HOW TO USE ZOLA

Benoit Lelandais[1,2] (benoit.lelandais@pasteur.fr)

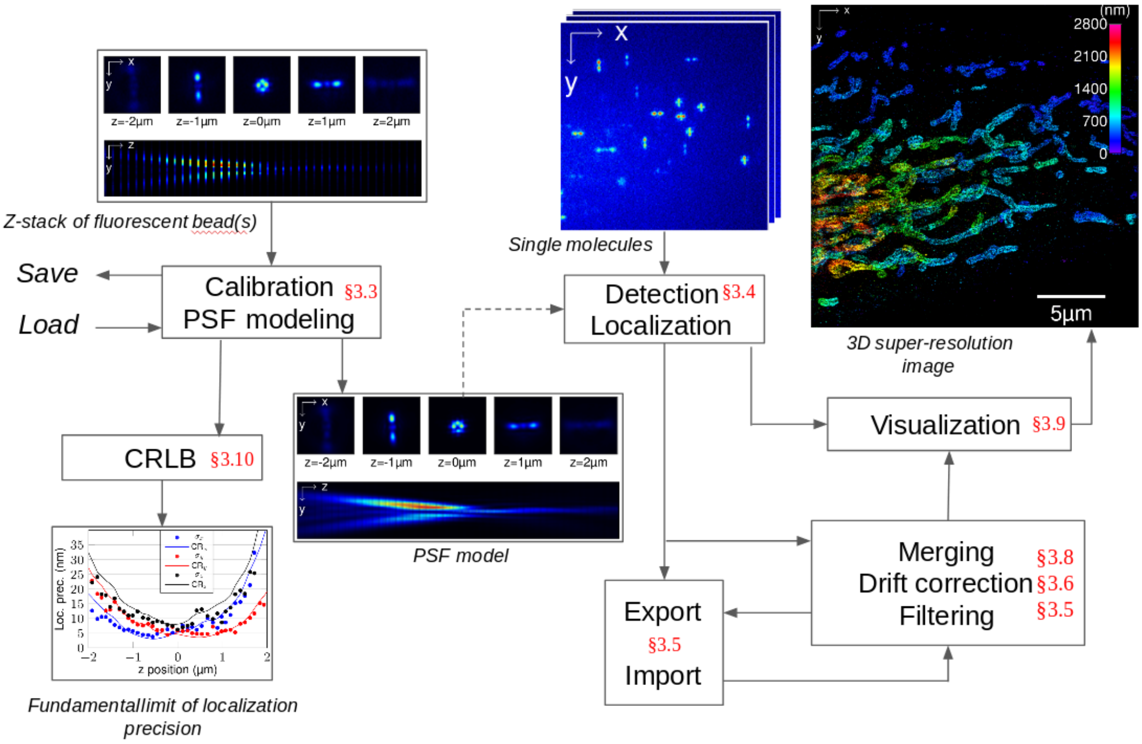
[1] Institut Pasteur, Unité Imagerie et Modélisation, 25 rue du Docteur Roux, 75015 Paris, France

[2] 2UMR 3691, CNRS; C3BI, USR 3756, IP CNRS

Contents

1 Introduction

Thank you for downloading ZOLA. ZOLA is an ImageJ/Fiji plugin that allows you to determine the PSF model corresponding to your optical system, and use this model to reconstruct 3D single molecule localization microscopy (SMLM) images. ZOLA also provides tools to correct drifts, merge consecutive localizations or render images. The following scheme presents an overview of methods featured ZOLA:



2 Requirement

Our plugin requires the 64 bit version of ImageJ or Fiji software available here: https://imagej.net/Fiji/Downloads. The calibration, localization or drift correction tools are highly computationally expensive. To increase efficiency, the computations can be performed on GPU (Graphical Process Unit). Currently, these functions can be launched on Cuda compatible NVIDIA GPU.
Please, go to http://www.nvidia.fr/page/home.html and http://docs.nvidia.com/cuda/index.html to install your `NVIDIA GPU` and `Cuda-toolkit`.

3 ZOLA plugin

Plugin installation requires 15 minutes. If you plan to benefit from GPU acceleration, plugin installation requires 15 minutes more. Follow these instructions to install the plugin:

3.1 ZOLA plugin installation

3.1.1 ZOLA plugin

Moving ZOLA_-0.1.X-SNAPSHOT.jar to your *ImageJ/plugin* folder is enough to install the plugin. However, you can not benefit from the possibility to update new realeses of the plugin. Using Fiji, you can follow these instructions to install the plugin with the last version:

1. Launch `Fiji`.
2. Select `Help` → `Update`.
3. Select `Manage update sites`.
4. Select `IMOD-ZOLA` from the list.
5. Clic `Close`.
6. Clic `Apply changes`.
7. Restart `Fiji`.
8. ZOLA should now appear in the `Fiji` → `Plugins` dropdown menu

3.1.2 Nvidia GPU acceleration

Computation using GPU requires the use of some libraries. Follow these instructions to add them:

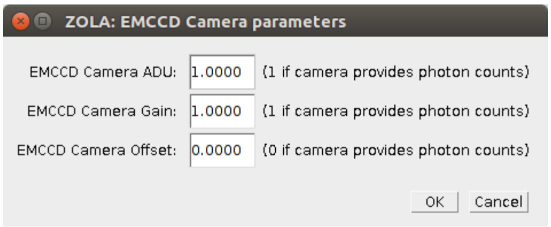
1. Open a terminal.
2. Check your version by executing `nvcc -version` in the terminal. We provide libraries for cuda 7.5, 8 and 9.
3. Open *JCuda libraries* folder and open the subfolder corresponding to your operating system and your Cuda version. If your Cuda version older than 7.5, the libraries can be found here: http://www.jcuda.org/downloads/downloads.html.
4. Paste the *lib* folder into your *ImageJ* folder.
5. Copy and paste the content of the *jars* folder into your *ImageJ/jars* folder (or *ImageJ/plugin/jars* folder depending on your version of ImageJ).

3.1.3 Source code

The source code of ZOLA is also available in *zola_source_code* folder. Please, check the *zola_source_code/README.md* file to compile it.

3.2 Camera photon count

Zola's algorithm requires camera pixel values to be in photon counts. New EMCCD camera provide photon counts. If this is not the case, Zola can manage both EMCCD and SCMOS camera specific noise. please, set camera parameters using `ZOLA` → `Camera setup` → `EMCCD/SCMOS` plugin.

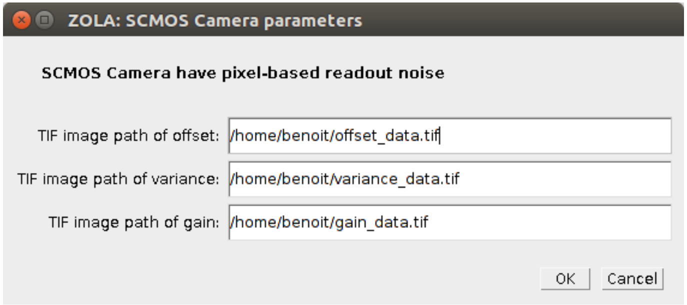


ZOLA: EMCCD Camera parameters

EMCCD Camera ADU: 1.0000 (1 if camera provides photon counts)

EMCCD Camera Gain: 1.0000 (1 if camera provides photon counts)

EMCCD Camera Offset: 0.0000 (0 if camera provides photon counts)



ZOLA: SCMOS Camera parameters

SCMOS Camera have pixel-based readout noise

TIF image path of offset: /home/benoit/offset_data.tif

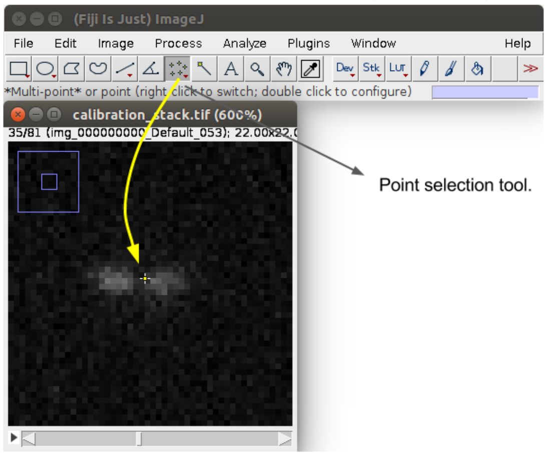
TIF image path of variance: /home/benoit/variance_data.tif

TIF image path of gain: /home/benoit/gain_data.tif

If you use EMCCD camera, pixel photon conversion needs three parameters: ADU and Offset that are given in your Camera specifications, and the Gain choosen in the image acquisition software. If you use SCMOS camera, pixel photon conversion is a bit more complicated since each pixel has its own statistic. We propose the photon conversion proposed by Huang et al. *Nat. Methods, (10) 2013*. It uses three maps that have to be calibrated (Offset, Variance and Gain) allowing to convert each pixel in photon count in ZOLA. The three maps correspond to TIF files whose width and height have to be the same as images used for calibration and localization.

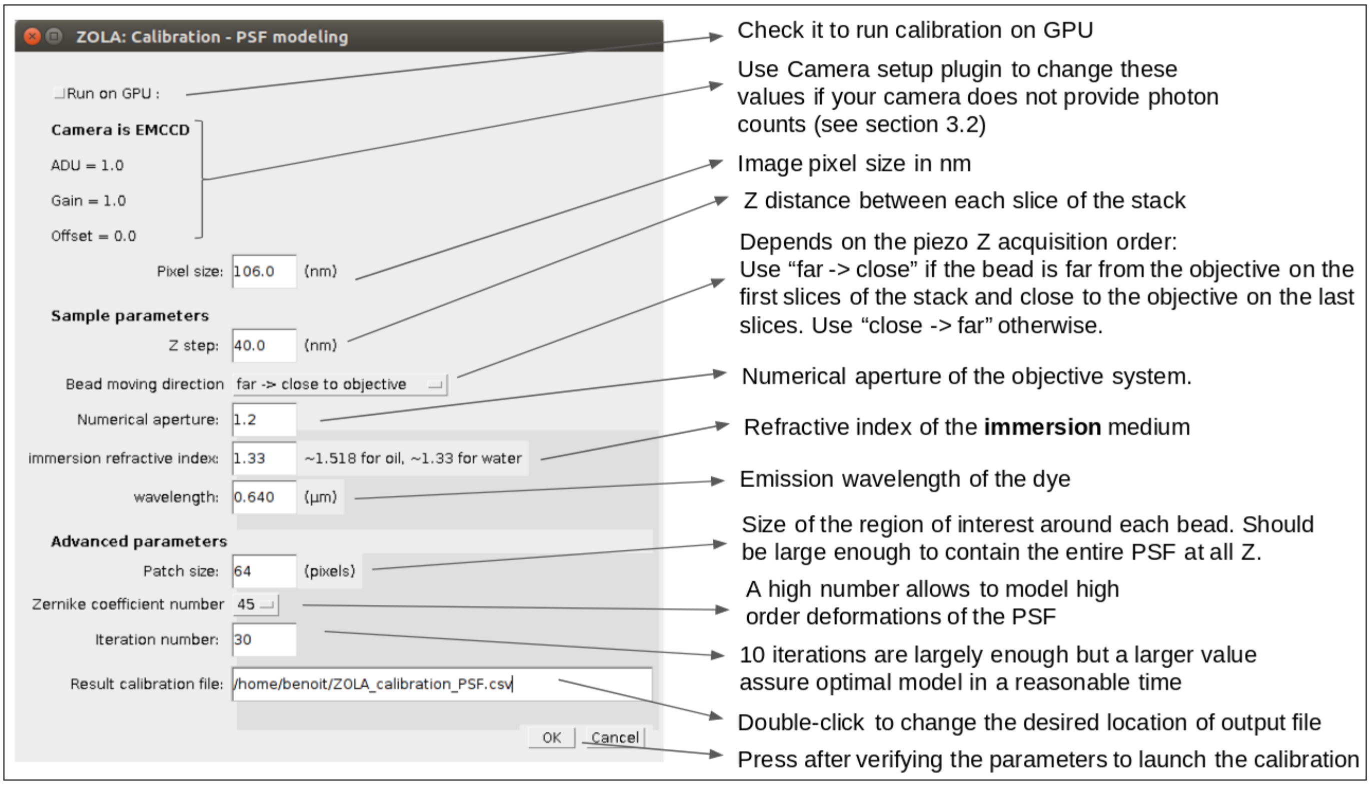
3.3 Calibration - PSF modeling

ZOLA's calibration tool allows to model the PSF from a stack of images of fluorescent beads at different Z positions. The algorithm assumes that there is no drift during the acquisition. Thus, it is recommended to acquire the stack of images fast with stable temperature in the room. Usually, the axial distance between slices is set to 50 *nm* and the acquisition time per slice is 50 *ms*. Thus, the acquisition time for a 2 µm Z range is 2 seconds.
First, open a stack of images and select one or multiple beads using the ImageJ Point selection tool:



(Fiji Is Just) ImageJ
File
Edit
Image
Process
Analyze
Plugins
Window
Help
Multi-point or point (right click to switch; double click to configure)
calibration_stack.tif (600%)
35/81 (img_000000000_Default_053); 22.00x22.0
Point selection tool.

A single bead is usually enough to accuratly model the PSF. If your image contains overlapping beads or has a low signal to noise ratio, it is still possible to model the PSF by selecting multiple beads.
Next, open the `ZOLA` → `Calibration PSF modeling` plugin. The following window pops up:



ZOLA: Calibration - PSF modeling

□Run on GPU :

Camera is EMCCD

ADU = 1.0

Gain = 1.0

Offset = 0.0

Pixel size: 106.0 (nm)

Sample parameters

Z step: 40.0 (nm)

Bead moving direction far -> close to objective

Numerical aperture: 1.2

immersion refractive index: 1.33 ~1.518 for oil, ~1.33 for water

wavelength: 0.640 (µm)

Advanced parameters

Patch size: 64 (pixels)

Zernike coefficient number 45

Iteration number: 30

Result calibration file: /home/benoit/ZOLA_calibration_PSF.csv

Check it to run calibration on GPU

Use Camera setup plugin to change these values if your camera does not provide photon counts (see section 3.2)

Image pixel size in nm

Z distance between each slice of the stack

Depends on the piezo Z acquisition order: Use "far -> close" if the bead is far from the objective on the first slices of the stack and close to the objective on the last slices. Use "close -> far" otherwise.

Numerical aperture of the objective system.

Refractive index of the **immersion** medium

Emission wavelength of the dye

Size of the region of interest around each bead. Should be large enough to contain the entire PSF at all Z.

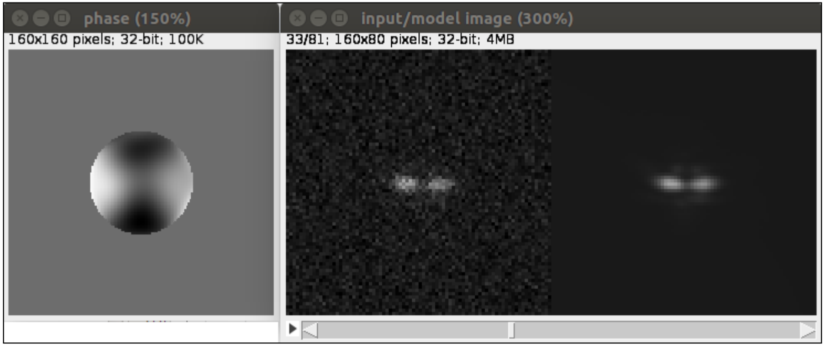
A high number allows to model high order deformations of the PSF

10 iterations are largely enough but a larger value assure optimal model in a reasonable time

Double-click to change the desired location of output file

Press after verifying the parameters to launch the calibration

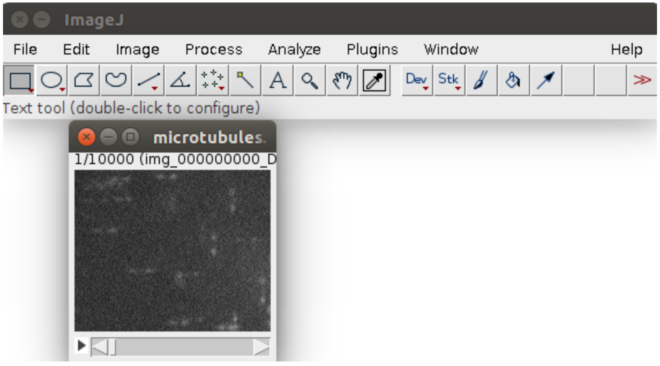
After verifying the parameters, then press OK. Phase retrieval begings. This can take few minutes. At the end of the phase retrieval, the plugin shows the phase and the model of the PSF as shown below. If the modeled PSF does not match the bead image well, try to reduce the number of Zernike coefficients or use more beads.


phase (150%)
160x160 pixels; 32-bit; 100K
input/model image (300%)
33/81; 160x80 pixels; 32-bit; 4MB

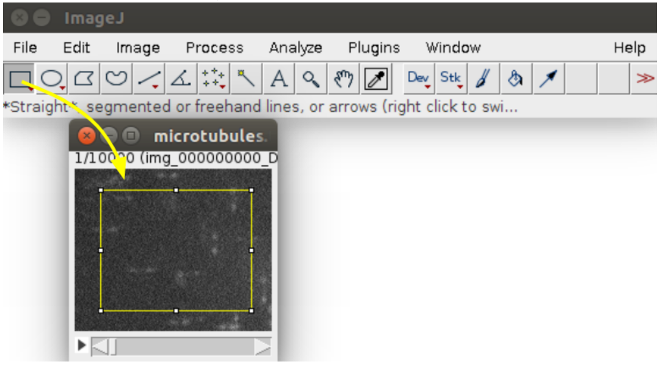
Retrieved phase   Bead image   PSF model result
(Use the slide to change Z)

3.4 Single molecule Localization

ZOLA's localization plugin allows to detect and precisely localize single molecules in a sequence of difraction limited images as obtained in SMLM experiments. The process is the following:
First, open the sequence of images. If desired, a region of interest (ROI) can be drawn to reduce the size of the region to reconstruct as shown below (this increases the reconstruction speed).

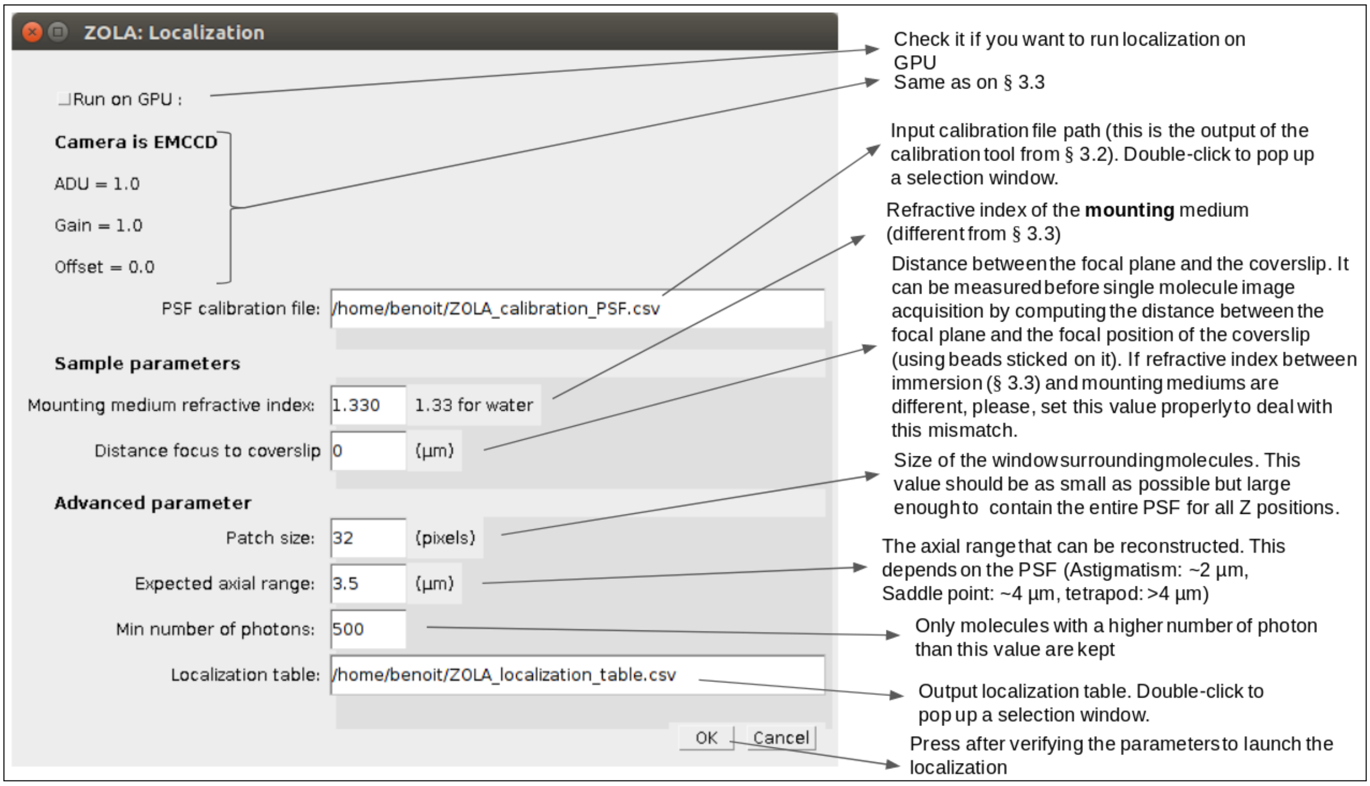

ImageJ
File Edit Image Process Analyze Plugins Window Help
Text tool (double-click to configure)
microtubules.
1/10000 (img_000000000_D

Full sequence of images


ImageJ
File Edit Image Process Analyze Plugins Window Help
Straight, segmented or freehand lines, or arrows (right click to swi...
microtubules.
1/10000 (img_000000000_D

Sequence of images with ROI selection

Then, open the ZOLA → Localization plugin. The following window pops up:



ZOLA: Localization

□ Run on GPU :

Camera is EMCCD

ADU = 1.0

Gain = 1.0

Offset = 0.0

PSF calibration file: /home/benoit/ZOLA_calibration_PSF.csv

Sample parameters

Mounting medium refractive index: 1.330 1.33 for water

Distance focus to coverslip 0 (µm)

Advanced parameter

Patch size: 32 (pixels)

Expected axial range: 3.5 (µm)

Min number of photons: 500

Localization table: /home/benoit/ZOLA_localization_table.csv

Check it if you want to run localization on GPU

Same as on § 3.3

Input calibration file path (this is the output of the calibration tool from § 3.2). Double-click to pop up a selection window.

Refractive index of the **mounting** medium (different from § 3.3)

Distance between the focal plane and the coverslip. It can be measured before single molecule image acquisition by computing the distance between the focal plane and the focal position of the coverslip (using beads sticked on it). If refractive index between immersion (§ 3.3) and mounting mediums are different, please, set this value properly to deal with this mismatch.

Size of the window surrounding molecules. This value should be as small as possible but large enough to contain the entire PSF for all Z positions.

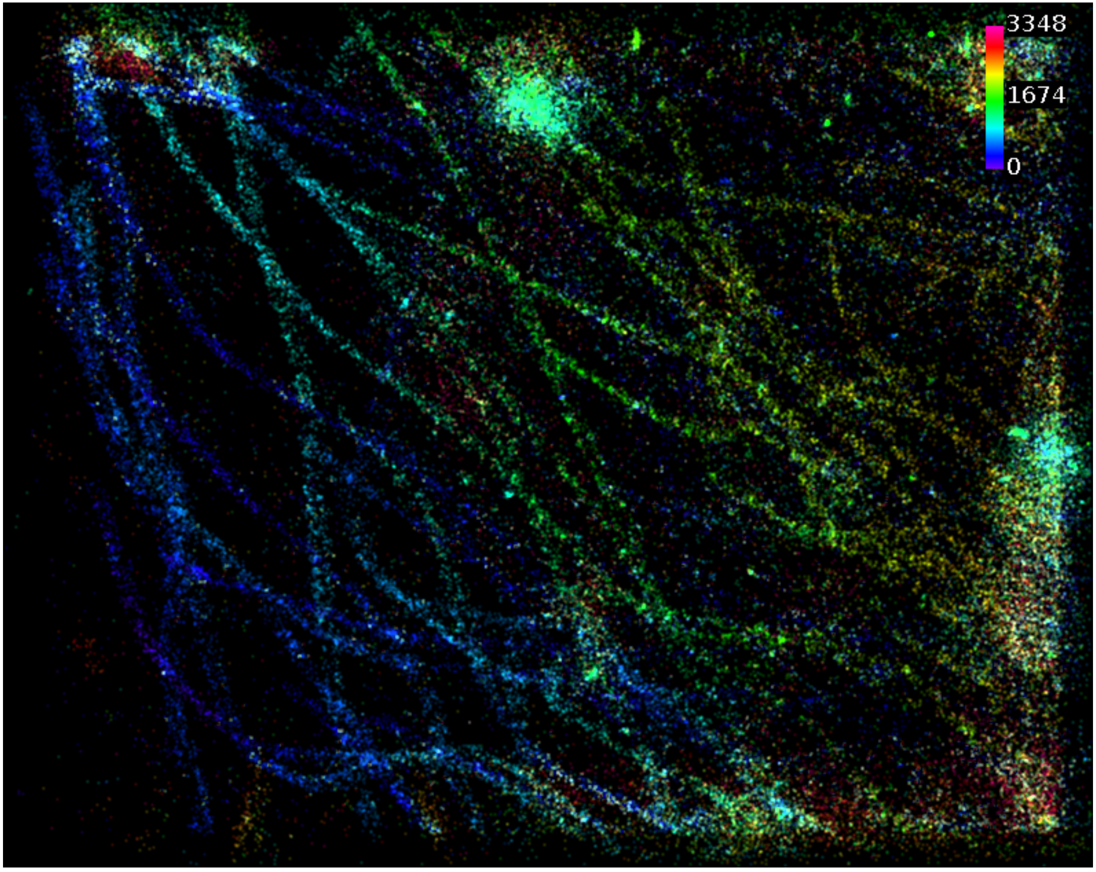
The axial range that can be reconstructed. This depends on the PSF (Astigmatism: ~2 µm, Saddle point: ~4 µm, tetrapod: >4 µm)

Only molecules with a higher number of photon than this value are kept

Output localization table. Double-click to pop up a selection window.

Press after verifying the parameters to launch the localization

The localization starts after clicking on `OK`. On test_data we provide, localization requires around 10 minutes. During localization, a colored super-resolution image (whose color encodes Z position) is updated every 30 seconds. The following figure shows the result obtained at the end of the localization. Other visualizations can be obtained using the ZOLA visualization tool (see section 3.9). Note that the quality of this image can be improved applying drift correction or filtering (see sections 3.6 and 3.7).



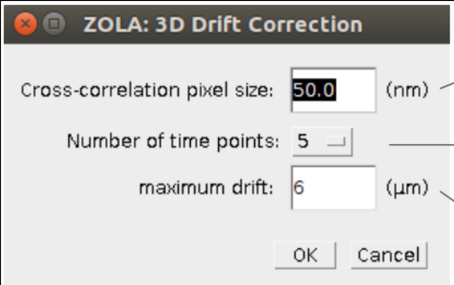
3.5 Import/Export

ZOLA allows to load a localization table in memory, or save it on disk.

- Import: load a localization table.
- Append: Append another localization table to the existing table (useful when consecutive image sequences were processed separatly).
- Export: Save a localization table (useful to save data after post-processing, *e.g.* after drift correction, filtering or merging (see sections 3.6, 3.7 and 3.8).

3.6 Drift correction

ZOLA features a 3D drift correction method based on redundant cross correlation. The drift correction window is as follows:



ZOLA: 3D Drift Correction

Cross-correlation pixel size: 50.0 (nm)

Number of time points: 5

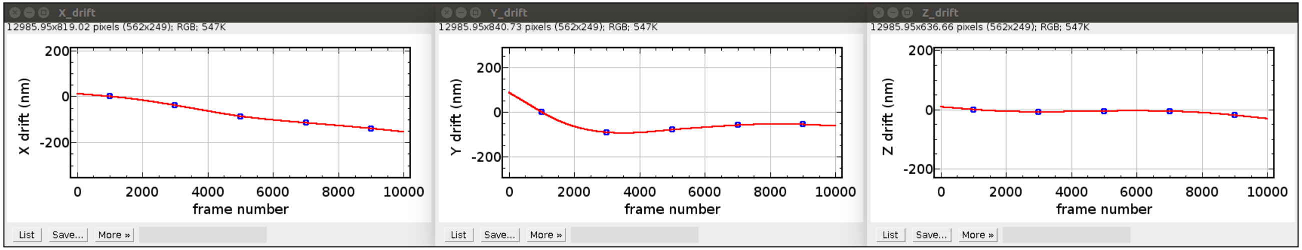
maximum drift: 6 (µm)

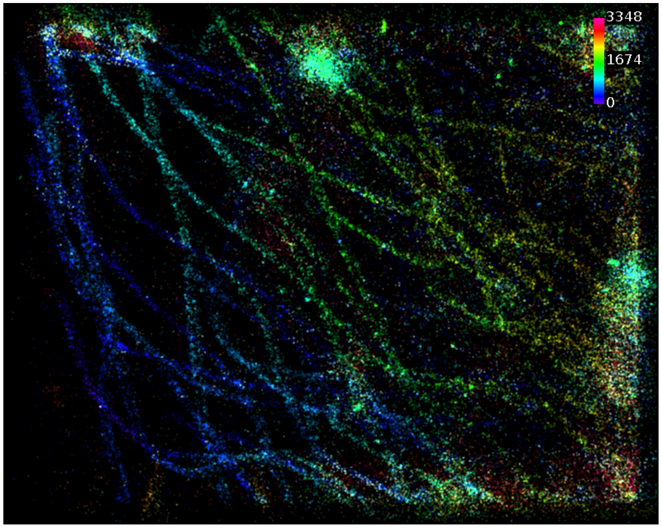
Pixel size of the rendered images used to compute drift. We recommend to use 50 nm for fast processing. Smaller values can potentially improve precision of drift correction (and hence resolution) but precision will be slower. If the available GPU memory is not sufficient, this value will be adjusted automatically

Number of times the localization table is splitted in non-overlapping time windows. Usually, we choose the total frame number divided by 10000.

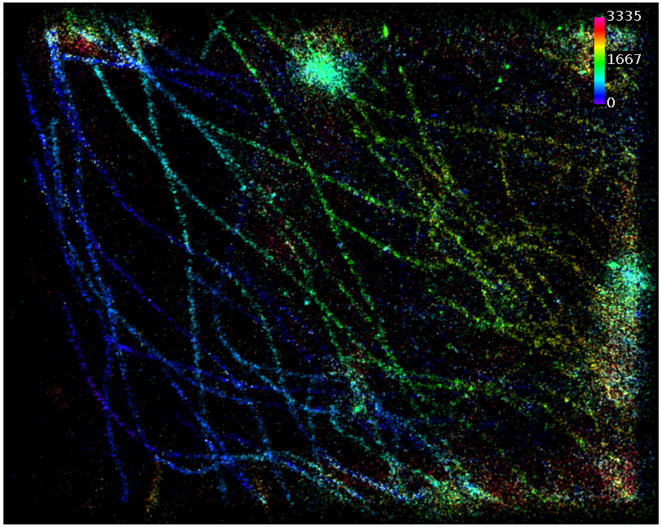
Lower values reduce memory consumption. However, if the value chosen is lower than the actual drift, the drift correction will fail. We recommend to set this value to twice the maximum expected drift. In our experiments, we expect to have a maximum drift about 3 µm, and therefore set this parameter to 6 µm.

As output, the drift correction plugin shows three curves corresponding to X, Y and Z drifts. A drift-corrected super-resolution image (whose color encodes Z position) is also shown:





Super-resolution image before drift correction

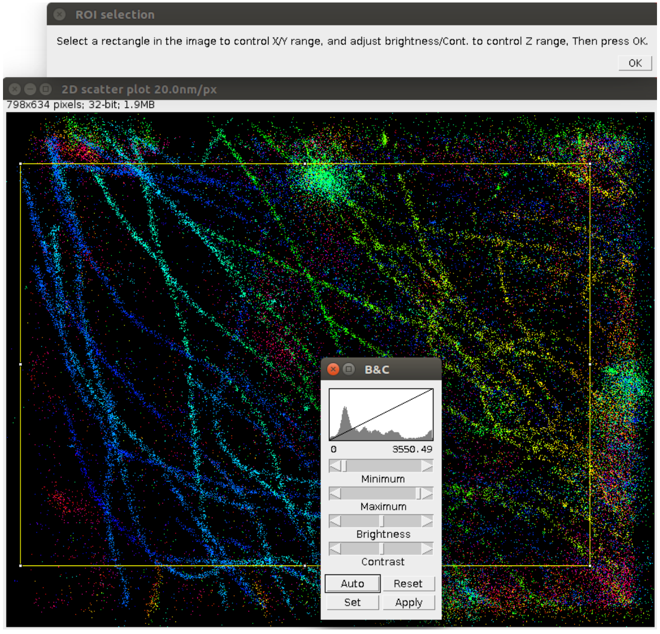


Drift-corrected super-resolution image

After applying drift correction, use `ZOLA→Export` plugin to save the localization table (see section 3.5).

3.7 Filtering

Filtering is useful to define a sub region of a 3D image, or to remove localizations with low precision or unsatisfying goodness-of-fit. First, the user selects a region of interest in the image (optional), and can use `Brightness&Contrast` tool (automatically launched) to select the desired axial range. An example is shown on the next figure:

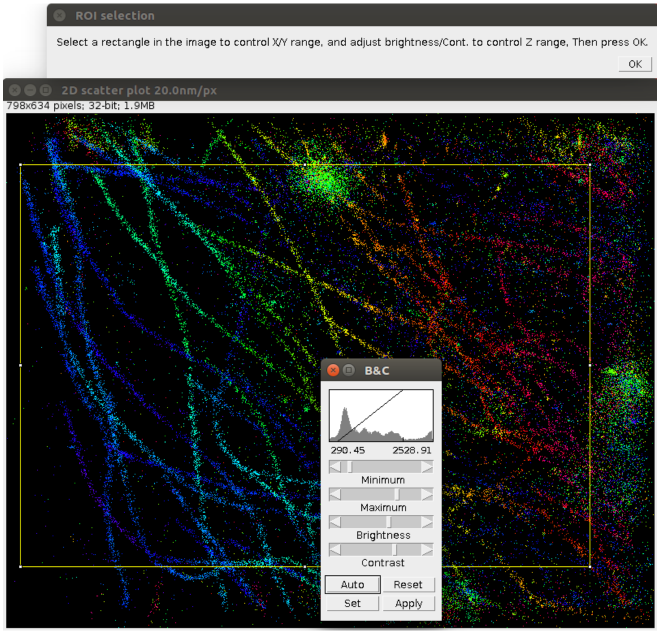

ROI selection

Select a rectangle in the image to control X/Y range, and adjust brightness/Cont. to control Z range, Then press OK.

2D scatter plot 20.0nm/px

798x634 pixels; 32-bit; 1.9MB

Full axial range: 3.5 µm


ROI selection

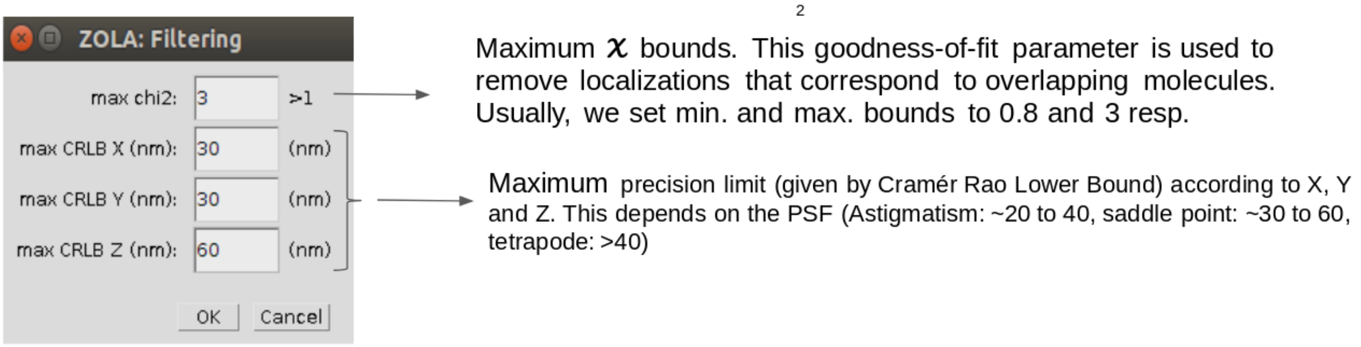
Select a rectangle in the image to control X/Y range, and adjust brightness/Cont. to control Z range, Then press OK.

2D scatter plot 20.0nm/px

798x634 pixels; 32-bit; 1.9MB

Clipped axial range: 2.3 µm

Then, after pressing `OK`, the following panel is opened allowing to filter localizations according to their good goodness-of-fit and their precision.


ZOLA: Filtering

max chi2: 3 >1

max CRLB X (nm): 30 (nm)

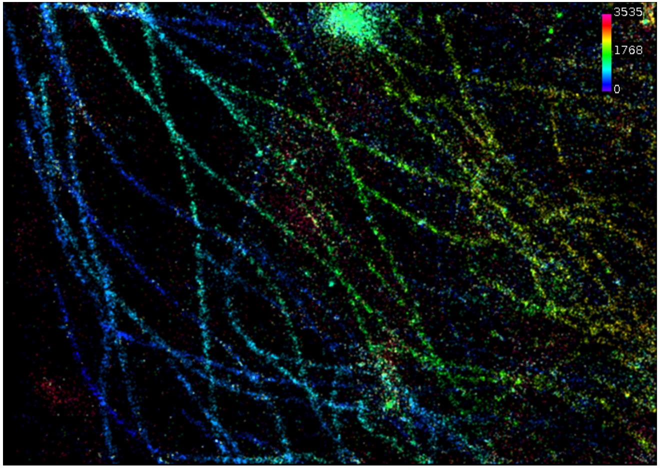
max CRLB Y (nm): 30 (nm)

max CRLB Z (nm): 60 (nm)

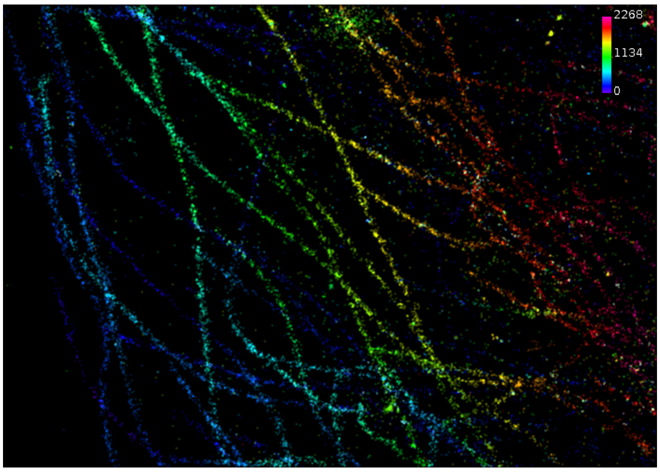
Maximum χ bounds. This goodness-of-fit parameter is used to remove localizations that correspond to overlapping molecules. Usually, we set min. and max. bounds to 0.8 and 3 resp.

Maximum precision limit (given by Cramér Rao Lower Bound) according to X, Y and Z. This depends on the PSF (Astigmatism: ~20 to 40, saddle point: ~30 to 60, tetrapode: >40)

After pressing `OK`, the super-resolution resulting image pops up as shown below. As we can see, filtering allows to clean up data.



Super-resolution image before filtering

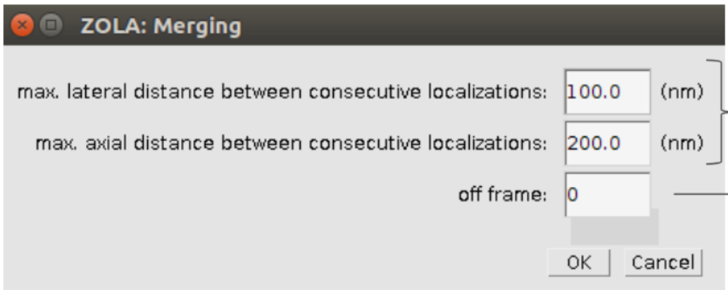


Super-resolution image after filtering

After applying merging, use `ZOLA→Export` plugin to save the localization table (see section 3.5).

3.8 Merging

This plugin is useful to merge localizations that are in close proximity in consecutive frames and are likely to originate from the same molecule. Open `ZOLA→Merging` and the following window pops up:



ZOLA: Merging

max. lateral distance between consecutive localizations: 100.0 (nm)

max. axial distance between consecutive localizations: 200.0 (nm)

off frame: 0

Maximum distance between localizations to be merged

Maximum number of frames a molecule is allowed to disappear before reappearing. Usually, we set it to 0, meaning that only molecules that appear in consecutive frames will be merged.

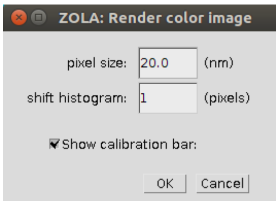
Then, click `OK` to launch the merging. As output, the plugin gives the number of localization that are merged and show the resulting super-resolution image.
After applying merging, use `ZOLA→Export` plugin to save the localization table (see section 3.5).

3.9 Visualization

ZOLA visualization plugin allows to render super-resolution images. The visualization can be launched during or after localization or after importing a localization table. Three rendering methods are proposed based on histograms:

- Automatic 2D color histogram: Renders a 2D color histogram adjusting automatically the axial range and localizations that have a good precision (This method is launched automatically during localization and after import, drift correction, filtering and merging functions).
- 2D color histogram: Renders a 2D color histogram.
- 2D/3D histogram: Render a 2D or 3D histogram.

The three visualization windows are as follow:



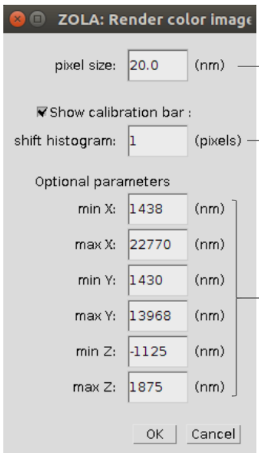
ZOLA: Render color image

pixel size: 20.0 (nm)

shift histogram: 1 (pixels)

☑ Show calibration bar:

automatic color histogram



ZOLA: Render color image

pixel size: 20.0 (nm)

☑ Show calibration bar :

shift histogram: 1 (pixels)

Optional parameters

min X: 1438 (nm)

max X: 22770 (nm)

min Y: 1430 (nm)

max Y: 13968 (nm)

min Z: -1125 (nm)

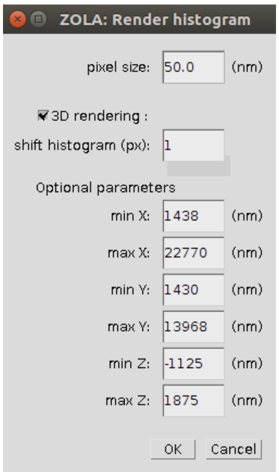
max Z: 1875 (nm)

Isotropic pixel size of super-resolution image

0: pure histogram. Increasing this value will produce less noisy but slightly blurred images

Optional parameters to define the bounds of the rendered image. (useful to fix the field of view)

Color histogram



ZOLA: Render histogram

pixel size: 50.0 (nm)

☑ 3D rendering :

shift histogram (px): 1

Optional parameters

min X: 1438 (nm)

max X: 22770 (nm)

min Y: 1430 (nm)

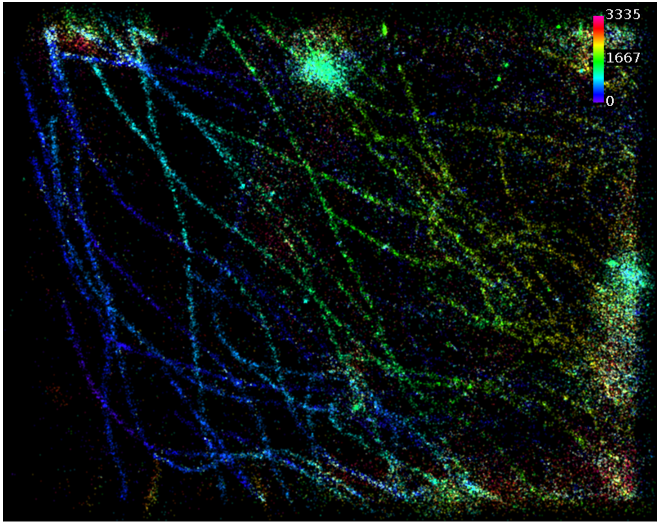
max Y: 13968 (nm)

min Z: -1125 (nm)

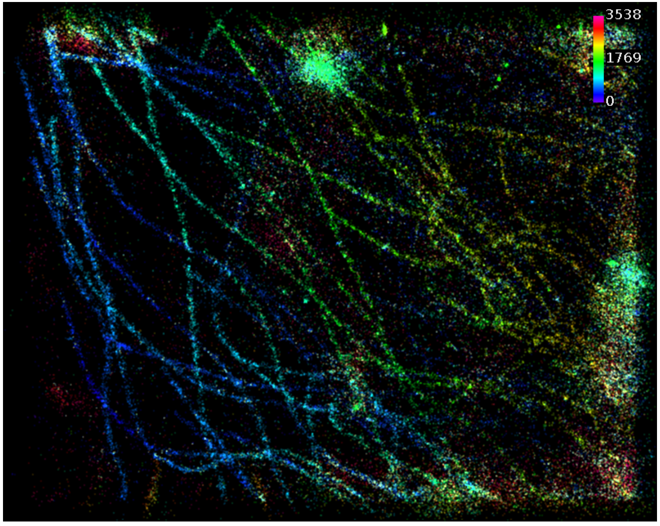
max Z: 1875 (nm)

2D/3D Histogram

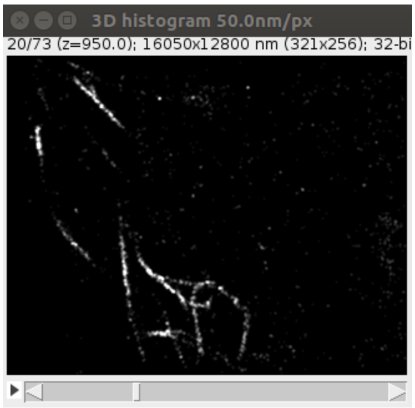
According to the method used, the resulting visualization is the following:



automatic color histogram (clipped axial range)



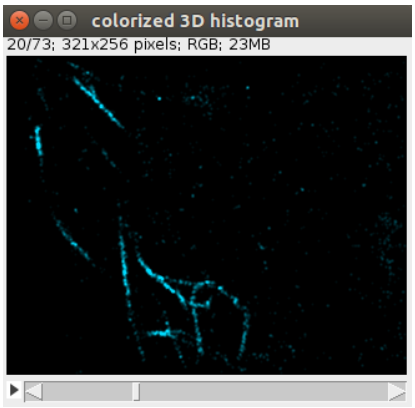
Color histogram (full axial range)



20/73 (z=950.0); 16050x12800 nm (321x256); 32-bi

3D histogram (Z stack)

⇒

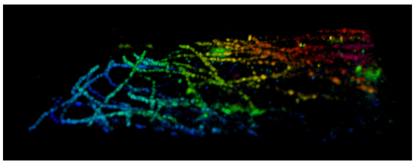


20/73; 321x256 pixels; RGB; 23MB

Colorized 3D histogram (Z stack)

`ZOLA→additional tool→colorize opened 3D stack`

⇒



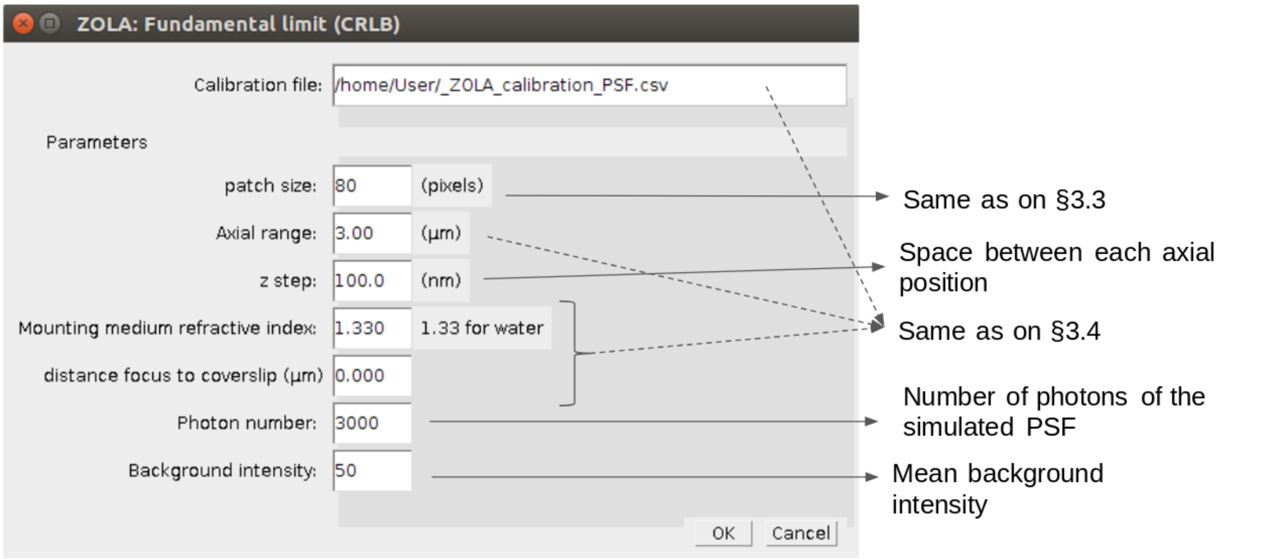
3D visualization

`3D viewer` Fiji plugin

3.10 Additional tools

3.10.1 Theoretical localization precision limit: CRLB (Cramér Rao Lower Bound)

Zola's CRLB plugin allows to compute the fundamental localization precision limit for a given PSF and an expected molecule number of photon and background The CRLB of X, Y and Z is calculated as a function of axial position. Open the `ZOLA` → `CRLB` plugin. The following window pops up:



Calibration file: /home/User/_ZOLA_calibration_PSF.csv

Parameters

patch size: 80 (pixels)

Axial range: 3.00 (µm)

z step: 100.0 (nm)

Mounting medium refractive index: 1.330 1.33 for water

distance focus to coverslip (µm) 0.000

Photon number: 3000

Background intensity: 50

Same as on §3.3

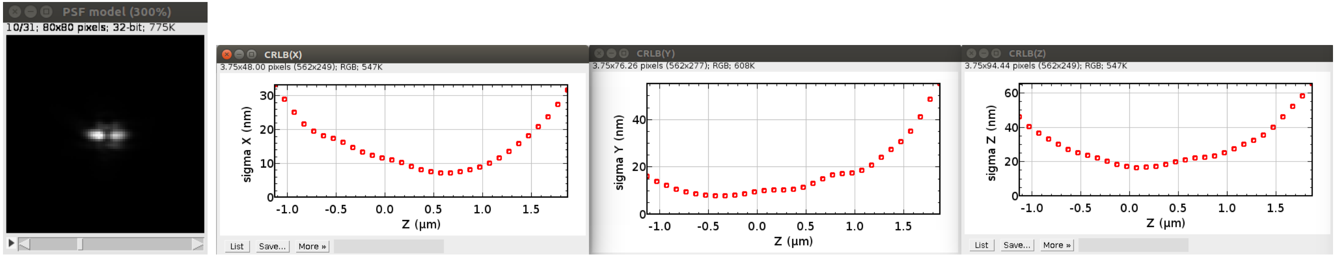
Space between each axial position

Same as on §3.4

Number of photons of the simulated PSF

Mean background intensity

As a result, the tool shows the PSF and 3 plots showing the precision limit (the standard deviation of localization errors) for X, Y and Z as a function of axial position Z:



10/31; 80x80 pixels; 32-bit; 775K

3.75x48.00 pixels (562x249); RGB; 547K

3.75x76.26 pixels (562x277); RGB; 608K

3.75x94.44 pixels (562x249); RGB; 547K