**Frequency Multiplexed STORM Demodulation Software Guide**

This workflow explains how to use the software. We use DAX images from an Andor EM-CCD Camera and Insight3 (from Bo Huang) for the localization of the single molecules. The software could be adapted to accept other file types for the images and localization lists.

The DAX files should contain an info file (.inf) with the same name, where the number of frames, the laser modulation frequencies, the camera frame rate and image size is specified. An example of this file is also attached.

**1. Define the folders with the experimental data, training data sets and z-calibrations for different fluorophores (if using 3D):**

The config.cfg file contains the directories of all those folders. In particular, “data directory” contains the experimental data to classify, “cal#dye directory” contains the training data sets for each fluorophore. “cal#z directory” contains the z-calibrations for each fluorophore. Modify the config.cfg file to ensure that all nine lines have valid directories. The directories don't necessarily need to have anything useful in them, they just need to exist. If you are only using 2-colors, just fill the rest of the directories with something valid. They need to end with a “\”. For example:

data directory = W:\fmSTORM\Cell1\

cal1dye directory = W:\fmSTORM\TrainingData\AF647\

The inputs for the experimental data will be DAX files containing the RAW data images, imaged with sine-wave illumination as described in the methods part of the manuscript. The cal#dye directories contain the training data sets. A DAX file needs to be selected for each fluorophore in use on the top right panel of the GUI. The input for the Z-calibration files will be text files containing the values for the parameters used to determine the z-position for each fluorophore (see **section 6** for the format).

**2. Execute the main.py file:**

Execute main.py file in Phyton. Some libraries are required: tkinter, importlib, os, numpy, shutil, matplotlib.pyplot, distance from scipy.spatial, optimize from scipy, default\_timer from timeit, ShuffleSplit from skelearn.model\_selection and SVC from skelarn.svm). The rest of them are custom ones and are already included with the software.

This will pop up a user-friendly GUI for using the software. All the directories can be changed directly in the GUI, but the user needs to click on the button “Reload directories” every time that a modification is introduced.

**3. Demodulate each one of the training data sets:**

The first step will be to demodulate all the training data sets, for each fluorophore.

-In “data directory” input the folder corresponding to the training data sets (one by one). Remember that at the end of the directory a “\” is required.

- After changing the directory, click on "Reload directories" (at the bottom of the window).

- Select the proper entry from the drop-down "Data file" field. Here you can chose the DAX file that you want to demodulate from the selected folder.

- Input the correct number of frequency bins. The frame window size (m) will be the double of the number of available frequency bins. For example, for 3 frequency bins, the frame window size will be m=6.

- Select the appropriate modulation frequencies next to "Active lasers". Here you define the number of lasers that you used and in which frequency bins you allocate their modulations. (See Supplementary Material for calculation of the appropriate frequency bins). For example, for a camera frame rate F=60Hz, and 3 frequency bins, the modulation frequencies will be 30Hz, 20Hz and 10Hz.

- (Optional) Input the wavelength of each laser and the corresponding name of the fluorophore. This is just a label.

- Click on "Reload directories". In general, any time you change any field in the program you should click this button.

- Remove background. This applies a median filter to the DAX file and removes the background. It creates a new file in the directory with the same name and the suffix: “\_no\_bg”

- Localize the background-free data in Insight3 or other appropriate localization software. This step is performed outside the Python code, in the stand-alone program Insight3 or other appropriate localization software. Insight3 output is a .bin list with the localizations. It is a table with Nx18, where N is the number of localizations. The columns are organize like this:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| x | y | xc | yc | height | area | width | phi | aspect |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| background | Intensity | channel | iterations | frame | trackLength | link | z | zc |

For each localization, the Python software retrieves the spatial coordinates (x and y), in columns 1 and 2, respectively, the height of the Gaussian fit (height), in column 5, the camera frame index (frame), in column 12 and for 3D measurements the aspect ratio (aspect; a=wx/wy), in column 9 and the effective width (width; w = wx\*wy), in column 7. The rest of the parameters in the localization list are not used as inputs.

After the SVC step (**section 5**), the software will add a value to the channel, in column 12. The number of channels will be equal to the number of modulation lasers in use. After the z position calculation of localizations (**section 6**), the software will add a z coordinate to the list, in column 17 for each localization.

- Categorize localizations into single- of multi-frame localizations, from the Insight3 localization list. You can define the distance threshold from frame to frame in pixels of a molecule to be considered the same molecule. In the manuscript, we used 80 nm. This step will output two .bin lists with the single- and multi-frame localizations

- (Optional) Read a single localization frame to ensure that the previous steps were successful. If an empty array [] is printed that is all right, it just means there were no localizations in that frame. You can define the Frame to read (frame number) for this step.

- Demodulate at each localization. This will operate on both the single- and multi-frame data sets by performing the Discrete Fast Fourier Transform and saving the intensities in the frequency domain for all the frequency bins in use. It will create files with those intensities for single- and multi-frame localizations. Those files will be .ints files with the suffixes “indiv\_single\_list.ints” and “indiv\_multi\_list.ints”.

**4. Demodulate the experimental data:**

Now that all of the calibration data sets have been processed, it's time to move to the real multi-color experimental data. All of the processing steps for the demodulation of the real data are the same as above (**step 3**). We will obtain all the output files, in the same folder, containing the intensities in the frequency domain, for the different frequency bins. Remember that the localization step must be performed on the background-free DAX file, if not, the program will produce errors.

**5. Support Vector Classification of the fluorophores:**

At this point, you should re-edit your config.cfg file to include the proper paths to the calibration files. Note that the order is important. In the data directory, you should select the folder containing the real data with all the output files for the intensities in the frequency domain of single- and multi-frame localizations obtained in previous steps.

Now the software will classify the fluorophores of the real data:

- Ensure that you have the correct number of frequency bins, active lasers, and wavelength/fluorophore fields selected.

- Click on "Reload directories".

- Select a rejection ratio of 1.0 to start. This number now represents the (approximate) percentage of localizations that are used, so 1.0 = 100%, 0.9 ~ 90%, etc. Lower numbers result in less crosstalk, but fewer localizations (see Supplementary material). Then you can reduce that value accordingly with your needs and check the results. In the manuscript we used a threshold value of 0.96.

- Train the SVC. You will get two critical pieces of information after this step. First, a new window will appear that contains a scatter plot of the calibration localizations overlaid on the decision boundary from the SVC. This gives a visual representation of how 'good' the training is. Second, in your Python console window you will see the error rate from the calibration data on that decision boundary for each fluorophore as well as the number of fluorophores that have been used to train the SVC. Error rates of ~5% are acceptable, but you can play with the rejection ratio to increase or decrease the error rate according to your needs.

- (Optional) Reduce the rejection ratio until the main cluster of points on the equal-intensity line is eliminated.

- Test the SVC on the real data. This tests the SVC on the file selected above in “Data file”. First, select “single” for single-frame localizations and then “multi” for multi-frame localizations. The output of these steps are bin lists with the suffixes “indiv\_single\_SVC\_list” and “indiv\_multi\_SVC\_list”, respectively. Here the different colors are separated into different channels (i.e. Column 12 of the localization list now specifies the channel of the localization).

- Check the resulting localizations in Insight3 or appropriate visualization software. There should be very little crosstalk and the fluorophores should have been classified into the different channels.

- If you are happy with the results, test SVC on the single-frame data set. Note that different rejection ratios can used for single- and multi-frame localizations. Typically in single-frame localizations one needs to be more restrictive.

- Merge the single- and multi-frame localizations together. This will merge all the accepted localizations from single- and multi-frame lists into a single .bin list with a suffix “indiv\_test\_comb\_SVC\_list”

At this point you should have three new .bin files with '\_SVC\_list.bin' suffixes. These are the final outputs of the software.

**6. Use for 3D localizations:**

The appropriate z-calibration files for each fluorophore must be contained on the cal#z folders. Those values will be use according to Huang et al, Science, 20081. These are text files with the format below (without spaces), and values obtained experimentally with fluorescent beads for different colors:

wx0=310.48;zrx=313.46;gx=-185.24;Ax=-0.27752;Bx=0.01521;Cx=0.0000;Dx=0.000000; wy0=275.73;zry=426.75;gy=264.75;Ay=-0.39166;By=0.02283;Cy=0.0000;Dy=0.000000

For 3D application (PSF engineering approach, see Supplementary material) the same procedure should be followed, but:

- In the single molecule localization step with Insight3 in the background-subtracted data the astigmatic fit should be used. This will save two parameters in the localization list that contain the shape of the elliptical Point Spread Functions (see Supplementary material). Here, the ax and w values are written into the \*filename\*\_no\_bg.bin file.

After all the procedure of demodulation and classification (see **steps 3** and **5**):

- Calculate the real position of each localization on the merged list (appropriate z calibration files should be selected in lower-right panel). This step produces a new .bin file with the suffix: indiv\_Zcal\_comb\_SVC\_list.bin file that contain the calibrated z-position of each localization (i.e. Column 17 of the localization list now specifies the z coordinate of each localization).

1 Huang, B., Wang, W., Bates, M. & Zhuang, X. Three-dimensional super-resolution imaging by stochastic optical reconstruction microscopy. *Science* **319**, 810-813, doi:10.1126/science.1153529 (2008).