

Easy-DHPSF 2.0: open-source software for three-dimensional localization and two-color registration of single molecules with nanoscale accuracy

Camille Bayas¹, Alex von Diezmann¹, Anna-Karin Gustavsson^{1,2}, and W. E. Moerner¹

¹*Department of Chemistry, Stanford University, Stanford, CA 94305*

²*Department of Biosciences and Nutrition, Karolinska Institutet, Stockholm, SE-17177*

Reprinted from Protocol Exchange:

Camille Bayas, Alex von Diezmann, Anna-Karin Gustavsson et al. Easy-DHPSF 2.0: open-source software for three-dimensional localization and two-color registration of single molecules with nanoscale accuracy, 10 June 2019, PROTOCOL (Version 2) available at Protocol Exchange [https://doi.org/10.21203/rs.2.9151/v2] Code available on sourceforge.net/projects/easy-dhpsf/

Abstract

Automated processing of double-helix (DH) microscope images of fluorescent single molecules (SMs) streamlines the protocol required to obtain super-resolved three-dimensional (3D) reconstructions of ultrastructures in biological samples by single-molecule active control microscopy (SMACM). Here, we present a suite of MATLAB subroutines, bundled with an easy-to-use graphical user interface (GUI), that facilitates 3D localization of single emitters (e.g. SMs, fluorescent beads, or quantum dots) with typical precisions of tens of nanometers in multi-frame movies acquired using a wide-field DH epifluorescence microscope. The algorithmic approach is based upon template matching for SM recognition and least-squares fitting for 3D position measurement, both of which are computationally expedient and precise. While this approach is straightforward, overlapping images of SMs are ignored, and the precision of least-squares fitting is not as high as maximum likelihood-based methods. Version 2.0 of Easy-DHPSF augments the approach of Version 1.0 by incorporating code that facilitates combining localization data from two spectral channels using a locally-weighted quadratic 3D registration function.

Introduction

Three-dimensional localization of single emitters for super-resolution imaging and single-particle tracking continues to be a vibrant field of study¹. The double-helix microscope² provides 3D information from 2D images using PSF engineering to generate images of single emitters which have two lobes, where the z-position information is contained in the angle between the two lobes. The Easy-DHPSF software provides an easy-to-use set of MATLAB programs for analysis of the “Double-Helix PSF” data generated from this kind of microscope, and with minor changes, can be adapted to other PSFs which use lobe arrangement to encode position³.

Overview of version 2.0 update

Easy-DHPSF version 2.0 utilizes the same algorithm for identification of SMs in a DH microscope image as version 1.0: template matching in each region of interest via phase correlation, then double-Gaussian fitting of the DHPSF⁴ (here called DH for brevity) images through non-linear least squares minimization (**Figure 1**). This algorithm represents a good compromise between fitting accuracy and computational complexity. For a more detailed description of this algorithm, the reader should consult the documentation for Easy-DHPSF v1.0⁵ as the basic details will not be repeated here. Version 2.0

introduces functions for two-color registration and for defining the calibration curves from many points, such as in a regularly-spaced nanohole array (NHA)⁶.

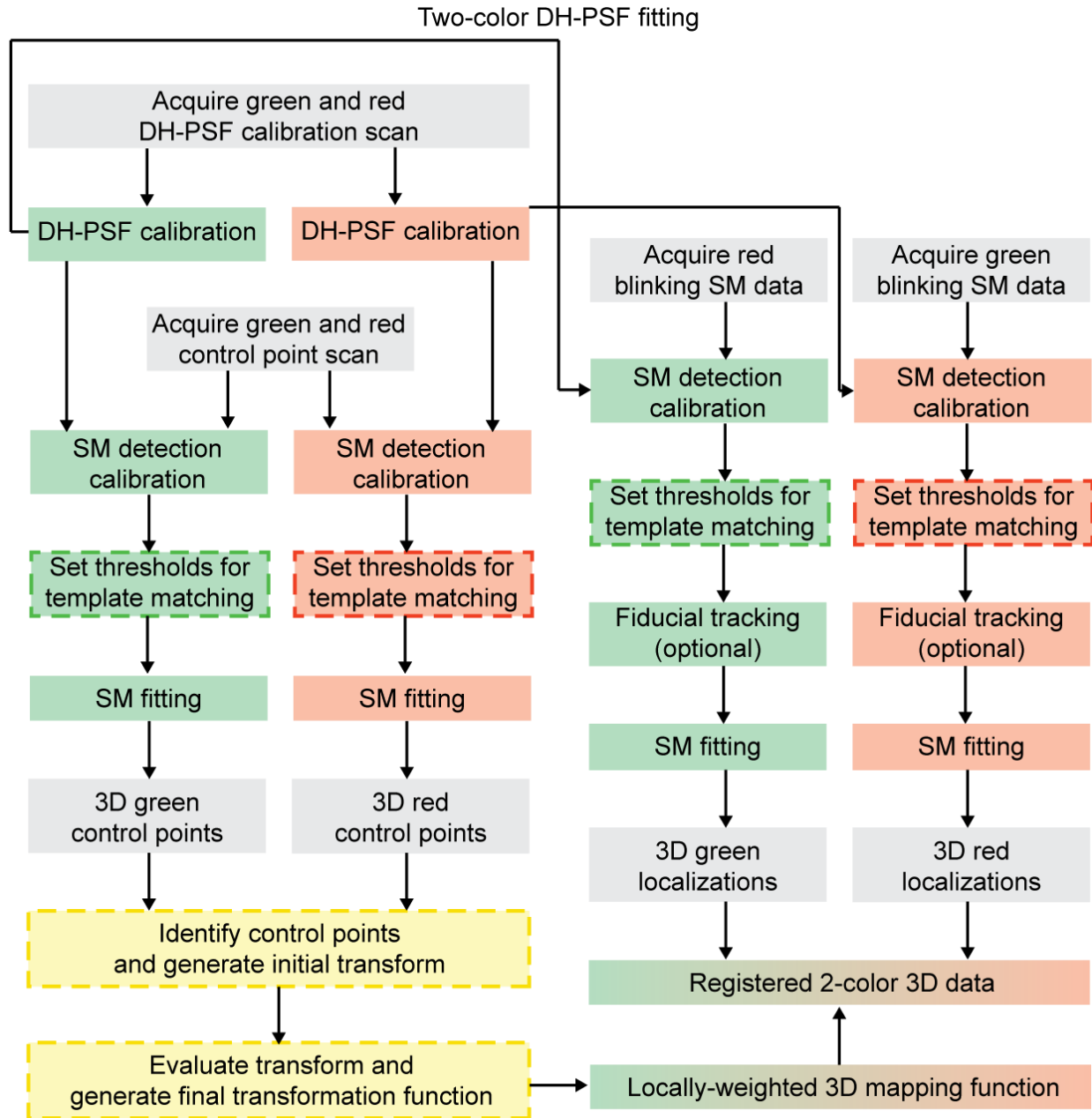


Figure 1: Schematic of Easy-DHPSF v2.0 workflow. Begin flowchart at gray boxes near the top of the image. Green boxes indicate steps for the green color channel, and red boxes indicate steps for the red color channel. Yellow boxes indicate steps that involve both color channels. Dashed outlines indicate that user input is needed during these modules. The final result is the red/green box, “Registered 2-color 3D data.”

Registration of two color channels with a locally-weighted quadratic transform

One key aspect in most two-color wide-field imaging experiments is the procedure used to quantitatively map one channel onto the other to provide one consistent coordinate system. Simple implementations have used affine transforms or other global mapping approaches. Although global mapping functions have the advantage that they can be estimated from only a few corresponding control point pairs, they may not be flexible enough to accurately register high-resolution SM localization data. An accurate mapping function defined over the entire 3D field of view is essential for wide-field high-resolution microscopy.

The mapping function implemented in Easy-DHPSF v2.0 is a 3D locally-weighted quadratic transformation. The mapping function \mathbf{t} transforms points from one coordinate system into another⁷, $\mathbf{t}(\mathbf{x})=\mathbf{x}'$. For coordinate transformations in 3D, $\mathbf{t}(\mathbf{x})$ has 3 components, t_x , t_y , and t_z , that separately generate the three transformed coordinates x' , y' , and z' :

$\mathbf{t}(\mathbf{x}) = (t_x(x, y, z), t_y(x, y, z), t_z(x, y, z)) = (x', y', z')$. The component functions t_x are given as the weighted mean of polynomial transformations $L_{i,x'}$ using the local weights W_i associated with control

point i with coordinates \mathbf{x}_i , $t_x(\mathbf{x}) = \frac{\sum_{i=1}^N L_{i,x'}(\mathbf{x})W_i(\mathbf{x})}{\sum_{i=1}^N W_i(\mathbf{x})}$, where the sum is over the full set of N control

points in the image domain.

The polynomial transformation functions $L_{i,x'}$ are defined exactly for each pair of control points: i.e., $L_{i,x'}(\mathbf{x}_i) = \mathbf{x}_i'$ without error. The x' component of a 3D quadratic transformation is given by the equation below, and y' and z' components are similarly defined:

$$\begin{aligned} L_{i,x'}(x, y, z) = & a_{i,x'}(x - x_i)^2 + b_{i,x'}(y - y_i)^2 + c_{i,x'}(z - z_i)^2 \\ & + d_{i,x'}(x - x_i)(y - y_i) + e_{i,x'}(x - x_i)(z - z_i) + f_{i,x'}(y - y_i)(z - z_i) \\ & + g_{i,x'}(x - x_i) + h_{i,x'}(y - y_i) + i_{i,x'}(z - z_i) \end{aligned}$$

The functions $L_{i,x'}$ are sufficient to define the map between the control points, i.e. the points \mathbf{x}_i and \mathbf{x}_i' , while the weighting coefficients W_i are used to interpolate between these points. The weight associated with control point i is given as a 3D Gaussian sphere centered at \mathbf{x}_i . The width of the Gaussian sphere is defined by a global width parameter s , and a local width parameter σ_i that varies from point to point to accommodate the local density of control points:

$$W_i(\mathbf{x}) = \exp\left(-\frac{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2}{2(s\sigma_i)^2}\right)$$

For the automated computation of the complete transformation function, the user needs to specify three parameters: the number of neighboring control points n used to calculate the coefficients of the polynomial transformation by least squares estimation, the global width parameter s , and the generating function used to define σ_i . These three parameters are systematically determined by the optimization program for each set of control point pairs to produce the lowest overall registration errors as described below in Step 8. The design of the registration algorithm is discussed further in Ref. 7.

Below, we demonstrate the use of open-source Easy-DHPSF v2.0 by analyzing an SM dataset of microtubules immunolabeled with blinking Alexa647 (Life Technologies) and CF568 (Biotium) in BSC-1 cells. This code has been validated in several different imaging conditions (Refs. 6, 8, 9) and is provided free-of-charge under the 3-clause New BSD License at <https://opensource.org/licenses/BSD-3-Clause>. Specifically, this code is provided “as is” without guarantees of any kind.

Experimental Design

Calibration and control points – beads vs. NHA

Calibration and control-point scans can be performed with either fluorescent beads immobilized on a coverslip⁸ or with a metal nanohole array (NHA) filled with a solution containing the fluorophore labels of interest⁶. Whereas use of beads on a coverslip allows for faster fitting and generation of the transformation function, the NHA provides finer sampling of the field of view. This allows both the DH calibration and two-color registration function to account for field-dependent errors arising from aberrations in the imaging system. Moreover, by scanning the NHA laterally, we can sample the field of view in a finer and more controllable way than with beads randomly dispersed onto a coverslip.

Below, to generate the set of control points used to calculate the registration function, we imaged bright fluorescent beads (200 nm Tetraspeck, Invitrogen) dispersed on a coverslip with emission simultaneously visible in both detection channels. We repeatedly scanned the positions of beads throughout a 3D field of view to assemble a large set of control point pairs to finely calibrate the spatial differences in DH response at many locations in the two color channels. A similar procedure of axial and lateral scanning of a fluorophore-filled NHA can also be performed.

Equipment

Software

The Easy-DHPSF package consists of MATLAB functions and subroutines and requires the image processing, optimization, statistics, and wavelet toolboxes. Easy-DHPSF has been tested on workstations running MATLAB 2017b. There are no known issues with earlier versions of MATLAB, but some built-in functions (e.g. `fft2`, `imread`) may have reduced performance in earlier versions.

Data file considerations

Easy-DHPSF v2.0 requires several .tif stacks to run.

The first image stacks are the raw data file(s) of the two color channels with SMs to be localized. If desired, the field of view may contain one or more fiducials to correct for sample drift, ideally located near the edge of the field of view and not overlapping with the region containing single molecules to be localized. The two-color imaging may be performed on a standard two-color SM microscope with each color channel imaged on separate cameras, or the two color channels may be imaged onto different regions of the same camera.

The second image stack is a calibration scan. This should comprise a series of frames at different z -positions of either beads on a coverslip or a NHA that are detected in both color channels simultaneously. As an example: The calibration scan is taken by translating the piezoelectric z stage in 50

nm steps from -1 micron to +1 micron relative to the focus (with ~20 acquired frames at each step to improve precision of the measurement), ensuring that there are enough frames between each step to allow for relaxation of the stage or oil and the sample. A sequence log file should be generated with the sequence of steps used to generate this calibration scan. An example sequence log file for this calibration step is provided (sequenceLog_calibrationScan.dat).

The third image stack is a control point scan. The control point scan is taken by translating the z stage in known step sizes across the range of the DHPSF, i.e., 200 nm steps from -1 micron to +1 micron, with multiple frames (~20) at each step to improve precision. After each complete axial scan, the stage is then translated laterally and the axial scan is repeated to ensure fine sampling of the field of view. The user should ensure that the relative brightness of the DHPSFs are similar in both color channels. The user should also ensure that this scan is taken with the same filters as those used for SM data. A sequence log file should be generated with the sequence of steps used to generate this calibration scan. An example sequence log for this control point scan step is provided (sequenceLog_controlPointScan.dat).

The last image stacks are ‘dark offsets’ for the calibration and the data files. The frames in the dark offset .tif stacks are averaged and subtracted before data processing to account for the offset and dark counts from the EMCCD. Acquire a series of a few hundred frames with the camera shutters closed, and use the same acquisition parameters (e.g. integration time, EM gain, shift speed) as those used for the calibration and data acquisition. If these parameters are the same between the calibration and raw data images, only one dark offset file is necessary.

Procedure

Step 0. Image acquisition and program setup

Acquire the raw data, calibration scan, control point scan, and dark offset images described in Equipment. Ensure that you have saved a sequence log file for the calibration and control point scans.

Step 1. Two-color DHPSF calibrations

This section will outline the steps to perform two-color calibrations of the DHPSF either with beads or with a NHA.

- 1.1 Open an instance of the Easy-DHPSF GUI by entering ‘easy_dhpsf’ in the MATLAB command line.
- 1.2 Before running any of the modules, set the conversion gain and pixel size of the detector using the fields under ‘Setup.’ Note that changing these parameters after processing the data does not retroactively change the calculated results (**Figure 2**).
- 1.3 Under Setup > Channel, select a channel ‘G’ for green or ‘R’ for red.
- 1.4 Under ‘Calibrate DHPSF’, click ‘Run.’
- 1.5 Follow the prompts if you would like to automatically save the partly processed dataset after this module.

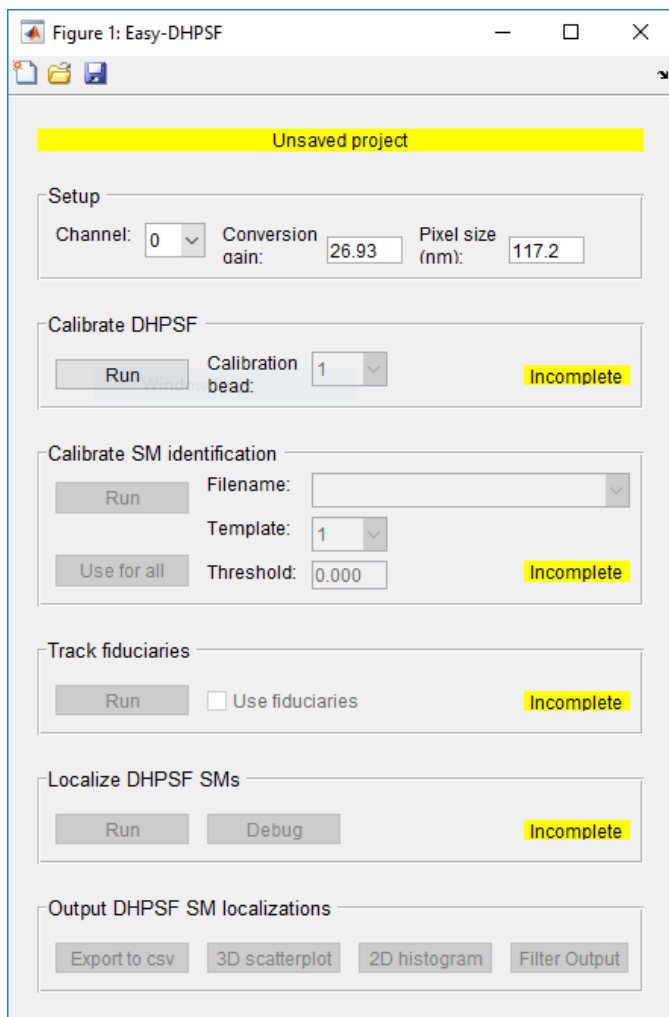


Figure 2. The Easy-DHPSF GUI before any of the modules are run. The user can switch between colors using the dropdown menu under Setup > Channel.

- 1.6 Open the calibration images file, the dark offset image file matching the calibration, and the sequence log file (either .mat or .dat) for the calibration. The program will prompt you for these files via dialog boxes.
- 1.7 Set the EM gain used during the acquisition of the calibration images, whether you would like use wavelet subtraction, NHA holes if using NHA and whether the holes are at 45 degrees, whether the DHPSF's two lobes are horizontal when in focus, and the box radius for fitting.
- 1.8 Select a region of interest to analyze. Follow 1.8.1 if using beads on a coverslip, or 1.8.2 if using an NHA.
- 1.8.1 Select a region of interest (ROI) containing at least one bright fluorescent object by adjusting the box and double-clicking inside it. Even though the calibration bead may be isolated, it is helpful

to select an entire wide-field excitation region so that the background fluorescence can be estimated accurately. Select fluorescent objects by clicking the center of each DHPSF (**Figure 3**).

- 1.8.2 Select an ROI by adjusting the box and double-clicking inside it. Select 4 holes that form a large square, then select fiducials to outline the edge of your field of view (**Figure 4**). Make sure to try to click the center of each DHPSF, halfway between the two lobes. Hit enter when finished DHPSFs.
- 1.9 Each frame of raw data, as well as the accompanying double Gaussian fit, is displayed as it is processed.
- 1.10 After processing, an output of calibration information will be generated in the 'calibration [DateAndTime]' folder. Inspect this for each bead to ensure that the angle vs. z position curve is roughly linear, and that the variation of angle measurements in each step is not unreasonably high (error bars in upper-left plot) (see **Figure 5** below). Additional information from the calibration, such as the observed shift in x and y position with axial position or wobble, may be useful when aligning the DH phase mask.
- 1.11 Select the most reliable calibration bead in the Easy-DHPSF main window using the drop-down menu. The calibration information and templates generated from this bead will be used in all later processing steps.
- 1.12 Go back to step 1.2 and repeat the calibration process for the other color channel. When done, click the save icon (located on the top left of the window) to save your calibrations. You can open this calibration file for use in the subsequent steps.

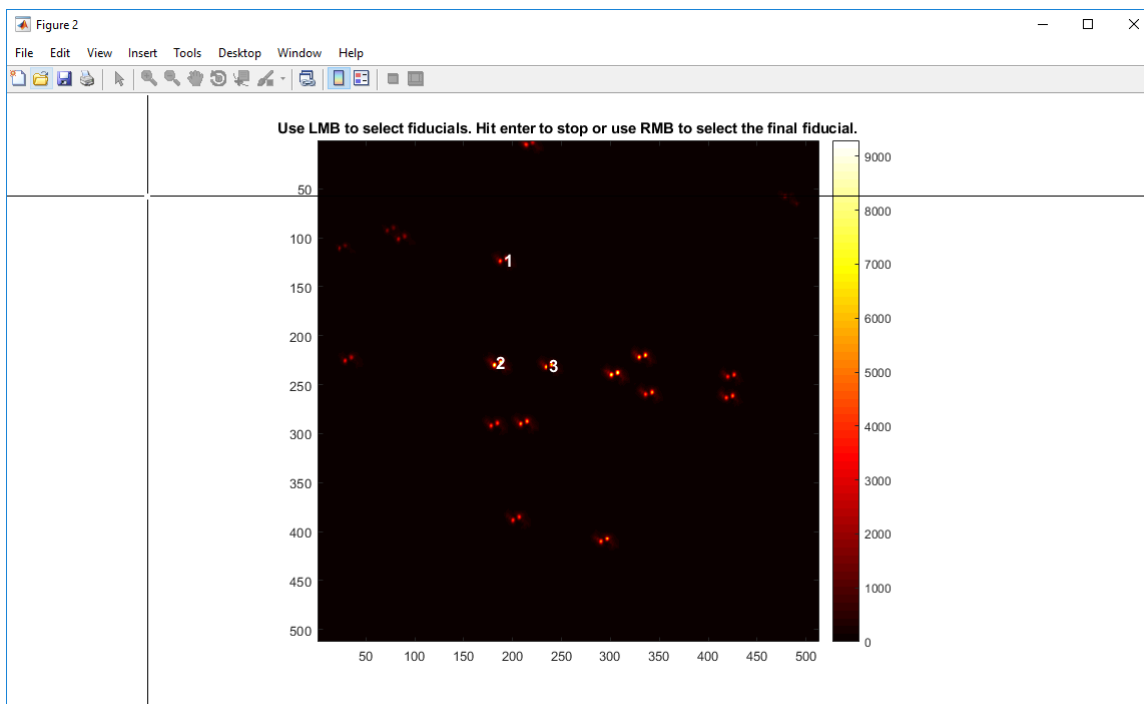


Figure 3. DHPSF calibration with beads. An example of the GUI that prompts user to select DHPSFs during the calibration step if beads are used for calibration. The program will prompt the user to select at least 1 bead.

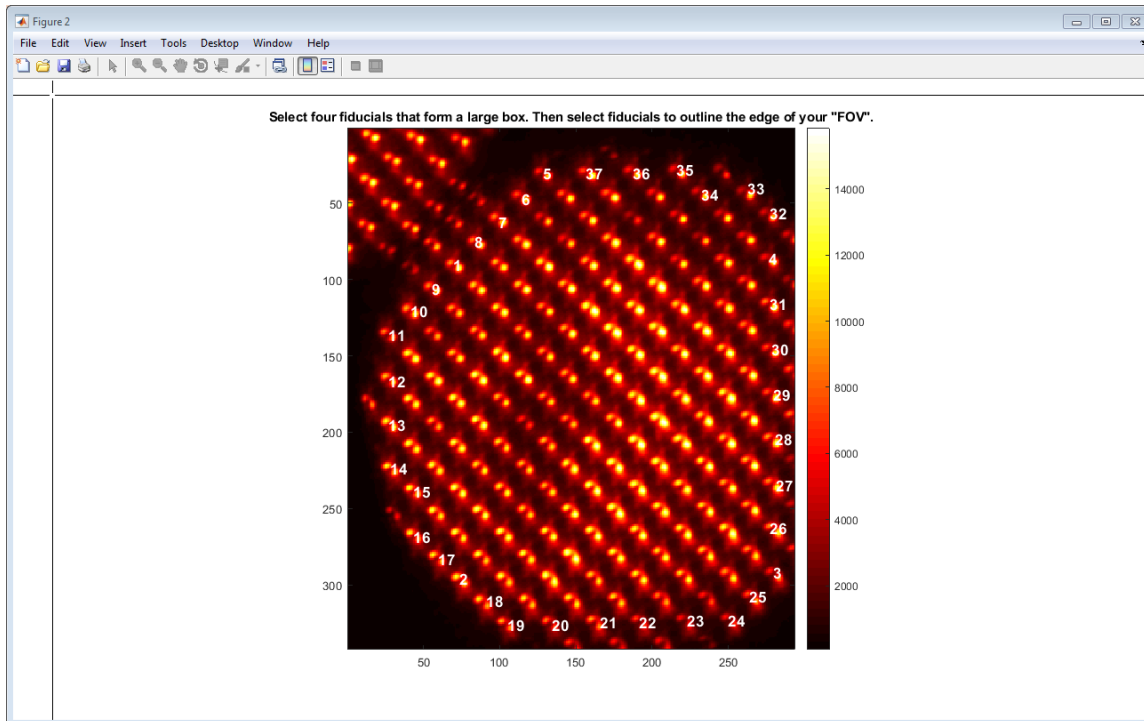


Figure 4. DHPSF calibration with NHA. An example of the GUI that prompts user to select DHPSF during the calibration step if an NHA is used for calibration. The program will prompt the user to select four fiducials that form a square (DHPSFs 1-4 in the image above), then to select fiducials that out the edge of the field of view.

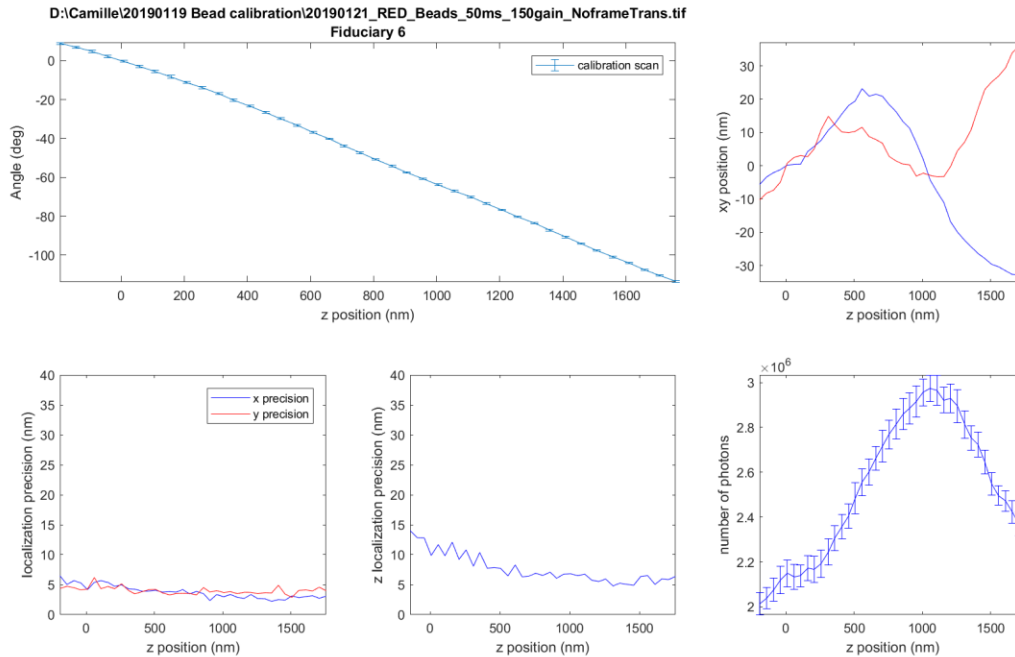


Figure 5. DHPSF calibration statistics. For a good calibration standard, the calibration curve (upper-left) should be roughly linear, the xy apparent motions (upper-right) should deviate less than ± 50 nm, and the localization precision (lower-left, lower-middle) in x, y, and z should be ≤ 15 nm over the course of the scan.

Step 2. Single-molecule detection calibration for SMACM data

In this module, the templates generated from the calibration subroutine are used to generate a large array of matches to the raw SMACM image data. This step serves to assess what value of phase correlation is typical for a good match to an SM image. After this step, the user will define a threshold for each template such that only DHPSF images of reasonable quality will be analyzed with the double-Gaussian fitting algorithm.

- 2.1 Under Setup > Channel, select a channel 'G' for green or 'R' for red.
- 2.2 Click the 'Run' button in the 'Calibrate SM identification' panel.
- 2.3 Follow the prompts if you would like to automatically save the GUI after this module.
- 2.4 Set the EM gain used, load in the SM data .tif file(s), select what frames to perform the SM detection calibration on, and select the templates to be used. By default, templates are chosen by the program such that the angle of the line connecting the two lobes is given by $\{-60^\circ, -30^\circ, 0^\circ, 30^\circ, 60^\circ, 90^\circ\}$, and this selection should generally be appropriate (**Figure 6**).
- 2.5 Load in the dark offset .tif file with the same parameters as the SM data.

- 2.6 The program will then prompt the user to choose a background processing algorithm such as median filtering, wavelet filtering, no background subtraction, or select if the data is an NHA, **Figure 7**. Median filtering parameters should be filled in if choosing this option.
- 2.7 Select a region containing the SMs of interest. This exact ROI will be used for the single-molecule fitting module and ideally should not contain fluorescent objects that are much ($\geq 10x$) brighter than the SMs you wish to analyze. Template matches are indicated as circles drawn over the raw data. Stronger matches are drawn as larger circles, and each color corresponds to a different template match.
- 2.8 After the template-matching module is complete, open the 'threshold [DateAndTime]' folder. This contains a selection of .png files that represent potential template matches. The filenames describe which template matched the data and the value of the phase correlation for the match (e.g. 'template 3 threshold 272.png').
- 2.9 Select appropriate thresholds for each template, and enter these into the 'threshold' field in the GUI. These thresholds should be chosen such that the two lobes of the DHPSF are faintly visible and that there is a low rate of false matches for higher correlation values. You should not be concerned if there are a few incorrect matches, say to one lobe of the DHPSF from a very bright molecule, because these will be rejected by the subsequent double-Gaussian fit module.

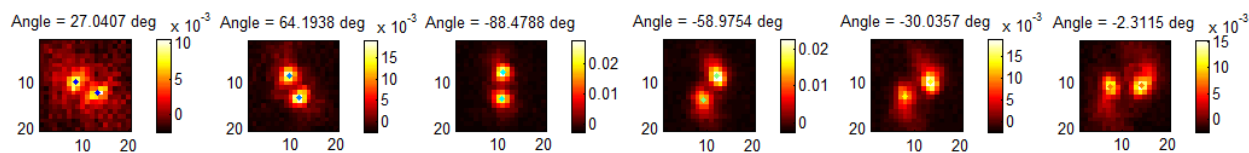


Figure 6. Typical DHPSF templates. Each template is chosen from the aforementioned calibration scan, where the rotation of the double helix between templates is evenly spaced at $\sim 30^\circ$.

Input filtering options _ _ X

Use (M)edian filtering, (W)avelet filtering, treat as (N)HA?, or n(O) subtraction?

If median filtering, what sigma size in pix (0 for no smoothing)

If median filtering, what total window size in frames?

If median filtering, number of frames to "interpolate"

When fitting data, display raw (not background-subtracted) data?

Figure 7. Input filtering options for SM calibration.

Step 3. Fiducial tracking (optional)

This module tracks the movement of one or more fluorescent beads or other stationary markers. Stage drift during a SM experiment can be thus be removed by subtracting the movement of the fiducial marker from the SM localizations.

- 3.1 Click the ‘Run’ button in the ‘Track fiduciaries’ panel.
- 3.2 Follow the prompts if you would like to automatically save the GUI after this module.
- 3.3 Select a region of interest containing at least one bright fiducial by adjusting the box and double-clicking inside it.
- 3.4 Select one or more fiducials by clicking in the center of each DHPSF.

Step 4. Single-molecule localization for SMACM data

Using the results of the previous processing steps, this module identifies single molecules that score above the template matching threshold identified in step 2.9, then fits them using nonlinear least squares minimization to a double-Gaussian function.

- 4.1 Click the ‘Run’ button in the ‘Localize DHPSF SMs’ panel.
- 4.2 Follow the prompts if you would like to automatically save the GUI after this module.
- 4.3 Choose the frames from the raw data to process, as well as whether you would like to estimate the laser profile and find true lobe widths (sigmas). By default, the entire image stack is processed.
- 4.4 Whenever starting analysis of a new experiment, monitor this process as it proceeds to ensure that the threshold used includes the single molecules visible in the raw data, and to ensure that the fits are performed successfully. Only matches that score above the thresholds are fit, and good fits are plotted in the frame-by-frame reconstruction. Here the definition of a good double-Gaussian fit is a fit that meets several straightforward criteria, hard-coded into the module. The tests performed to define a good fit are listed below under Troubleshooting (Step 4).
- 4.5 Save this Easy-DHPSF file using the save icon at the top of the GUI, ensuring that the Channel selected is the correct one.
- 4.6 Go back to Step 2, this time selecting the other color channel.
- 4.7 Ensure that you have two saved easy-dhpsf files, one for each color.

Step 5. Single-molecule detection calibration for control points

- 5.1 Load in the Easy-DHPSF GUI file from Step 1, saved after calibrating both the green and red channels.

- 5.2 Under Setup > Channel, select a channel 'G' for green or 'R' for red.
- 5.3 Click 'Run' in the 'Calibrate SM identification' panel.
- 5.4 Follow the prompts if you would like to automatically save the GUI after this module.
- 5.5 Set the EM Gain used, load in the control point scan .tif file(s), select what frames to perform the control point detection calibration on, and select the templates to be used. A subset of the frames may be used, but ensure that the frames you selected are ones that include at least 1-2 full axial scans so that different z positions are included in the calibration.
- 5.6 Load in the sequence log file (.dat or .mat) with the sequence of steps used for the control point scan.
- 5.7 Load in dark offset .tif with the same parameters as the control point scan.
- 5.8 The program will then prompt the user to choose background median filtering, wavelet filtering, no background subtraction, or select if the data is a NHA array. Median filtering parameters should be filled in if choosing this option. If an NHA is used, select NHA so that the program will fit localizations found in a grid – this will increase the number of PSFs detected. If NHA is selected, median background subtraction is performed.
- 5.9 A frame of the scan will pop up, and the user will be prompted to select an ROI of the correct color channel using a square ROI. The user will then be able to select an ROI with a polygon.
- 5.10 After the calibration, follow Steps 2.4-2.5 described above for selecting thresholds.

Step 6. Single-molecule localization for control points

This will outline the steps to fit control points in the two channels.

- 6.1 Click the 'Run' button in the 'Localize DHPSF SMs' panel.
- 6.2 Choose the frames you would like to fit, as well as whether you would like to estimate the laser profile and find true lobe sigmas.
- 6.3 As this step is for fitting the control points, enter how many stationary frames there are for each axial position and click 'OK.'
- 6.4 If using an NHA, select 4 fiducials that form a box that outlines the field of view (**Figure 8**). The program will find the rest of the DHPSFs located on a grid for each axial scan. Otherwise, this step does not show up for bead fitting (**Figure 9**).
- 6.5 Once the program is finished fitting, filter and export your results. Under 'Output DHPSF SM localization', click 'Filter Output.'

- 6.6 If using a NHA, choose 'Yes' when the program asks whether you would like to use spatially dependent calibrations. Doing this will reduce field-dependent aberrations by applying nearest-neighbor calibrations to your data with the NHA.
- 6.7 A different calibration may be used during this step. The program will prompt the user with this option.
- 6.8 Filter your control points as prompted by the program (for instance, filter by frame, z position, lobe distance, photons, localization precision, etc.).
- 6.9 Choose the region to be plotted in the scatter plot, inspect the localizations, then save your data when you are happy with the results. The filtered localizations will be in the file 'Output.mat', for use in the next step.
- 6.10 Repeat steps 5 and 6 for the other color channel. Take note of where you saved the filtered output files, as they will be needed for the next step.

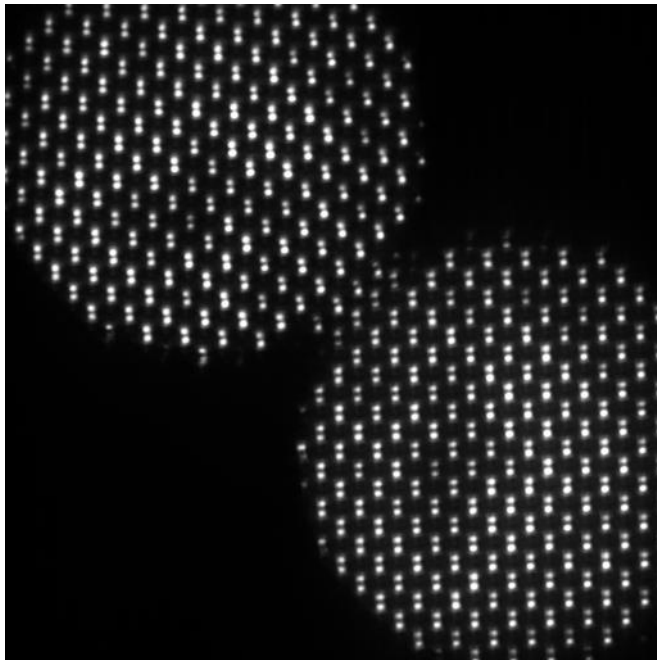


Figure 8. Example still image of a NHA during a control point scan. The green channel is on the upper left corner, and the red channel is on the lower right corner.



Figure 9. Example still image of beads on a coverslip during a control point scan. Still image of a field of view of fluorescent beads in the green (left) and red (right) channels during a control point scan.

Step 7. Identification of control points

This module will identify candidate control points between two channels.

KEY INPUTS: Output.mat file of GREEN and RED channels

KEY OUTPUTS: Identify_ControlPoints_3D_output.mat

- 7.1 Run the MATLAB function 'Identify_ControlPoints_3D_v1.m'
- 7.2 Open the 'Output.mat' data file with the molecule fits in reflected/GREEN channel. (If you are not using a single camera, ignore "reflected" vs. "transmitted".)
- 7.3 Open the 'Output.mat' data file with the molecule fits in transmitted/RED channel.
- 7.4 Open the .mat or .dat sequence log file for the control point scan.
- 7.5 If available, open a previous transform to use as an initial guess. Otherwise, click 'Cancel'
- 7.6 Input parameters of how many stationary frames were taken at each z position, how many frames to wait before starting the averaging, and if using an old transform, how closely the initial transform should match old transform points. The program will prompt you for these values.
- 7.7 The program will first find control point (CP) candidates, merging localizations at given z positions.
- 7.8 A figure will pop up with fitting statistics for the molecules fits you loaded. Enter filtering parameters for σ_x , σ_y , σ_z , and number of measurements.

- 7.9 Inspect averaged bead positions in both channels.
- 7.10 If an initial transform was not given in step 3.5, the program will ask you to hand click control points. Aim to hand click at least ~15 control points. Follow the prompts in the figure (**Figure 10**, **Figure 11**). Both the color and size of each localization encodes the relative brightness (number of photons detected) of that molecule. If the program is still prompting you to select control points but you have already selected >15 and would like to stop, get a new frame then hit 'esc' to end.
- 7.11 When finished hand clicking control points, the program will run and generate a preliminary transformation file that you will use for the next step. This file will be located in the same directory where your green channel 'Output.mat' is saved.

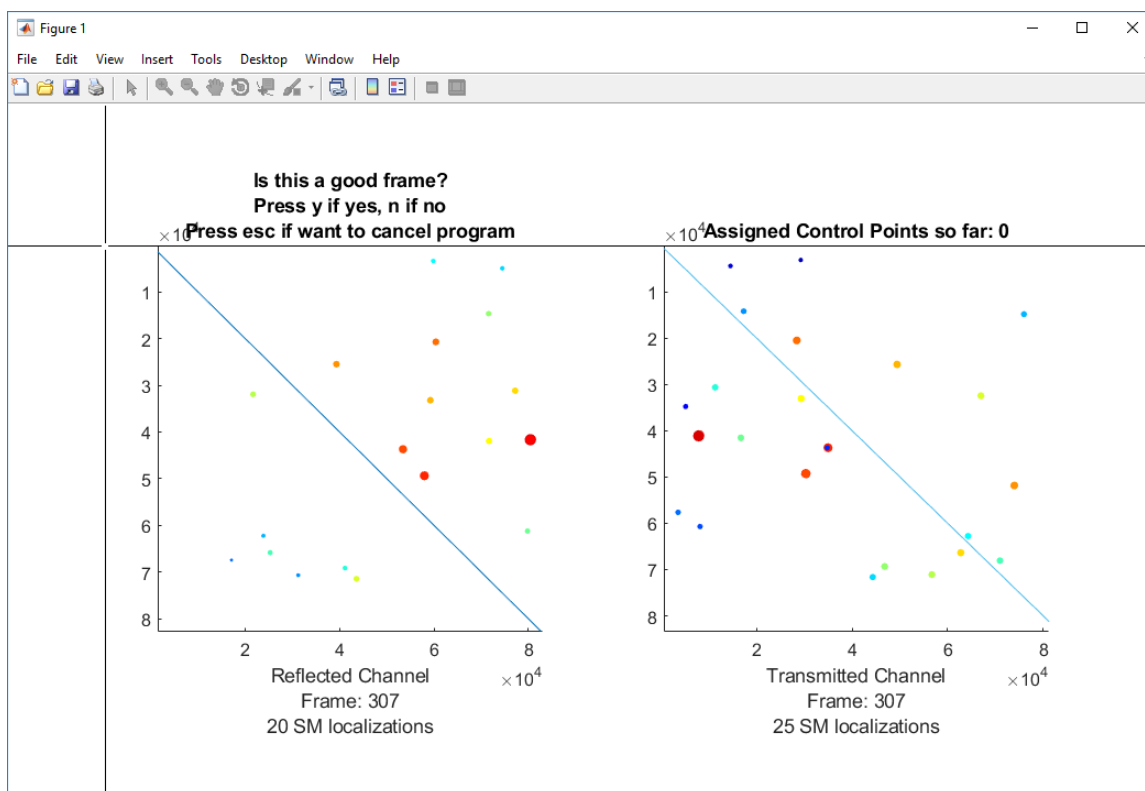


Figure 10. Example of the GUI that prompts user to select control point pairs when beads are used. There is a reflection between the reflected (green) and transmitted (red) channels in this example due to the optics needed to place both colors on the same camera. The light blue $y = x$ line is added as an additional visual reference.

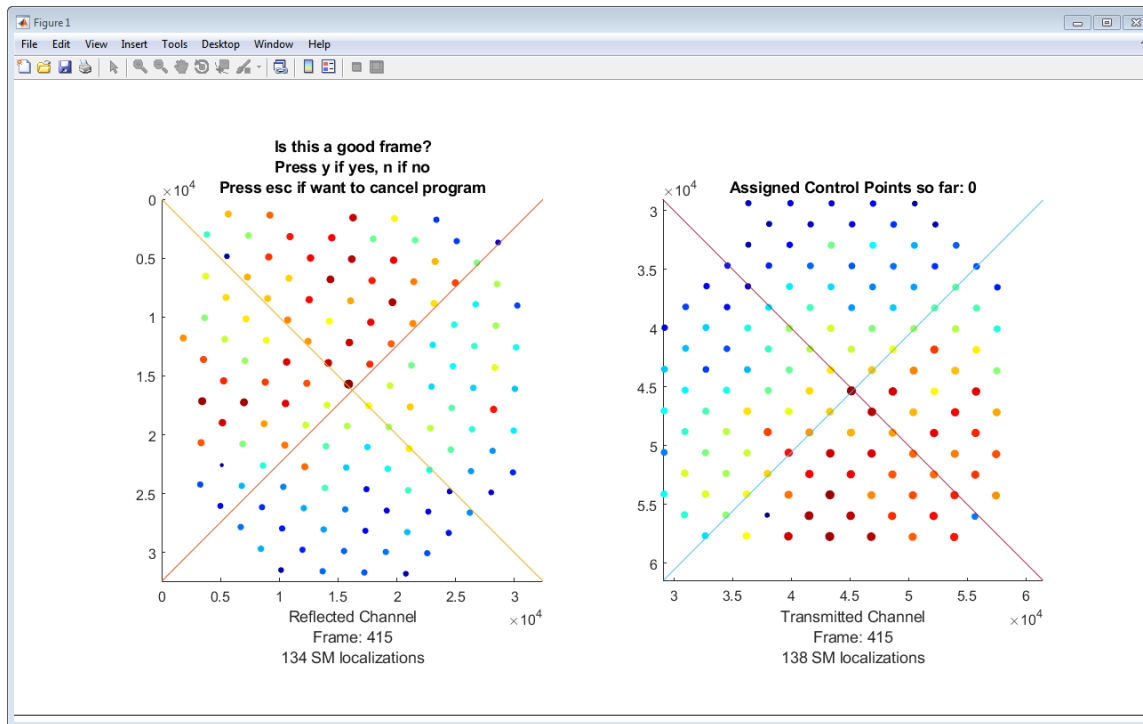


Figure 11. Example of the GUI that prompts user to select control point pairs when an NHA is used. There is a reflection between the reflected (green) and transmitted (red) channels in this example.

Step 8. Generating and evaluating final 3D transform from control points

This module will evaluate the initial transform calculated in Step 7 above and generate a final transform you can use in the next step.

KEY INPUTS: Identify_ControlPoints_3D_output.mat

KEY OUTPUTS: 3D_Transform_lwquadratic.mat

- 8.1 Run the MATLAB function 'EvaluateTransform_3D_v2.m'
- 8.2 Input transformation parameters (then proceed to Step 8.4) or click if you want to evaluate parameter ranges (proceed to Step 8.3).
- 8.3 Enter the range of parameters you would like to evaluate. The program will evaluate these parameter ranges then return the calculated target registration errors (TRE) and fiducial registration errors (FRE)¹⁰. An example result is in **Figure 12**. Inspect the results by clicking on the figure. Ideally, the user should aim for errors ≤ 10 nm.
- 8.4 Select the value of nearest neighbors, sigma, and kth nearest neighbor, and input these parameters into the dialog box that appears. The program will then calculate a transformation using the parameters you provided. For the ~1000 control point pairs in the example here, the mapping function took ~1 hour to generate using an Intel Xeon E5-1620 processor.

- 8.5 The program will return a histogram of TREs and FREs, as well as the spatial distribution of these metrics. The final transformation is '3D_Transform_lwquadratic.mat' and will be located in the same folder as 'Identify_ControlPoints_3D_output.mat.'

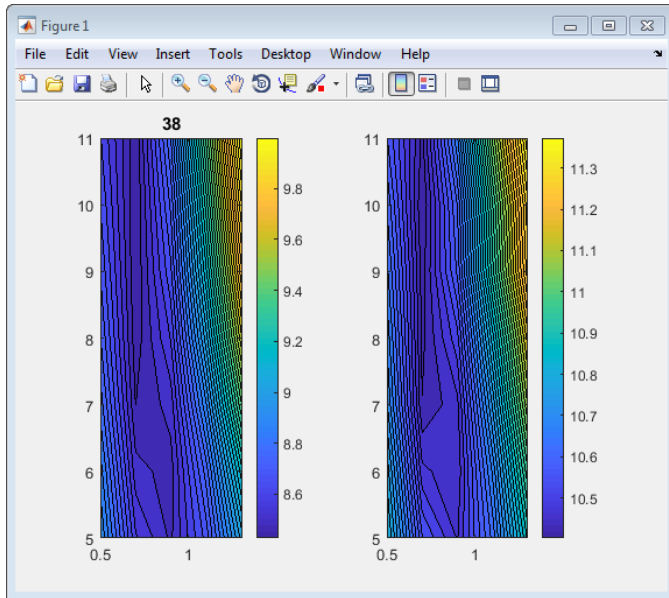


Figure 12. Example output from evaluating parameter ranges for transformation.

Step 9. Transforming SMACM data

This module will smooth the fiducial tracks, perform the 3D two-color registration, correct for sample drift, apply an index mismatch correction, and calculate localization precision.

KEY INPUTS: Easy-DHPSF files for each of the two color channels, transformation function (3D_Transform_lwquadratic.mat)

KEY OUTPUTS: MATLAB structure called dataSets

- 9.1 Run the MATLAB script 'TransformSMACMData_v6.m'
- 9.2 Indicate whether the green data or red data was acquired first, or if the two-color acquisition was interleaved.
- 9.3 Load in the green Easy-DHPSF file. The program will perform smoothing via wavelet filtering of the fiducial tracks and apply this correction to your data. Follow the prompts that appear.
- 9.4 Load in the red Easy-DHPSF file, and follow prompts that appear to smooth fiducial track data.
- 9.5 Load in the '3D_Transform_lwquadratic.mat' file generated in the previous step.

- 9.6 The program will prompt whether or not you would like to crop each dataset before the transformation.
- 9.7 Select the location and enter a file name for where to save your registered dataset.
- 9.8 The registered data is stored in the MATLAB structure 'dataSets,' with 'dataSets(1)' containing the green data and 'dataSets(2)' containing the red data. The fields xFid, yFid, and zFid_n contain the fiducial-corrected and index-corrected localizations. The rest of the fields are described below.
- 9.9 Select whether you would like to export the localizations into a csv file. If yes, choose the location and file name of each color file.

field	dataSets(1)	dataSets(2)
frame	Frame in which this localization was detected	
matchConf	Match confidence for initial peak finding	
amp1	Amplitude for lobe 1 from double Gaussian fitting	
amp2	Amplitude for lobe 2 from double Gaussian fitting	
xLoc1	x location of lobe 1 (pixel)	
yLoc1	y location of lobe 1 (pixel)	
xLoc2	x location of lobe 2 (pixel)	
yLoc2	y location of lobe 2 (pixel)	
sigma1	Sigma for lobe 1 from double Gaussian fitting	
sigma2	Sigma for lobe 2 from double Gaussian fitting	
bkgndMean	Mean background photons per pixel for this localization	
fitError	Total error from double Gaussian fitting	
goodFitFlag	Exit flag from double Gaussian fitting	
xCenter	Midpoint of two lobes, x (nm)	
yCenter	Midpoint of two lobes, y (nm)	
angle	Angle of DHPSF	
photons	Photons detected for this localization	
interlobe	Distance between two lobes of DHPSF	
ampRatio	Ratio of amplitudes of two lobes	
sigmaRatio	Ratio of sigma values of two lobes	
x	x localization (nm)	
y	y localization (nm)	
z	z localization (nm)	
LocFile	File location of the Easy-DHPSF file	
LocPath	File path of the Easy-DHPSF file	
xFid	Final x localization after fiducial correction	
yFid	Final y localization after fiducial correction	
zFid	Final z localization after fiducial correction	

fidTrackX	Denoised x fiducial track (nm)	
fidTrackY	Denoised y fiducial track (nm)	
fidTrackZ	Denoised z fiducial track (nm)	
xTrans	Empty	x location (nm) after transformation
yTrans	Empty	y location (nm) after transformation
zTrans	Empty	z location (nm) after transformation
zFid_n	Final z localization after fiducial correction and a simple (multiplicative) index mismatch correction	
sigmaX	Estimate of x localization precision (nm)	
sigmaY	Estimate of y localization precision (nm)	
sigmaZ	Estimate of z localization precision (nm)	

Anticipated results

Below is a 2D scatterplot of Alexa647- and CF568-immunolabeled tubulin in a BSC-1 cell as measured by SMACM blinking of the labels (**Figure 13**). Both colors are labeling the same structure which contains microtubules in a thin region of the cell. Note that this 2D scatterplot representation of 3D data is quick to render for inspection purposes but does not show depth (z) or density, overemphasizing sparse regions of the sample and often appearing noisier than a density plot. To avoid misinterpreting results, we strongly recommend exporting your two-color localizations to a different program for better visualization with a more reasonable reconstruction, such as that shown in **Figure 14**.

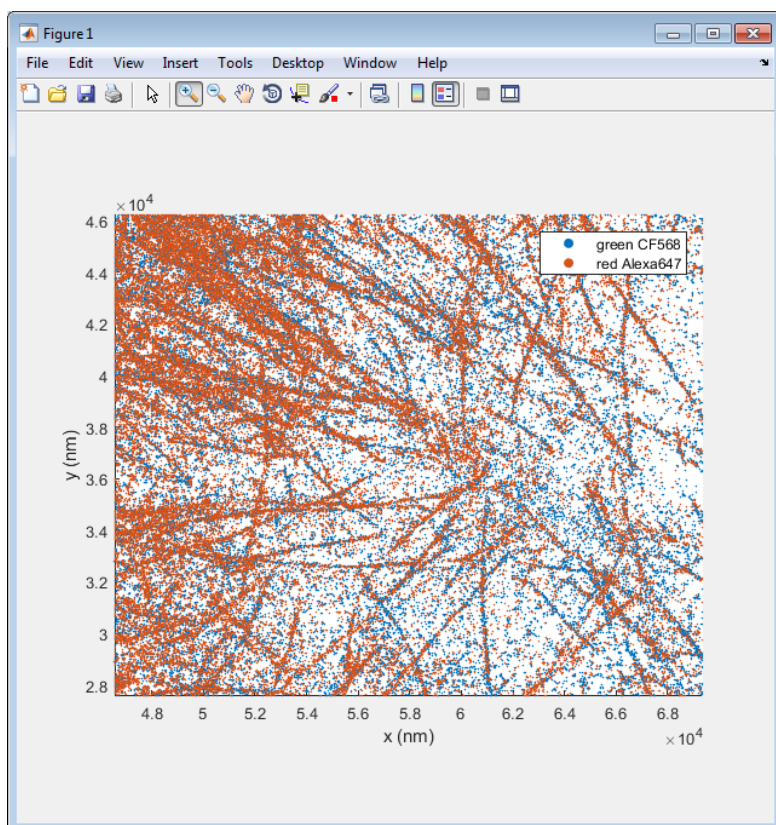


Figure 13. 2D scatterplot of two-color tubulin localizations in a fixed BSC-1 cell.

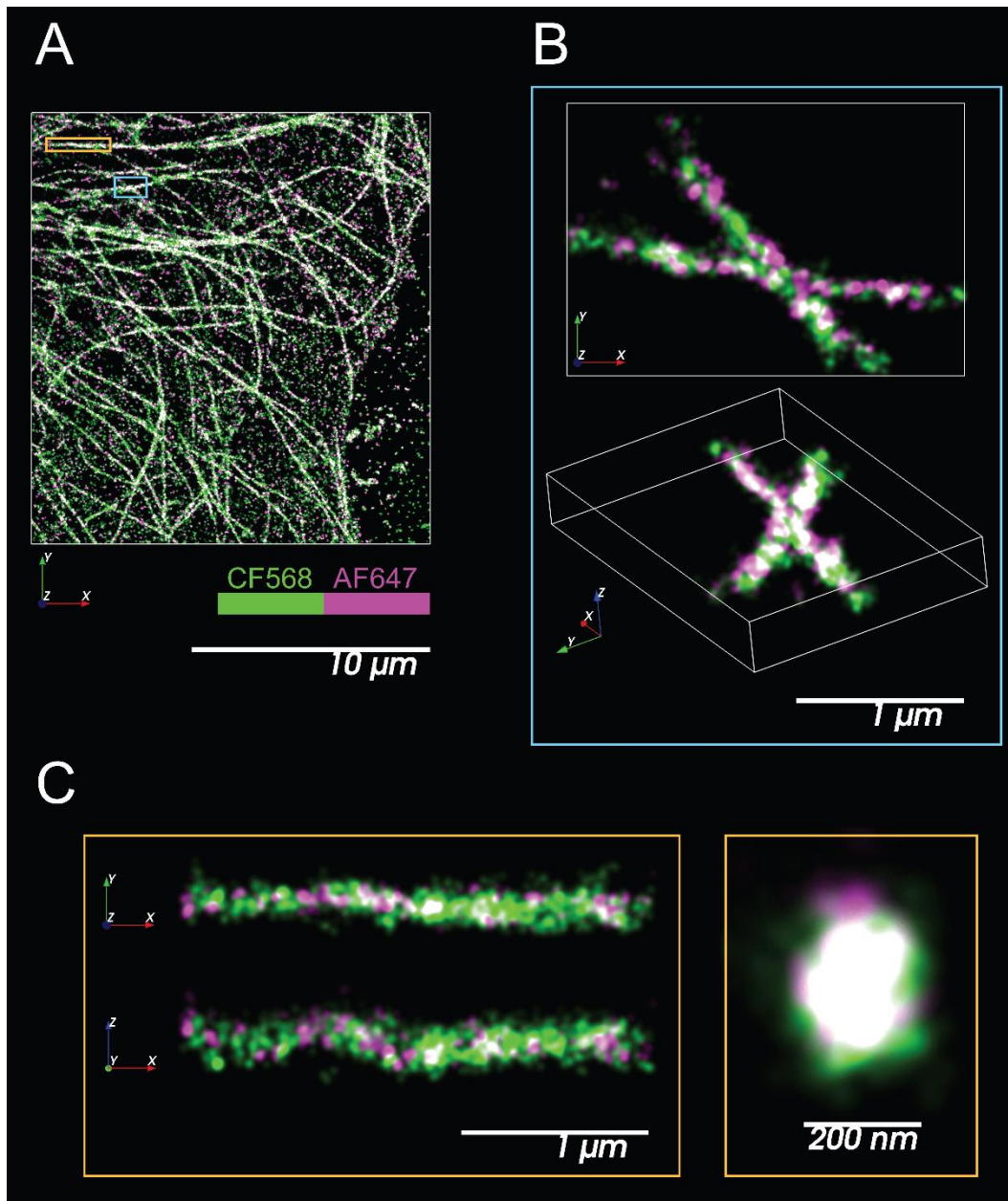


Figure 14. Two-color SR imaging of double-labeled microtubules, rendered using the program Vutara SRX.

A, 2D projection of 3D microtubule localizations in a fixed BSC-1 cell. Green indicates CF568 localizations and magenta indicates AF647 localizations. White indicates colocalization. Each localization is plotted as a 3D Gaussian with 50 nm width. **B**, 2D projection of blue region in **A** (top), and 3D perspective view of blue region (bottom). **C**, xy and xz projections of orange region in **A** (left), and a projection down the axis of this same tubule in (right)

Associated publications

1. Andreas Gahlmann, Jerod L. Ptacin, Ginni Grover, Sean Quirin, Alexander R. S. von Diezmann, Marissa K. Lee, Mikael P. Backlund, Lucy Shapiro, Rafael Piestun, and W. E. Moerner, “Quantitative Multicolor Subdiffraction Imaging of Bacterial Protein Ultrastructures in Three Dimensions,” *Nano Lett.* **13**, 987-993 (2013), published online February 15, 2013.
2. Mikael P. Backlund, Ryan Joyner, Karsten Weis, and W. E. Moerner, “Correlations of three-dimensional motion of chromosomal loci in yeast revealed by the double-helix point spread function microscope,” *Mol. Biol. Cell* **25** (22) 3619-3629 (2014) (DOI: 10.1091/mbc.E14-06-1127, published online October 15, 2014).
3. Alex von Diezmann, Maurice Y. Lee, Matthew D. Lew, and W. E. Moerner, “Correcting field-dependent aberrations with nanoscale accuracy in three-dimensional single-molecule localization microscopy,” *Optica* **2**, 985-993 (2015) (DOI: 10.1364/OPTICA.2.000985, published online November 19, 2015).
4. Camille A. Bayas, Jiarui Wang, Marissa K. Lee, Jared M. Schrader, Lucy Shapiro, and W. E. Moerner, “Spatial organization and dynamics of RNase E and ribosomes in *Caulobacter crescentus*,” *Proc. Nat. Acad. Sci. (USA)* **115**, E3712-E3721 (2018). (DOI: 10.1073/pnas.1721648115, published online April 2, 2018).

Acknowledgements

We thank Joshua Yoon for his contributions to the code, and Dr. Carl G. Ebeling, Worldwide Application Scientist for Bruker Fluorescence Microscopy, for his support and for the use of the Bruker SRX visualization and analysis software for rendering localization data. This work was supported in part by the U. S. National Institute of General Medical Sciences Grant No. R35GM118067.

References

1. A. von Diezmann, Y. Shechtman and W. E. Moerner, *Chem. Rev.* 117, 7244 (2017).
2. M. A. Thompson, Matthew D. Lew, M. Badieirostami and W. E. Moerner, *Nano Lett.* 10, 211 (2010).
3. A. S. Backer, M. P. Backlund, A. R. von Diezmann, S. J. Sahl and W. E. Moerner, *Appl. Phys. Lett.* 104, 193701 (2014).
4. S. R. P. Pavani, M. A. Thompson, J. S. Biteen, S. J. Lord, N. Liu, R. J. Twieg, R. Piestun and W. E. Moerner, *Proc. Natl. Acad. Sci. U. S. A.* 106, 2995 (2009).
5. Matthew D. Lew, Alexander R. S. von Diezmann and W. E. Moerner, *Protocol Exchange* 026 (2013).
6. Alex von Diezmann, Maurice Y. Lee, Matthew D. Lew and WE Moerner, *Optica* 2, 985 (2015).
7. A. A. Goshtasby, Image registration, (Springer-Verlag, London, 2012), p. 441.

8. Andreas Gahlmann, Jerod Louis Ptacin, Ginni Grover, Sean Quirin, Alexander R. S. von Diezmann, Marissa K. Lee, Mikael Paul Backlund, Lucy Shapiro, Rafael Piestun and W. E. Moerner, *Nano Lett.* 13, 987 (2013).
9. C. A. Bayas, J. Wang, M. K. Lee, J. M. Schrader, L. Shapiro and W. E. Moerner, *Proc. Natl. Acad. Sci. U. S. A.* 115, E3712 (2018).
10. L. Stirling Churchman and James A. Spudich, *Cold Spring Harbor Protocols* 2, 141 (2012).

Citing this work

If you use Easy-DHPSF for scientific work, please cite our protocol:

Camille Bayas, Alex von Diezmann, Anna-Karin Gustavsson et al. Easy-DHPSF 2.0: open-source software for three-dimensional localization and two-color registration of single molecules with nanoscale accuracy, 10 June 2019, PROTOCOL (Version 2) available at Protocol Exchange [+<https://doi.org/10.21203/rs.2.9151/v2>]