering

This module integrates different functions (xy-drift correction, filtering of the list of localizations) before reconstructing a super-resolution image.

1. From the *Rendering* menu, click on *Select file...* to open a file dialog, locate and highlight the file (.csv) and click *Open*. By default, the file dialog opens the folder created in the **Reconnection** menu. You can drag and drop a .csv file directly on the *Rendering* tab.
2. Click *Start rendering* to launch the rendering.
3. The *Rendering progress window* details the progression of the work and indicates the time at which the processing ended.

4. A visualization folder is automatically created whose name is based on this template: `Visu_date_time` and saved in the same folder. The file includes the reconstructed images generated in five different modes (see Fig. 4.6):

- BINARY - a binary mode that represents each localization as a white sub-pixel in the reconstructed image.
- INTEGRATED BINARY - an integrated binary mode which the intensity values assigned to each sub-pixel correspond to the number of localizations within each pixel.
- TIME - a time mode encoded in a blue to red look-up table as a function of the first temporal appearance of the particle.
- GAUSSIAN - a mode where each particle is represented by a Gaussian whose variance reflects the localization precision.
- INTEGRATED GAUSSIAN - an integrated Gaussian mode similar to the previous mode and accounting for the local density.

3.5 Batch

This module allows launching multiple files in a pipeline.

1. Load the data to analyze by clicking on *Select images or folder* for the first condition.
2. Set the values in the *Analysis parameters* box as in the 3.2 section.
3. Set the **Reconnection** parameters (optional).
4. Set the **Rendering** parameters (optional).
5. Click *Add to batch* to add a new job to the batch. All the jobs will be displayed in the *Batch* box.
 - to add another job, repeat steps 1-5
 - to cancel one job, select it and press *Delete selection*.
 - to remove all jobs, press *Clear all jobs*.
 - Click *Start batch* to launch sequentially all selected jobs. Each job analysis will be stored independently in individual data folder. Note: Click *Stop batch* to end all job analyzes.

3.6 Help

The *Help* window acknowledging the institutional fundings supporting the development of UNLOC provides useful links for:

1. The present UNLOC user's manual
2. The website to download the latest released version of the UNLOC plugin
3. The licence file
4. The email contact for further information

4

Detailed description of UNLOC

For the *Analysis* and *Rendering* modules, the functions of UNLOC are detailed separately for beginners and experts. We recommend to modify only the expert parameters by qualified persons.

4.1 Analysis



This module is at the core of UNLOC. As mentioned in the **Quick start** (see chapter 3), UNLOC requires only to set the PSF size. This unsupervised strategy does not require deep user expertise but aims at analyzing any kind of standard SMLM data meaning data with a maximum local density of 1 to 2 particles/ μm^2 /frame, a signal-to-noise-ratio (SNR) higher than 20 dB, no saturation and low spatio-temporal background variations.

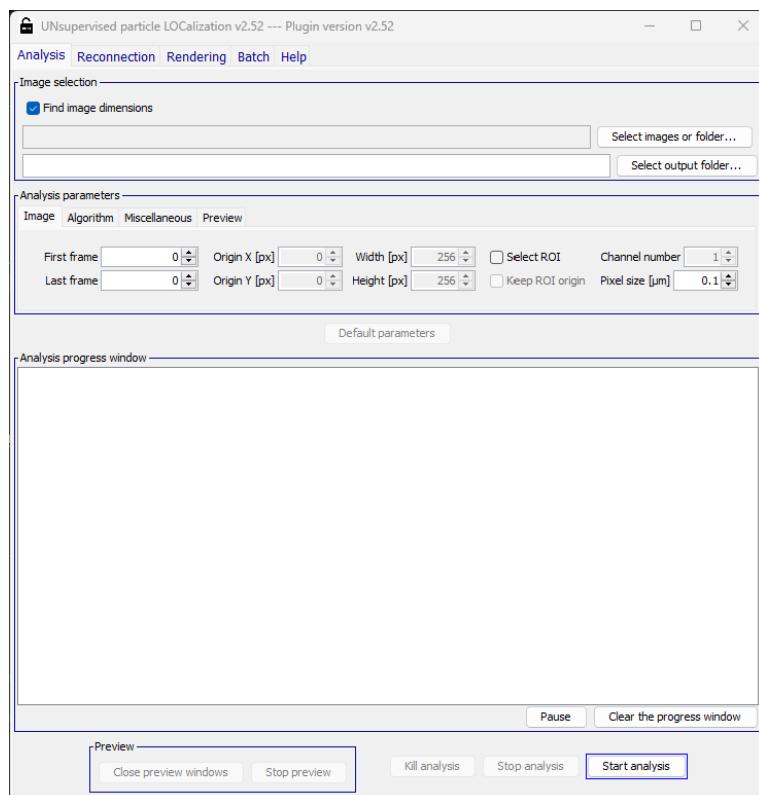


Figure 4.1: Analysis window

4.1.1 For beginners

1. From the *Analysis* menu, in the *Image selection* box click on *Select images of folder...*
 2. On the file dialog, locate and highlight the file or folder, and click *Open*. You can also drag and drop a file or folder directly in the window.
- Any data in .tif, .stk or .nd2 formats are recognized by UNLOC. By default, the results are saved in an output folder created within the one containing the data. Otherwise, you can open the *Select output folder...* to choose another folder at your best convenience. A unique folder whose name is based on the date will be automatically created in this folder (UNLOC_yymmdd_hhmmss_detectionmode).
3. Define the range of frames where the analysis is performed on in the *First frame / last frame* fields.
 4. Select the *Algorithm tab*. Set the "*Detection mode*" corresponding to the density of your data.

Choose Low Density (**LD**) if the density is lower than 0.1 part/ μm^2 , Medium Density (**MD**) if the density is lower than 0.5 part/ μm^2 or **HD** if the density is lower than 2 part/ μm^2 . Data with higher densities will provide erroneous and biased localizations.

5. Set in the *PSF model* box, the size of the PSF of your experimental setup. The value corresponds to the radius expressed in pixels.
6. Click the *Start analysis* to launch the **analysis** (*Stop analysis* and *Kill analysis* stops and kills (violent stop of the process) respectively the **analysis** process).
7. Once the analysis is started, you can follow in real-time its progress in the progress window.
8. An **Analysis** file (.csv) is automatically created and saved. This file includes the information on the parameters used for the analysis and a list of detected particles with the frame number in which the particle is detected (time), the i and j particle coordinates (position_i and position_j, respectively), the RMS precision (std_error_position_ij), the intensity value (alpha) and the standard deviation on the background (std_bg_noise).

4.1.2 For experts

Image parameters tab

- The selection of files/folders does not in practice load the data but only read the header of the files. The effective data loading is done by UNLOC during the analysis. Consequently, there is **no limitation** in terms of file size and number of files. For instance, we have performed UNLOC analyzes without problem on 65,000 images of 512×512 pixels which correspond to 32 GB.
- In the file chooser, you can copy and paste a directory path and keep **Ctrl** key pressed before clicking on the **Open** button. Thus, you can enter in the directory without loading its content.
- You can set a region of interest (ROI) *Select ROI / origin X / origin Y / width / height* to limit the analysis window to a Region Of Interest (**ROI**). When the *Select ROI* checkbox is activated, the first frame of the selected data is displayed where you can draw a ROI (minimal size: 16×16 pixels).
- Select *Keep ROI origin* to keep the original coordinates (otherwise resetting them to to (0,0)).
- Set the physical *Pixel size* of the raw data in μm .

Algorithm tab

- For the *Medium density* and *High density* detection modes only, we defined different possibilities to evaluate the background: *Low spatial frequency* (default, for classical experimental data) or *High spatial frequency* (for low quality illumination for example).
- Select the suitable values of your instrument PSF and the Gaussian mode (*Integrated gaussian* represents at best the real PSF imaged on a camera). The PSF size can be evaluated as follows:

$$\text{FWHM} = 2 \times r_0 \sqrt{2 \ln(2)}$$

or

$$r_0 = \frac{\text{FWHM}}{2\sqrt{2 \ln(2)}} \approx 0.42 \times \text{FWHM}.$$

- Select the TIRF checkbox if you acquired data in this mode. Otherwise, UNLOC evaluates the PSF size automatically. This is less efficient and slows down the analysis. UNLOC ensures high localization performances if data are acquired in 2D. A slight dye de-focus can be taken into account but not effective 3D data.

Localization error estimation tab

In order to properly evaluate the precision of localization, the camera parameters should be appropriately set or automatically evaluated by the algorithm.

- *Additive Gaussian mode* evaluates automatically the noise model of the Electron Multiplying Charge Coupled Device ([EMCCD](#)) or scientific Complementary Metal-Oxide-Semiconductor ([sCMOS](#)) camera.
- Otherwise, a signal-dependent Gaussian noise model is used. It needs the *EM gain* (different from the electron multiplying charge coupled device (EMCCD)), *Baseline level* (the gray level of the image without signal, typically set at 100 for Andor EMCCD cameras) and the *Amplification*. To let UNLOC determining automatically the *EM gain*, select the *Additive Gaussian noise* checkbox.

Miscellaneous tab

- Set *Parallel computing* to 0 if you want to use UNLOC as a mono-thread process (slower but starts faster), 1 to use the maximum number of cores of your processor (faster) and *n* to use *n* cores with $n \leq$ number of available cores.
- Clicking on *Reuse parameters* allows to upload the parameters set in a previous analysis by selecting the corresponding .csv file.

Preview tab

It is possible to visualize intermediate result of the analysis before the end of the process. We want to point out that the two modes displaying images and statistics increase the analysis duration. The results displayed are less accurate than the real results enabling a gain in speed.

- There are three options:
 1. *None*: default, faster mode, no preview.
 2. *Preview*: slow, to visualize localizations on the raw data, an intermediate binary image of the localizations and useful statistics, see Fig. 4.2.
 3. *Debug mode*: really slow, to visualize localizations on the raw data without the background (in HD mode), the deflation image corresponding to the raw data without the localized particles, an intermediate binary image of the localizations and useful statistics, see Fig. 4.3. The debug mode should be used only on small data and small number of frames (e.g. 256×256 pixels, 500 frames).
- *Image zoom factor* sets the sub-pixel zoom factor which increases the amount of resources needed (1: corresponds to the original data, > 1 : increases the resolution, 3: maximum) but improves the visual quality.
- Set the *Cursor size* of the ones displayed on the localization image (only available for the *Fixed radius circles* option).
- The *Precision coefficient* is the ratio of the circle whose radius is proportional to the precision of localization (only if the *variable radius circles* option is selected).
- Three *Cursor shapes* are available:
 1. *Variable radius circles* where each localization is displayed as a circle centered on the particle and whose radius is proportional to its precision of localization. Thus, the highest is the precision, the smallest circle is.
 2. *Crosshairs* where each localization is displayed as a cross-hair centered on the particle.
 3. *Fixed radius circles* where each localization is displayed as a circle whose radius is based on the *Cursor size*.
- To close or stop the preview visualization but without aborting the analysis process, click on *Close preview windows* or *Stop preview*.

Don't use the classical "zoom" tool of Fiji. Just click on the preview images to zoom into your image. You can zoom out by right-click and restore the default zoom by middle-click on Windows Preview

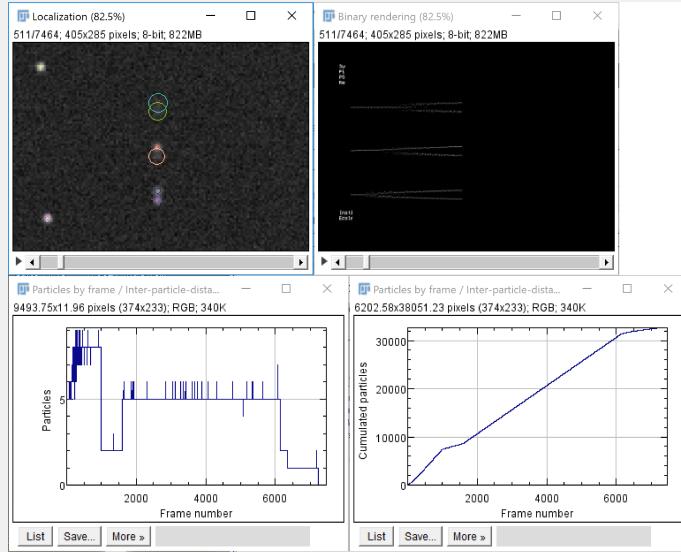


Figure 4.2: Analysis - Preview mode - (1) The localization of each detected particle on the raw data (visual simplification for gain of speed), (2) an integrated binary live rendering image (this is not the final result, be careful), (3) graphs of the number of detected particles per frame.

Debug mode

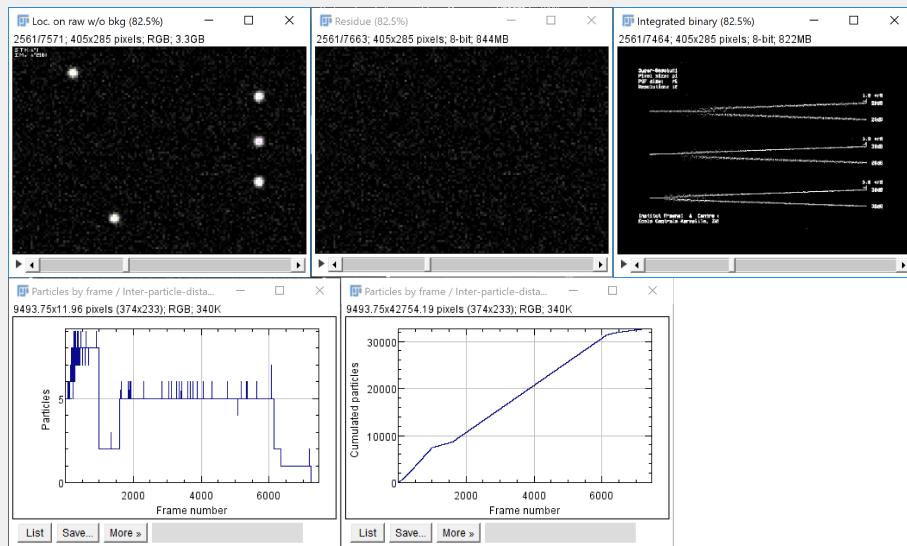


Figure 4.3: Analysis - Debug mode - (1) The localization of each detected particle (on the raw data without the background map), (2) the residues, (3) an integrated binary live rendering image (this is not the final result, be careful), (4) graphs of the number of detected particles per frame.

4.2 Reconnection



The **Reconnection** module determines statistically (based on the position and precision of localization) the possibility that the same fluorescent molecule has been seen on subsequent frames. For instance, in dSTORM, dyes can blink: disappear ("OFF-state") during few frames and reappear ("ON-state"). **Reconnection** is based on the assumption that the observations are static, i.e. the dyes are not moving. We define as "trajectory" the multiple localizations of the same dye over time. To track dynamic data(Single Particle Tracking (**SPT**), SPT-PALM), use the analysis file in a dedicated algorithm (e.g. Sergé et al., 2008).

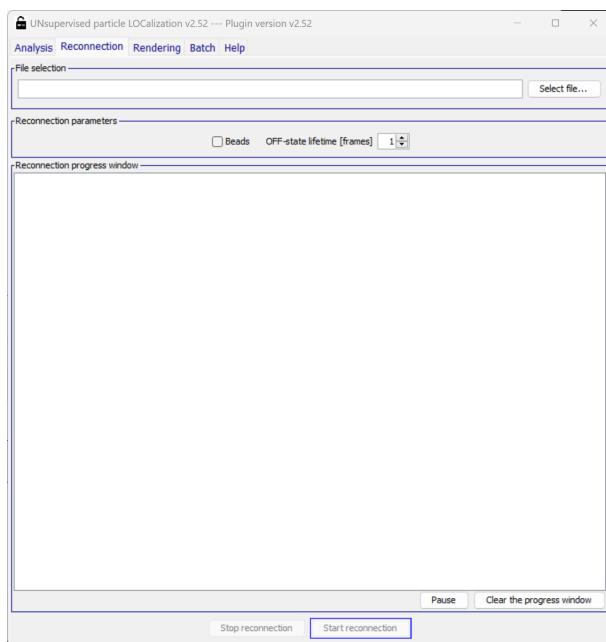


Figure 4.4: Reconnection window

1. From the *Reconnection* menu, in the *File selection* box click on *Select file...*
2. On the file dialog, locate and highlight an UNLOC .csv file, and click *Open*. You can also drag and drop a .csv file directly in the window.
3. Select the *Beads* checkbox option if fiduciary markers are present in your images. This will help the **reconnection** process and the future optional xy-drift correction.
4. Set the *OFF-state lifetime* number of frames that corresponds to the maximum successive number of frames during which a particle can be switched off before it is considered as finally extinguished. For example, a value ≈ 3 can lead to biased reconnection.
5. Click *Start reconnection* to launch the **Reconnection**.
6. The *Reconnection progress* window details the progression of the work.
7. A **Reconnection** file (.csv) is automatically created in a new folder whose name is based on this template: **Reconnect_yymmdd_hhmmss**. This file includes the information on the parameter values used for the **reconnection** and a list of trajectories with: the trajectory number (num_traj), the frame at which the trajectory is turned-off (time_end), the first frame of the trajectory (time_start), the i and j averaged particle coordinates (position_i and position_j), the RMS precision (std_error_position_ij), the number of frames where the particle is emitting signal (sum_on), the number of disappearance of a particle (nb_blink) and the sum of the intensity values (alpha_sum).

4.3 Rendering



The **Rendering** module is dedicated to data filtering (optional), xy-drift correction (optional) and rendering. *Note: only the rendered images obtained through this module are valid unlike the ones obtained during the "preview" mode of the analysis (see section 4.1.2).*

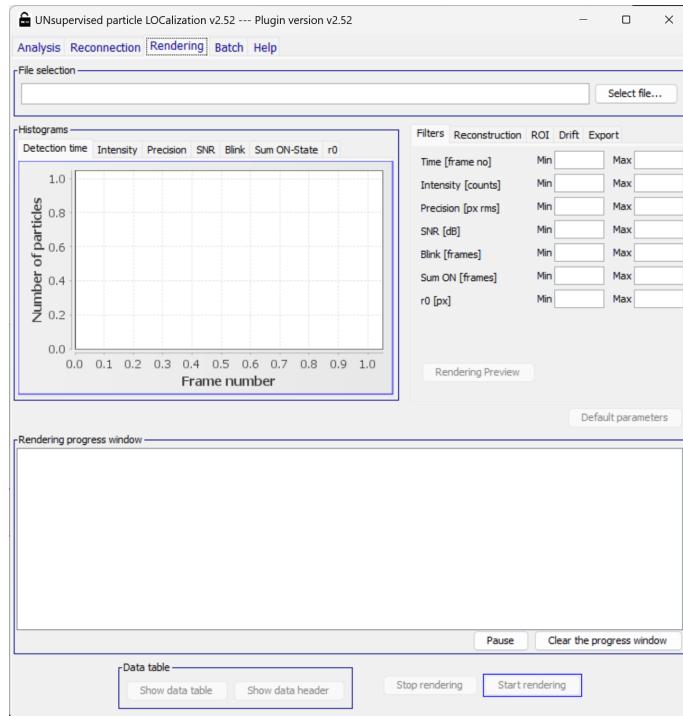


Figure 4.5: Rendering window

4.3.1 For beginners

1. From the *Rendering* menu, in the *File selection* box, click on *Select file...*
2. On the file dialog, locate and highlight an UNLOC .csv file, and click *Open*. You can also drag and drop a .csv file directly in the window.
3. In the *Histograms* box, five tabs display different histograms for the loaded data. They are detailed in the next 4.3.2 section.
4. In the *Reconstruction* tab, set the *Rendering zoom* factor and the *Pixel size* for the super-resolution image. The *Rendering zoom* factor corresponds to the ratio between the raw data pixel size and the reconstructed image (see the 4.3.2 section). The *Pixel size* in μm should correspond to the physical pixel size of the raw data and is used to scale properly the reconstructed image.
5. Set in the *Drift* tab if you want to perform xy-drift correction (see the 4.3.2 section).
6. Click on *Start rendering* to launch the UNLOC rendering.
7. The *Rendering progress window* details the progression of the work.
8. A visualization folder is automatically created whose name is based on this template: **Visu_date_time** and saved in the same folder. The file includes the reconstructed images generated in five different modes (see the 4.3.2 section).

4.3.2 For experts

Histograms box

The histograms corresponding to the loaded data are displayed:

- *Detection time*, i.e. the distribution of the number of particles detected per frame.
- *Intensity*, i.e. the distribution of the intensity of each particle.
- *Precision*, i.e. the distribution of the RMS precision of localization (σ_{XYRMS}) in pixels (the value in the .csv output file is expressed in pixels).
- *SNR*, i.e. the distribution of the signal-to-noise-ratio of the particles (*Note: the SNR is only available for non-reconnected data. As the SNR is calculated from the knowledge of the local background, this is difficult to calculate on "trajectories"*). SNR can be calculated from the alpha and std_bg_noise of the .csv file as:

$$SNR(dB) = 20 \times \log_{10} \frac{\text{alpha}}{\text{std_bg_noise}}$$

- *Blink*, i.e. the distribution of the number of times (falling edge) where the particle disappear during its "trajectory" (*only available if for reconnected data*).
- *Sum ON state*, i.e. the distribution of the number of frames where the particle has been seen (only when it was ON, not OFF, it is different from the length of the "trajectory") (*only available for reconnected data*).

The precision histogram is expressed both in pixels and in nanometers.

Conversion of the precision from pixels to nm

$$\begin{aligned} \text{Precision in nm} &= \sqrt{2} \times \text{RMS precision} \times \text{Pixel size } (\mu\text{m}) \times 1000 \\ &= \sqrt{2} \times 0.18 \times 0.107 \times 1000 \\ &\approx 27 \text{ nm}. \end{aligned} \tag{4.1}$$

This value is equal for example to 27 nm for a precision of 0.18 pixels. The $\sqrt{2}$ factor comes from the conversion of RMS value (σ_{XYRMS}) to a value for one axis (σ_X).

Filters tab

You can set the range (minimum and maximum values) for the different parameters stated above. This will update automatically the histograms and the image rendering. When you change a value, press "enter" to validate this new value. Click on *Rendering preview* to display an integrated binary rendering preview of the raw data and the filtered one.

Reconstruction tab

- The *Rendering zoom factor* sets the ratio of the original data pixel size and the rendered one.

Rendering zoom factor?

Here is an example for a total optical magnification of 150, a camera *Pixel size* of $16\mu m$ and a *Rendering zoom* factor of 8:

$$\begin{aligned}\text{Rendered pixel size} &= \frac{\text{Camera pixel size}}{\text{System magnification}} \times \frac{1}{\text{Rendering zoom factor}} \\ &= \frac{16\mu m}{100 \times 1.5} \times \frac{1}{8} \\ &\approx 0.107 \times \frac{1}{8} \\ &\approx 13.4 \text{ nm.}\end{aligned}\tag{4.2}$$

This value is equal to 10.7 nm for a *Rendering zoom* factor of 10. We set the *Rendering zoom* factor to 8 considering the localization precision obtained with our experimental data.

- Pixel size* sets the original data pixel size before rendering. This is to scale directly the rendered images in μm
- Gaussian filter size* sets a smoothing coefficient (> 0.4) to improve the rendered image to avoid high intensity peaks
- Select *Rescale images* to rescale the rendered images to 16 bits (only for the "Integrated binary" and "Time" rendering modes)

ROI tab

- Click on *Select ROI (Final rendering only)* to render only a part of the data into a ROI.
- The ROI parameters are set by *Origin XY, width and height*.

Drift tab

A xy-drift which can occur during the acquisition can be corrected if needed by two means: correlation (if structures are present) or fiduciary markers (typ. beads):

- Correlation* means that your data does not contain fiduciary markers. The correction is done by correlation. Internally, intermediate super-resolved images are built by subsets of *Correlation time window* frames and correlated to each other. Set *Correlation time window* corresponding to the number of frames per subset used to build intermediate super-resolved images before correction. The default value is arbitrary set in the plugin to 1,500 but it should be in accordance to your data.
- Beads* means that your data contains fiduciary markers. If you choose this mode, don't forget to be sure that you already specified that beads were present during the reconnection process. Uncheck *Keep beads* to remove the bead localizations on the rendered images. Select *Correct drift only on ROI* if you want to correct the drift only on the ROI.

Export tab

- Select *Save rendered images* (default) to save rendered images as .tif files. If you choose to save the rendered images, you can save them individually or as a stack of multiple frames. A visualization folder is automatically created whose name is based on this template: `Visu_date_time` and saved in the same folder.
The five output rendering modes are (see Fig. 4.6 and section 3.4).
 - *Supplementary data output: None / Before filters / After filters*: you have the possibility to save drift-corrected and/or filtered and/or rendered data as .csv files. This allows you to use the resulting data in an other rendering application. Saving a file without drift correction before filtering is discarded because such file is identical to the input file!

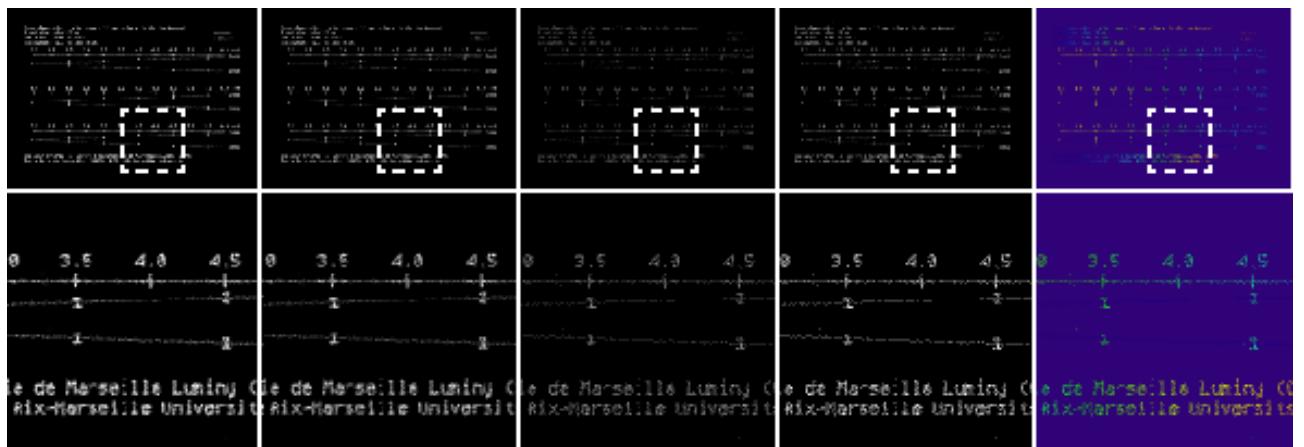


Figure 4.6: Five modes of rendering

Data table box

You can display directly from the plugin a table containing the data (filtered if filters are applied) by clicking on *Show data table* or the header of the .csv file used for the rendering by clicking on *Show data header*.

4.4 Batch



This mode is dedicated to `analyze`, `reconnect` (optional) and `render` (optional) multiple experiments analyzed sequentially.

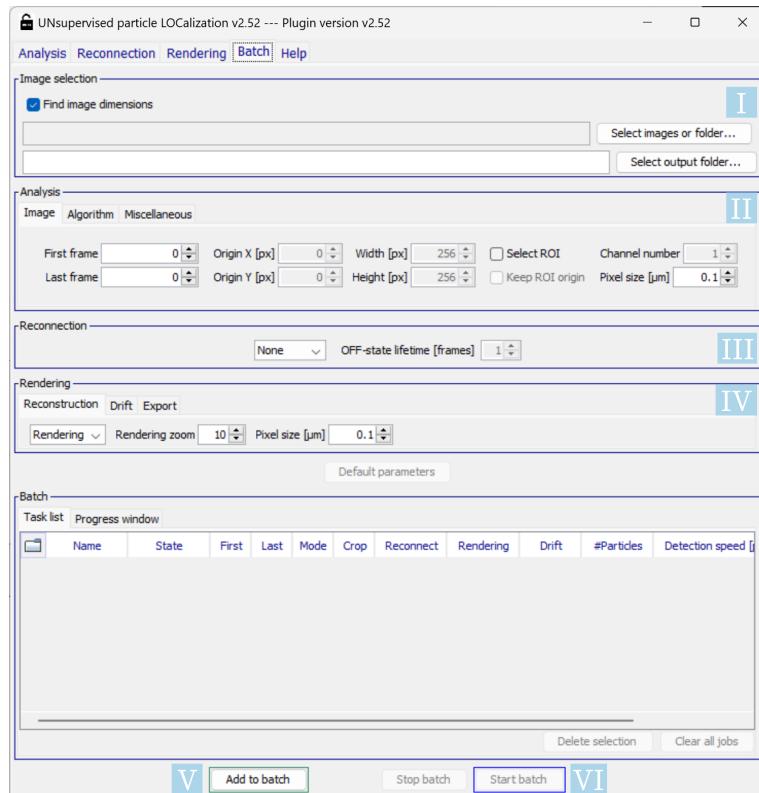


Figure 4.7: Batch - General window

4.4.1 Module specifications

Most of the `Batch` parameters have been already explained in the previous sections.

- ▷ **Analysis** - The `Analysis`, which is identical to the *Analysis* tab, is not optional. There is no preview mode.
- ▷ **Reconnection** - The `Reconnection`, which is identical to the *Reconnection* tab, is optional.
- ▷ **Rendering** - The `Rendering` box provides access to the same set of functions described previously except that filtering is not possible. The filtering which requires careful inspection of the data has to be performed independently.

4.4.2 How does it work?

As all of the parameters have been already detailed, we will just explain here the major steps.

- I Load the data to analyze by clicking on "**Select images or folder**" for the first condition
- II Set the "**Analysis**" parameters (mandatory)
- III Set the "**Reconnection**" parameters (optional)
- IV Set the "**Rendering**" parameters (optional)
- V Press the "**Add to batch**" button to add a job to the batch (displayed in the "**Task list**")
 - to add another job, repeat steps 1-5
 - to cancel one job, select it and press "**Delete selection**"
 - to remove all jobs, press "**Clear all jobs**"
- VI When everything is correct, press "**Start batch**". To stop the batch, press "**Stop batch**". Be careful, this will stop all jobs!

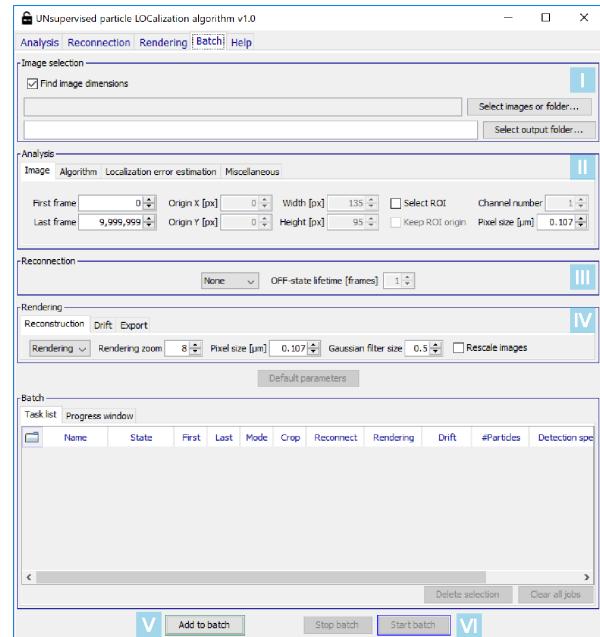


Figure 4.8: Batch - Steps to follow

Step VI will automatically launch all n jobs sequentially, the results will be stored independently in each individual data folder.

The current state of each job is displayed in the "**Task list**" and in the "**Progress window**". When the analysis is done, you can open directly the output directory by clicking on the left part of the "**Task list**". If an error occurs during a specific job, it will be indicated in the "State" column.

5

PSF size: r_0

As you know now, UNLOC only requires one physical parameter of the optical setup: (r_0) the PSF size to detect and localize particles.

r_0 only depends on the optical properties of the microscope: the wavelength, the numerical aperture of the objective and the total magnification of the system. While you don't change the dye, the objective nor the camera, r_0 does not change from experiment to experiment. Once sets, you can use it as a fixed parameter.

5.1 How to determine r_0 ?

5.1.1 How to choose between TIRF vs. non TIRF options?

There are two options in the Analysis tab:

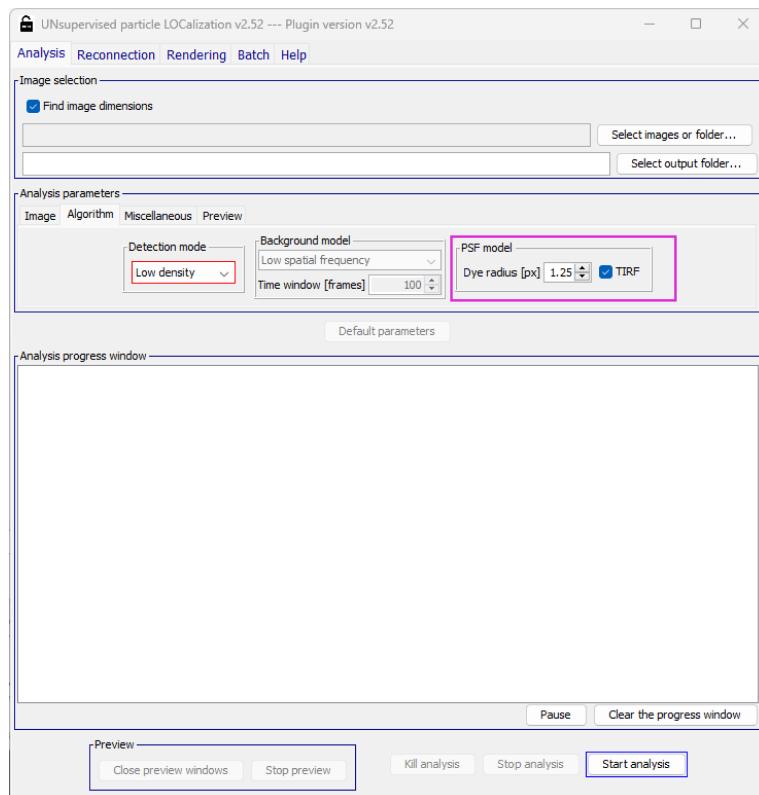


Figure 5.1: TIRF option

1. TIRF

Selecting the TIRF checkbox forces UNLOC to detect PSF with a specific r_0 size. If you are in a pure 2D experimental case (such as with a TIRF illumination), this mode has to be used. It allows to increase the density of particles to detect because UNLOC does not have to estimate the size of the PSF.

2. Non-TIRF

If the TIRF checkbox is not selected, UNLOC will estimate the PSF size of all the particles. When you have a situation with particles with different sizes (all the PSF are not at the focal plane), this mode has to be considered. Unfortunately, the density of particles detectable by UNLOC will decrease as a degree of liberty is added to the estimation model.

5.1.2 Determine r_0 for your optical setup

As r_0 is the only parameter to set in UNLOC, it has to be set properly if you use the TIRF mode (see [5.1.1](#)). To determine the r_0 value, please follow these few steps, summarized in the [5.2](#) figure:

- a) Acquire a stack of images containing dyes with a sufficient SNR (25-35dB) at low density and as much as possible in 2D in order to have PSFs with similar sizes. The more dyes and images you have, the more accurate the r_0 determination will be
- b) Run UNLOC with the LD and non-TIRF modes
- c) Open the .csv output file and calculate the average of the r_0 column
- d) Use now UNLOC with the TIRF mode and set the r_0 value

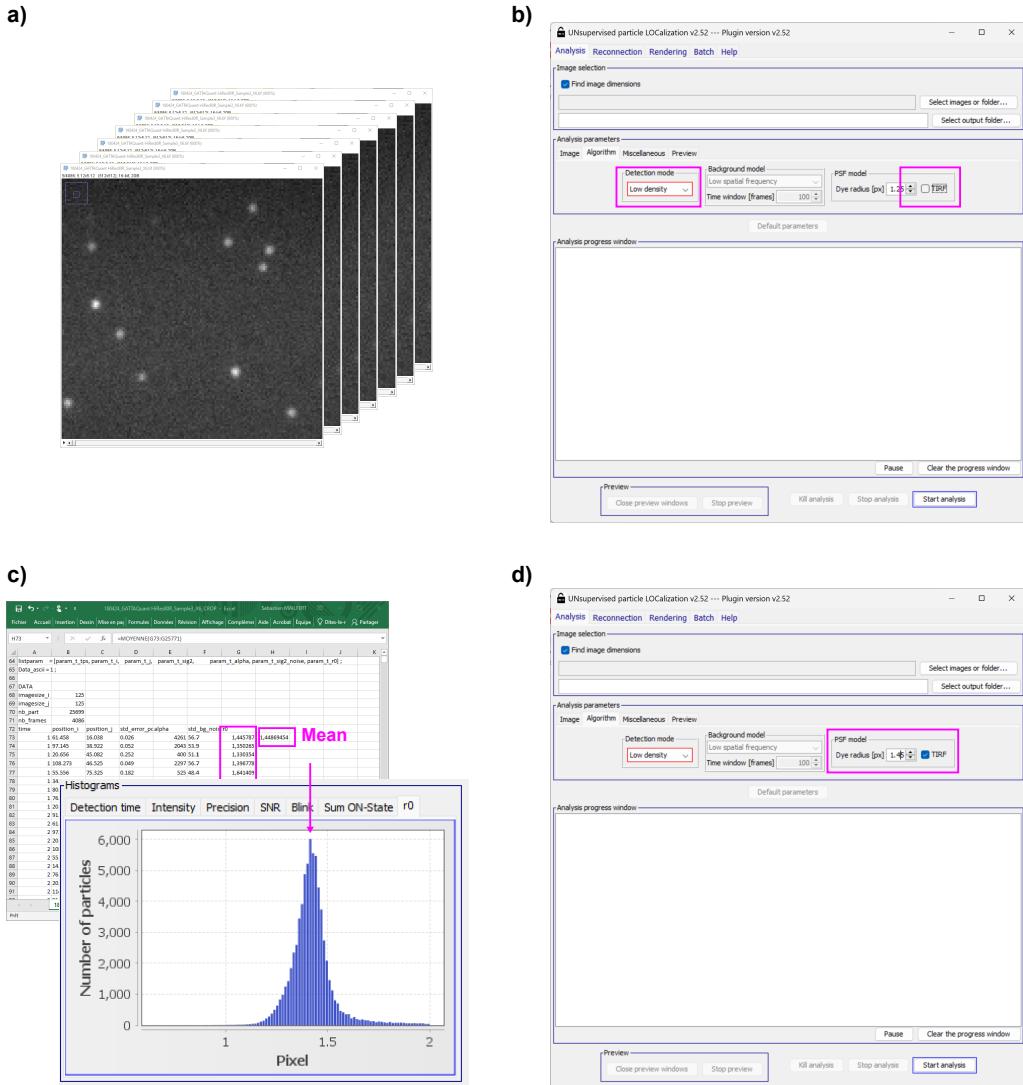


Figure 5.2: Steps to determine the r_0 value of your setup. a) A set of data with isolated (low density) and bright (high SNR) PSF is required. b) Analysis with the non-TIRF option. b) Look at the r_0 histogram or its value in the .csv file. d) Use this value of r_0 with the TIRF option in UNLOC (if your data have been acquired in a TIRF mode)

5.2 How to display r_0 in the Analysis .csv output file?

5.2.1 Check if r_0 is written in the .csv output file

If you downloaded UNLOC after November 2022, r_0 is automatically written in the .csv output file after the **Analysis**.

The .csv file if r_0 is written by opening it and look into the columns of the parameters obtained on the detected particles :

A	B	C	D	E	F	G
67	DATA					
68	imagesize_i	95				
69	imagesize_j	135				
70	nb_part	31258				
71	nb_frames	7840				
72	time	position_i	position_j	std_error_pcalpha	std_bg_nois	r0
73	1	11.926	22.284	0.031	1396	24.2
74	1	80.915	36.988	0.031	1412	25.0
75	1	22.786	121.587	0.029	1425	23.5
76	1	42.745	121.539	0.031	1393	24.0
77	1	62.759	121.647	0.034	1314	25.3
78	2	18.528	37.056	0.029	1427	23.9
79	2	78.361	60.642	0.031	1371	23.8
80	2	23.904	80.611	0.032	1447	26.3
81	2	49.522	80.750	0.031	1444	25.6

Figure 5.3: r_0 as column in the output file

NOTE: Depending on the TIRF mode (checked or not, see [see 5.1.1](#)), the r_0 is constant (TIRF) or variable (TIRF unchecked).

5.2.2 What should I do if r_0 is not written in the output file?

If the r_0 column does not appear, you can (1) download the new version of the plugin [here](#) or (2) edit a file as follows:

1. Close Fiji
2. Go to the "UNLOC" folder, located into the "plugin" folder of Fiji:

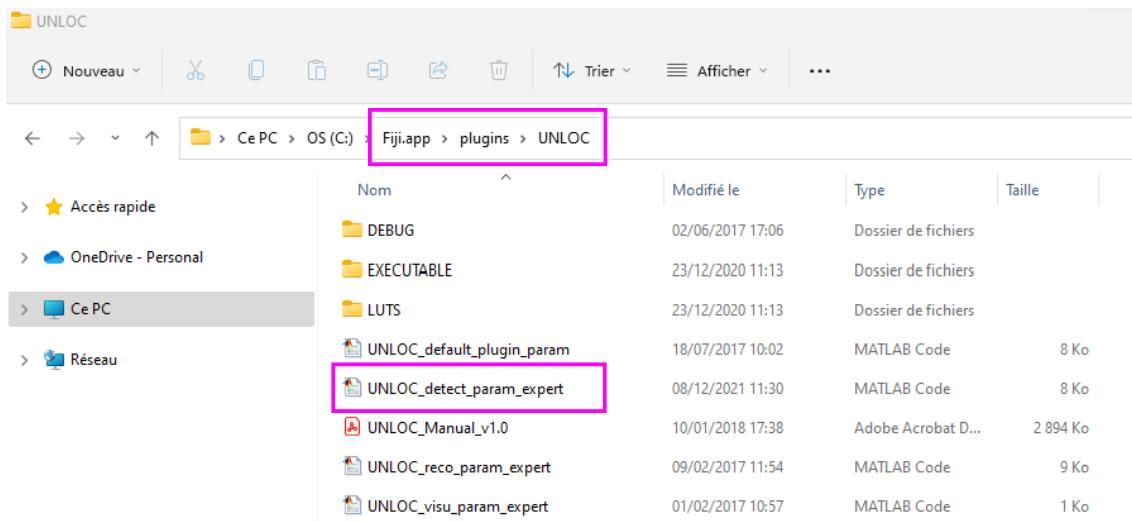
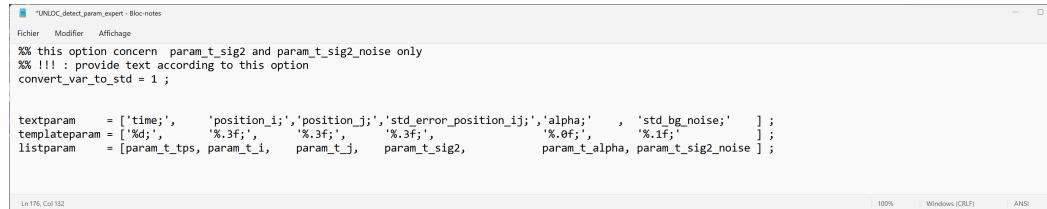


Figure 5.4: Open the "UNLOC" folder

3. Open the 'UNLOC_detect_param_expert.m' with Matlab or any text editor (Notepad, etc.)
4. Edit the three lines located at the end of the file by adding few information:

Without the *r0* column

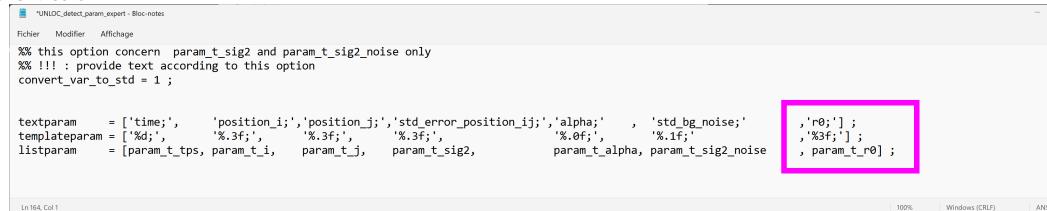


```
%UNLOC_detect_param_expert - Bloc-notes
Fichier Modifier Affichage
%% this option concern param_t_sig2 and param_t_sig2_noise only
%% !!! : provide text according to this option
convert_var_to_std = 1 ;

textparam    = ['time;',      'position_i;', 'position_j;', 'std_error_position_ij;', 'alpha;',     'std_bg_noise;' ] ;
templateparam = ['%d;',        '%.3f;',       '%.3f;',       '%.3f;',       '%.0f;',      '%.1f;' ] ;
listparam    = [param_t_tps, param_t_i,    param_t_j,    param_t_sig2,           param_t_alpha, param_t_sig2_noise ] ;

Ln 179, Col 132
100% Windows (CRLF) ANSI
```

With the *r0* column



```
%UNLOC_detect_param_expert - Bloc-notes
Fichier Modifier Affichage
%% this option concern param_t_sig2 and param_t_sig2_noise only
%% !!! : provide text according to this option
convert_var_to_std = 1 ;

textparam    = ['time;',      'position_i;', 'position_j;', 'std_error_position_ij;', 'alpha;',     'std_bg_noise;' ] ;
templateparam = ['%d;',        '%.3f;',       '%.3f;',       '%.3f;',       '%.0f;',      '%.1f;' ] ;
listparam    = [param_t_tps, param_t_i,    param_t_j,    param_t_sig2,           param_t_alpha, param_t_sig2_noise , 'r0;'] ;
listparam    = [param_t_tps, param_t_i,    param_t_j,    param_t_sig2,           param_t_alpha, param_t_sig2_noise , '%3f;'] ;
listparam    = [param_t_tps, param_t_i,    param_t_j,    param_t_sig2,           param_t_alpha, param_t_sig2_noise , param_t_r0] ;

Ln 164, Col 1
100% Windows (CRLF) ANSI
```

Figure 5.5: Edit three lines in the 'UNLOC_detect_param_expert.m' file

6.1 Supported systems

- Fiji
 - The plugin was developed with java development kit (JDK) version 1.8.0_101. It will not work with older version of java especially version 6 and older. Please update Fiji with the latest version of java.
 - UNLOC has been tested on Fiji (bundled with 64 bits Java 1.8.0_322) available [here](#).
- Matlab / Octave. If you want to use the Matlab / Octave code without Fiji, you can use any kind of computer with supported operating system (Windows 11 64-bit, LinuX and Mac). The Matlab codes have been tested on Matlab 2024a .

6.2 Advanced installation procedure

Disable the multi-thread option in the BIOS to optimize the speed of analysis.

On Linux and Mac OS 10 platforms, the plugin searches the MCR in the default installation path: `/usr/local/MATLAB/MATLAB_Runtime/R2024a` or `//Applications/MATLAB/MATLAB_Runtime/R2024a` respectively.

But, if you choose another location, you must add manually the MCRROOT to the environment variables. For example under Linux, you can proceed as this in a terminal window: `/sudo sh -c 'echo export MCRROOT=/my/mcr/path/R2024a >> /etc/profile'`. Then a re-login is required.

Moreover, in Linux, if you want to create a shortcut to Fiji, you have to specify the working directory. It is sometimes directly possible when you create the shortcut with your user interface. For a desktop entry files, you can also add (or modify) the "Path" key accordingly. Otherwise, write the executing command like this template: `bash /-c "cd /path/to/Fiji && /path/to/Fiji"`.

6.3 Disclaimer

This manual describes the 2.4 version of UNLOC of 2024, April. Although the authors has made every effort to ensure that the information contained in it was correct at published time, they do not assume and hereby disclaims any liability to any party for any loss, damage or disruption caused by errors or omissions, whether such errors or omissions result from negligence, accident or any other cause.

List of acronyms

CRB	Cramér-Rao Bound
dSTORM	direct STochastic Optical Reconstruction Microscopy
EMCCD	Electron Multiplying Charge Coupled Device
GSDIM	Ground State Depletion microscopy followed by Individual Molecule return
HD	High Density
JDK	Java Development Kit
LD	Low Density
MCR	Matlab Compiler Runtime
MD	Medium Density
MTT	Multiple Target Tracing
PAINT	Point accumulation in nanoscale topography
PALM	Photo-Activated Localization Microscopy
PSF	Point Spread Function
ROI	Region Of Interest
sCMOS	scientific Complementary Metal-Oxide-Semiconductor
SNR	Signal to Noise Ratio
SMLM	Single Molecule Localization Microscopy
SPT	Single Particle Tracking
STORM	STochastic Optical Reconstruction Microscopy
TIRFM	Total Internal Reflection Fluorescence Microscopy
UNLOC	UNsupervised particle LOCalization