DeepFRET

A fully open-source, all-inclusive software platform for doing total internal reflection microscopy (TIRFm) single molecule FRET (smFRET) fast and efficiently. The key feature is reverse-loading of ASCII traces, and *Keras/TensorFlow-powered automatic trace sorting.* Features include

- Advanced trace sorting
- Optical correction factors
- Hidden Markov model fitting and lifetime plotting
- Memory-free movie batch loading
- Distribution plotting and fitting
- Simulation of smFRET data
- Backwards-compatibility with iSMS-exported data

If you'd like to play around with just the Keras/TensorFlow model, please go to https://github.com/komodovaran/DeepFRET-Model

How to cite

Publication coming soon!

1. Installing and running the software

Launching the DeepFRET GUI from executable

Download the pre-compiled version here.

(Note: currently only available for MacOS)

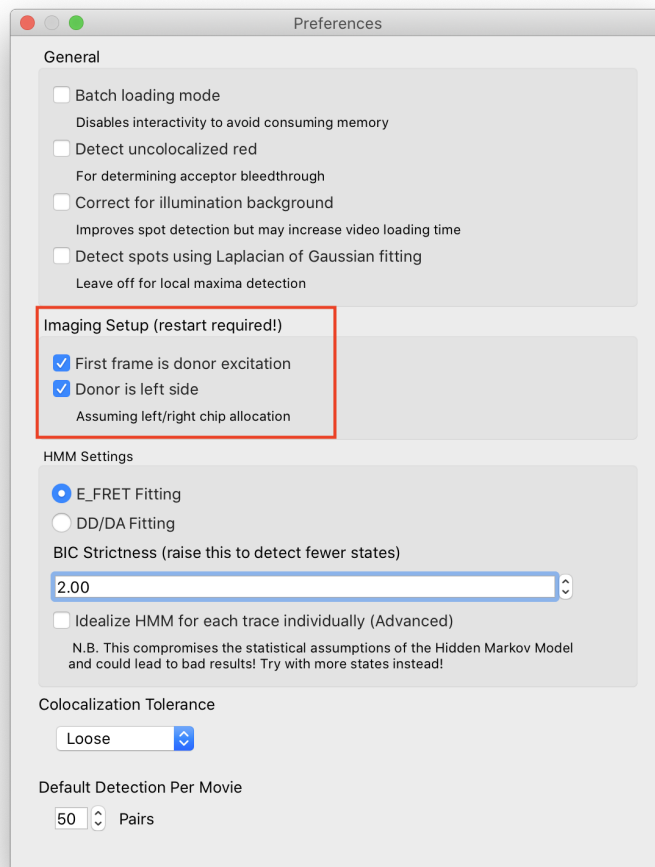### Launching DeepFRET GUI from Python source code.

1. Install Python 3.7.0. Other versions *might* work, but this is not guaranteed.

2. Download repository contents, by cloning the repository or unzipping the zip file, and open the folder in the terminal.

3. Create a virtual environment environment with `python3 -m venv venv` in the current directory.

4. Activate environment with `source venv/bin/activate` if on MacOS/Linux or `call venv\scripts\activate.bat` if on Windows.

5. While still in the environment, install all dependencies with `pip install -r requirements.txt`

6. Launch the GUI with `python3 src/main/python/main.py` or `python src/main/python/main.py`, depending on the setup of your environment.

2. Loading data

1. To load videos, open the application's main window ('Images')
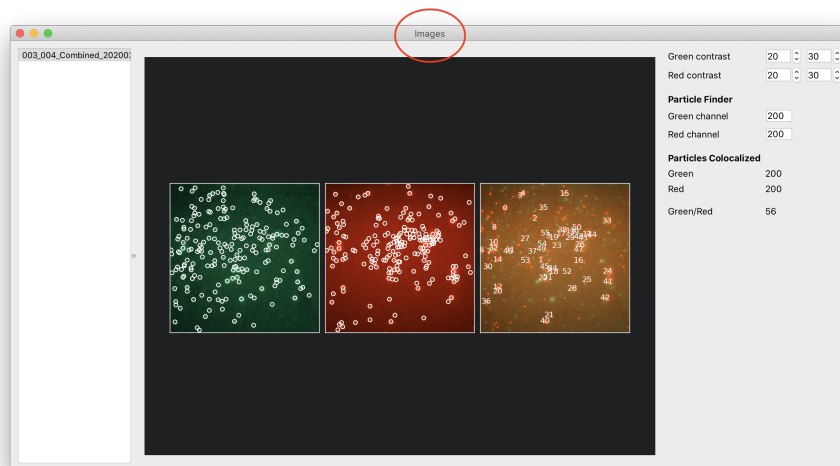
and go to File → Open files. The current version of DeepFRET supports *only* videos made with alternating laser excitation (ALEX). Donor order and appearance can be set in Preferences. The rest of the video layout is auto-guessed by the software and doesn't allow for fine-tuning, so be sure to set up new experiments properly!

Preferences

General

☐ Batch loading mode
  Disables interactivity to avoid consuming memory

☐ Detect uncolocalized red
  For determining acceptor bleedthrough

☐ Correct for illumination background
  Improves spot detection but may increase video loading time

☐ Detect spots using Laplacian of Gaussian fitting
  Leave off for local maxima detection

Imaging Setup (restart required!)

☑ First frame is donor excitation
☑ Donor is left side
  Assuming left/right chip allocation

HMM Settings

🔘 E_FRET Fitting
⚪ DD/DA Fitting

BIC Strictness (raise this to detect fewer states)

2.00

☐ Idealize HMM for each trace individually (Advanced)
  N.B. This compromises the statistical assumptions of the Hidden Markov Model
  and could lead to bad results! Try with more states instead!

Colocalization Tolerance

Loose

Default Detection Per Movie

50 Pairs

2. If you would like to extract traces from a large number of videos that don't fit into memory, you can tick the 'batch mode' option in Preferences, as well as set a number of detections per movie. This disables the option to interactively re-run spot detection analysis, but allows the software to process an essentially unlimited number of videos.

3. If you've already exported a lot of smFRET traces from another software, but want to do your analysis in DeepFRET, you can even load in traces directly, without movies. Simply make sure that the trace window ('Traces') is the active window, and follow the same steps as above for loading data. This works both for traces with and without ALEX.

3. Classifying the data

1. Go to Analyze → Predict to predict the trace type using deep learning. A confidence score will be given for each trace, which tells you how certain the model is that this is a true smFRET trace.

2. To sort traces by different things, go to the View menu. The option "advanced sort" includes, among other things, a lower confidence threshold.

Stoichiometry Median Range:	`0.00` `1.00`
FRET Median Range:	`0.00` `1.00`
Minimum Trace Confidence:	`0.75`
Minimum Dynamics Confidence	`0.00`
Minimum Number of Frames:	`15`
Trace Must Bleach:	☐
	Find

4. Statistical analysis

1. To get an overview of data distributions, go to Windows → Histogram. This also allows to fit the FRET distribution with gaussians to estimate the number of underlying conformational states. These plots update with the number of traces selected. Because a large number of operations have to be re-computed every time a trace is selected/de-selected, it can be a bit slow if the window is left open while traces are being selected.

2. Additionally, DeepFRET includes a Hidden Markov model with the possibility to fit each trace individually (as a smoothing, "step-finding" method), as well as a global fit option, for a true Hidden Markov model fit. This also automatically fits transition lifetimes and plots transition density plots (which can be found in Windows → Transition Density Plot). One can base the model on the donor/acceptor signal, or directly on the FRET signal (though this is less accurate)

5. Export options

Export plot automatically exports the current plot for each window. If desired, one can also export the data present in each window such as traces, histogram datapoints, etc. available from the File menu.

6. Modifying and compiling DeepFRET from source

Note that this is not necessary to make changes. You can still run the source without compiling it. This step is only to create a portable, standalone executable.

1. Follow all steps for launching DeepFRET, above.
2. Unzip the `hooks.zip` and overwrite the files in `venv/lib/python3.7/site-packages/PyInstaller/hook`
3. Run `compile.py` and wait until the script finishes.

If the above steps worked, you can now edit any part of the code, and re-compile it.

Most interface elements are arranged in the `.ui` files, and can be edited through Qt Creator and then converted to Python files with `generate_ui.py`.