locFISH v1.0.0

Matlab toolbox to study sub-cellular mRNA localization from simulations and experiments

Florian MUELLER, muellerf.research@gmail.com
Institut Pasteur Paris, Computational Imaging & Modeling Unit

Thomas WALTER, thomas.walter@mines-paristech.fr
Mines ParisTech, Centre de BioInformatique

Aubin SAMACOITS, aubin.samacoits@gmail.com
Institut Pasteur Paris, Computational Imaging & Modeling Unit

This toolbox has been developed for the following publication. Validations and results can be found therein.

Samacoits A, Chouaib R, Safieddine A, Traboulsi AM, Ouyang W, Zimmer C, Peter M, Bertrand E, Walter T, Mueller F. *A computational framework to study sub-cellular RNA localization*.

1 Overview of software package and workflow

In this section, we provide a brief overview of this package and its main functionality. The following sections provide more detailed information and instructions for how to use the provided source code.

This **Matlab package provides functionality to simulate and analyze sub-cellular mRNA localization**. The main features are (Figure 1):

- **Simulating realistic smFISH images** in 3D with different mRNA localization patterns.
- Provides Matlab scripts to **analyze simulated and experimental data**.
    - Single mRNA detection (from FISH-quant[1])
    - Decompose mRNA foci in individual mRNAs with a Gaussian Mixture Model.
    - Validated localization features to summarize mRNA localization in individual cells.
- Provide Matlab GUIs to perform clustering of cells based on their localization features.

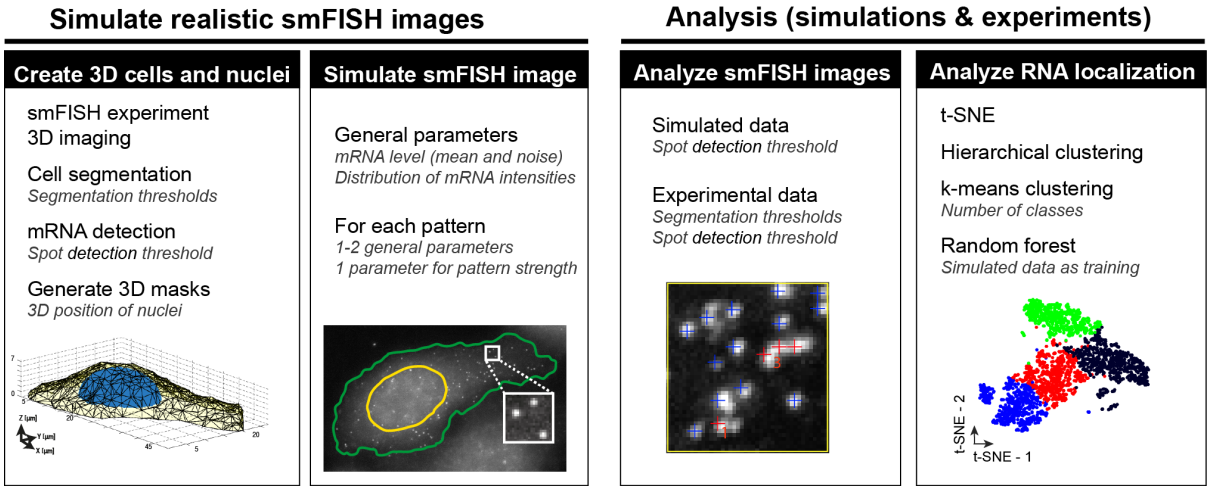


***Figure 1**. Overview of different modules to study intra-cellular mRNA localization. Listed are the main steps and in italic the most important user-defined parameters.*

1.1 Simulating realistic smFISH images

As detailed in the accompanying publication, we **use experimental data to obtain realistic cell shapes and image**. The experimental data serve two main purposes (1) allow an accurate description of cells and nuclei in 3D, (2) provide a realistic background smFISH background image. We provide already analyzed data for HeLa cells that can be used directly to start simulating images. They are provided as a library of cells with 3D masks for the cytoplasm and the nucleus, as well as the corresponding smFISH background. In the section "***Data needed to simulate smFISH images***", we describe how these data were obtained, and these data can be generated with the provided Matlab scripts. In short,

Once such a library is established, **smFISH images with different localization patterns can be simulated**. The section "***Simulating smFISH images***", describes in detail how these images can be generated, how the different parameters can be set, and what kind of data is stored.

1.2 Analyzing simulated and experimental data

We provide scripts to analyze simulated or experimental images. This analysis consists of three mains steps

(1) Detection the mRNA molecules,
(2) Calculation of localization features provide a description of the spatial mRNA distribution in each cell,
(3) Class cells based on similarity in their mRNA localization pattern.

The section "***Analyzing simulated smFISH images***" focuses on the analysis of simulated data, the section "**Analysis of experimental data**" on how to analyze and further interpret experimental data.

2 Getting started

The locFISH package is distributed together with FISH-quant and comes with a small test-data set that allows to immediately simulate cells. We provide a brief tutorial for how to get started and familiarize yourself with the different modules. More details about each step can be found in the dedicate chapters.

1. **Install** FISH-quant (see chapter ***Installation***).

2. **Simulate 3D cells** with different localization patterns

    1. Open the script `WRAPPER_simulate_smFISH_v1`
    2. To speed-up computation time you can change two parameters:
        a. Simulate fewer cells per condition: `param_sim.n_cell = 20;`
        b. Simulate only one expression level: `param_sim.mRNA_level.low = [200 100];`
    3. **Save and execute script**: Specify the folder where you want to save the images
    4. **Inspect the simulated images**
        a. Script will generate a folder for the expression level, containing subfolders for each pattern, containing subfolder for each pattern strength.
        b. Scripts saves the full 3D image, but also a **maximum intensity projection** (starting with MAX_). These files are convenient for a fast inspection with FIJI (https://fiji.sc/) or other image processing packages.

3. **Perform mRNA detection**

    1. Open the script `WRAPPER_analyze_SMFISH_SIM_v1`
    2. Execute script with default parameters. When asked which folder to analyze, specify the folder where the simulated images are stored. You can choose the parental folder, e.g. the one with the mRNA_level, since the script will search recursively in all sub-folders for images to analyze.
    3. Script will perform automated mRNA detection in these images. It uses pre-defined settings (most importantly a detection threshold.
    4. Script will save different analysis results in the folder that has been specified to contain the images to be analyzed. Two folders will be saved
        a. ***_results_detection**:* folder containing all detection results. It uses the same sub-folder structure as the folders containing the actual images. The actual detection results are then stored in two folders *noGMM* (containing a detection with decomposition of potential foci in individual mRNA molecules) and *GMM*, (containing the results of the decomposition). The files are text files containing the detected spots and can be opened in FISH-quant. The GMM folder also contains a sub-folder _plot containing 2D images of the detection results for quick inspection. Individual mRNAs are shown as blue dots, while foci are shown with a red number indicating how many mRNAs have been placed in each foci).
        b. ***_results_localization**:* folder containing for each cell as csv file summarizing the different localization features.
        c. ***localization_features.csv:*** contains the localization features for ALL analyzed cells. This file can then be used to cluster cells based on their localization patterns.

4. **Analyze localization patterns**

    1. To allow for a first inspection of the results, we provide a Matlab GUI: `classif_look_up`
    2. This GUI allows to generate some of the analysis shown in the accompanying paper
        a. Perform k-means clustering and show confusion matrix
        b. Calculate t-SNE plot
        c. Inspect classification results and show features for each class
        d. Show cells belonging to each class

3 Installation

3.1 Requirements

locFISH is distributed together with FISH-quant, which can be obtained together with detailed installation instructions at

https://bitbucket.org/muellerflorian/fish_quant

locFISH was developed and tested

- in **Matlab R2017b** on Mac OS 10.13.4
- on a Mac Pro Mid 2010 (2 x 2.66 GHz 6-core Intel Xeon, 56 GB RAM)

Some function might **NOT** work in earlier versions. Please contact us if you encounter any other problems.

Several function of *the Piotr's Image & Video Toolbox for Matlab*[2] are used.

The following **Matlab toolboxes** are needed:

- Optimization Toolbox
- Statistics Toolbox
- Image Processing Toolbox
- *Parallel Computing Toolbox* (***Optional***, speeds up processing time on computers with multiple CPUs)

4 Simulating smFISH images

We provide a Matlab script `WRAPPER_simulate_smFISH_v1.m` to simulate smFISH images with non-random mRNA distribution. The bitbucket archive already contains a small library of 5 cell shapes, which allows to use the script directly for initial testing. The other parameters defined in the script correspond to the default parameters used in our publication. The entire library with more than 300 cells can be downloaded and installed as described in the chapter ***Data needed to simulate smFISH images***, section ***Download and use full HeLa cell library.*** We further detail in this chapter how these data can be generated from scratch.

4.1 Overview of required data

In order to simulate smFISH images, three different files are needed. These are stored in the sub-folder `data_simulation` of the locFISH installation folder:

1. Library of **cell shapes in 3D**: Matlab file `cell_library_v2.mat`
2. Cropped **background image** for each of the cells in the library. These images are save in a subfolder `/data_simulation/cropped_img`
3. **Image of an individual mRNA molecule:** `PSF.tif`

4.2 Overview of required parameters

In addition to the above-mentioned data, some few parameters have to be specified. More details are provided below and can also be found as comments in the actual Matlab script.

- **How many cells should be simulated** per condition (mRNA level and pattern strength).
- **Intensity distribution of individual mRNA molecules:** described as a skewed Gaussian distribution.
- **mRNA levels.** The user can define an average mRNA level, and a noise term. The average mRNA level is used to determine an mRNA density based on the average cellular volume of all cells in the data-base. For a given cell, the mRNA level is then set such that this cell has the same mRNA density. This level is then further modulated by the specified noise term.
- **Localization patterns**. Each pattern is described by a number of different parameters, in particular to control the pattern strength.

mRNA intensity distribution

The intensity of each placed mRNA is randomly chosen a skewed Gaussian distribution described by 4 parameters (kurtosis, skewness, standard deviation, mean). These parameters can be obtained from experimental data and can be obtained by fitting the experimental distributions with a skewed normal distribution (using the Matlab functions *kurtosis*, *skewness*, *mean* and *std*).

Parameters describing localization patterns

Currently 8 different patterns are implemented. Additional patterns could be added easily due to the open design of the code.

- Random mRNA localization
- Polarized mRNA localization
- Localization in cellular extensions
- Localization towards the cell membrane in 2D
- Localization towards the cell membrane in 3D
- Localization towards the nuclear envelope in 2D
- Localization towards the nuclear envelope in 3D
- mRNA foci

Each pattern – except random - is controlled **with a few parameters**. For a detailed description please consult the supplementary material of the accompanying paper. By changing these parameters, different pattern strength. We considered three cases (i) weak, (ii) moderate, (iii) strong.

How to obtain parameters describing mRNA foci

We extracted the parameters for mRNA foci from experimental DYNC1H1 data, which shows strong foci. We performed mRNA detection combined with the GMM reconstruction described in the Supplementary Material of the accompanying publication. We kept only the spots that were reconstructed with more than 5 mRNAs. From this, we extracted the distribution of the number of foci per cell, and the number of mRNAs in each foci. We fitted these distributions with a skewed normal distribution (using the Matlab functions *kurtosis*, *skewness*, *mean* and *std*). These parameters can then be set in the script. These parameters are used for the pattern level "strong", "moderate" is obtained by reducing them by 25%, "weak" by reducing them by 50%.

4.3 Generating of smFISH images and data storage

After providing the data and the parameters needed for the simulation, you can execute the script. You will be asked where the images should be saved. Within this folder, the script will generate a **folder structure** to save the image: one folder per mRNA level, within this folder a folder for each localization pattern, which further contains sub-folders for each pattern strength. The script generates the **following data:**

- **3D smFISH image**. Name is composed of the name of the background image, plus the name of the pattern, plus a running index.
- 2D **maximum intensity projection for rapid inspection** with name of the 3D image and a prefix *MAX_*.
- **FISH-quant file** containing the 2D outline of the cell and nucleus, the simulated 3D mRNA positions, and a comment specifying the mRNA density and the pattern strength.

For each simulated cell, a cell from the cell-library with the corresponding background image is randomly selected. Then the mRNA level is determined considering the actual cell volume and the specified mRNA density and the noise term. mRNA positions are calculated based on the specified localization patterns. To generate the actual image, each mRNA is simulated as an PSF (Fig 2a-b) and the intensity randomly chosen from the specified intensity distribution (Fig 2c). The mRNAs are placed in the background image (Fig 2d) at their determined location (Fig 2e).

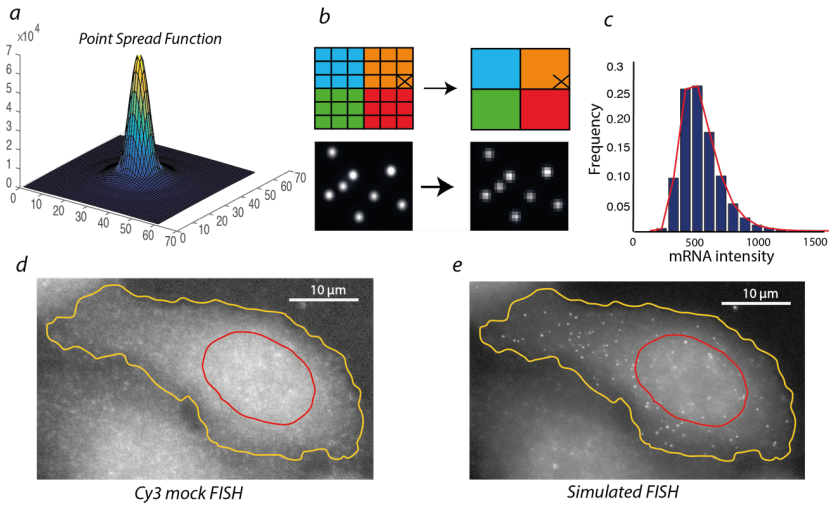


Figure 2. Simulating realistic smFISH images. ***a)*** *3D plot of PSF simulated with the FIJI plugin PSFGenerator.* ***b)*** *Subpixel-placement of mRNA.* ***c)*** *mRNA intensity distribution extracted from experimental data (Kif1c) and fit with a skewed normal distribution (red).* ***d)*** *Image of HeLa cell with background smFISH signal only.* ***e)*** *Simulated FISH image with random mRNA localization.*

Auto-save. The script automatically saves a Matlab file containing information about the current status of the simulation. In case of a Matlab crash, this file can be used to continue the processing after a relaunch of Matlab. For this, you simply have to define the same folder to be used to save the simulations, and Matlab will ask you if you want to use the auto-save file.

5 Data needed to simulate smFISH images

locFISH requires different data to simulate smFISH images as explained in the preceding section. A small test-data set is directly provided with the software package, we provide the full data as a download. In this section we explain

- How this full data-set can be downloaded and used.
- How these data can be generated from scratch with the provided source code.

5.1 Download and use full HeLa cell library

We also provide the full cell library used in the accompanying publication. It can be downloaded from the link listed below. To simulated smISH images, you need the archive `data_simulation.zip`

https://doi.org/10.5281/zenodo.1413488

1. Download and unzip this archive.
2. If you want to keep the smaller test-data (or another cell library), rename the already existing folder `locFISH / data_simulations`.
3. Copy the unzipped folder `data_simulations` in the installation folder of `locFISH`.

5.2 Generating cell library from scratch

In the rest of this section, we describe in detail how these data were obtained, and we provide the necessary Matlab scripts to generate them. In order to follow this workflow, you can download the necessary data from the link below (GAPDH.zip).

https://doi.org/10.5281/zenodo.1413488

Generating the cell library consists of **5 main steps** (Figure 3).

Step 1: Experimental data to obtain realistic 3D cell shapes

We use experimental data to obtain realistic 3D cellular and nuclear shapes. We acquire 4 different channels (for more information see the Material and Methods of the accompanying publication).

1. **DAPI**: 2D nuclear segmentation.
2. **Cytoplasmic marker** (HCS CellMask™ Deep Red, Molecular probes): 2D cellular segmentation.
3. **smFISH against GAPDH** (Cy5): infer the 3D cellular shape.
4. **smFISH against not expressed gene** (Cy3) to obtain realistic background image.

Example images are available in stored in the subfolder `images`: subfolder3D image data (smFISH against GAPHD, background smFISH, DAPI)

Step 2: cell segmentation and mRNA detection

We use the DAPI and CellMask™ to perform **2D segmentation of nuclei and cells** with the open-source software CellCognition[3] using a standard segmentation workflow.

1. Otsu thresholding and watershed separation to segment nuclei in the DAPI channel.
2. Each nucleus then serves as a seed for a watershed segmentation to for cell segmentation in the CellMask™ channel.

The segmentation results are then converted into text files, that can be read with our mRNA detection software FISH-quant[1]. This segmentation workflow is explained in detail in a recent paper[4]. We then **localize individual GAPDH mRNA molecules** with FISH-quant[5] for each segmented cell.

These cell segmentation and mRNA detection results are stored in the subfolder `FQ_results`

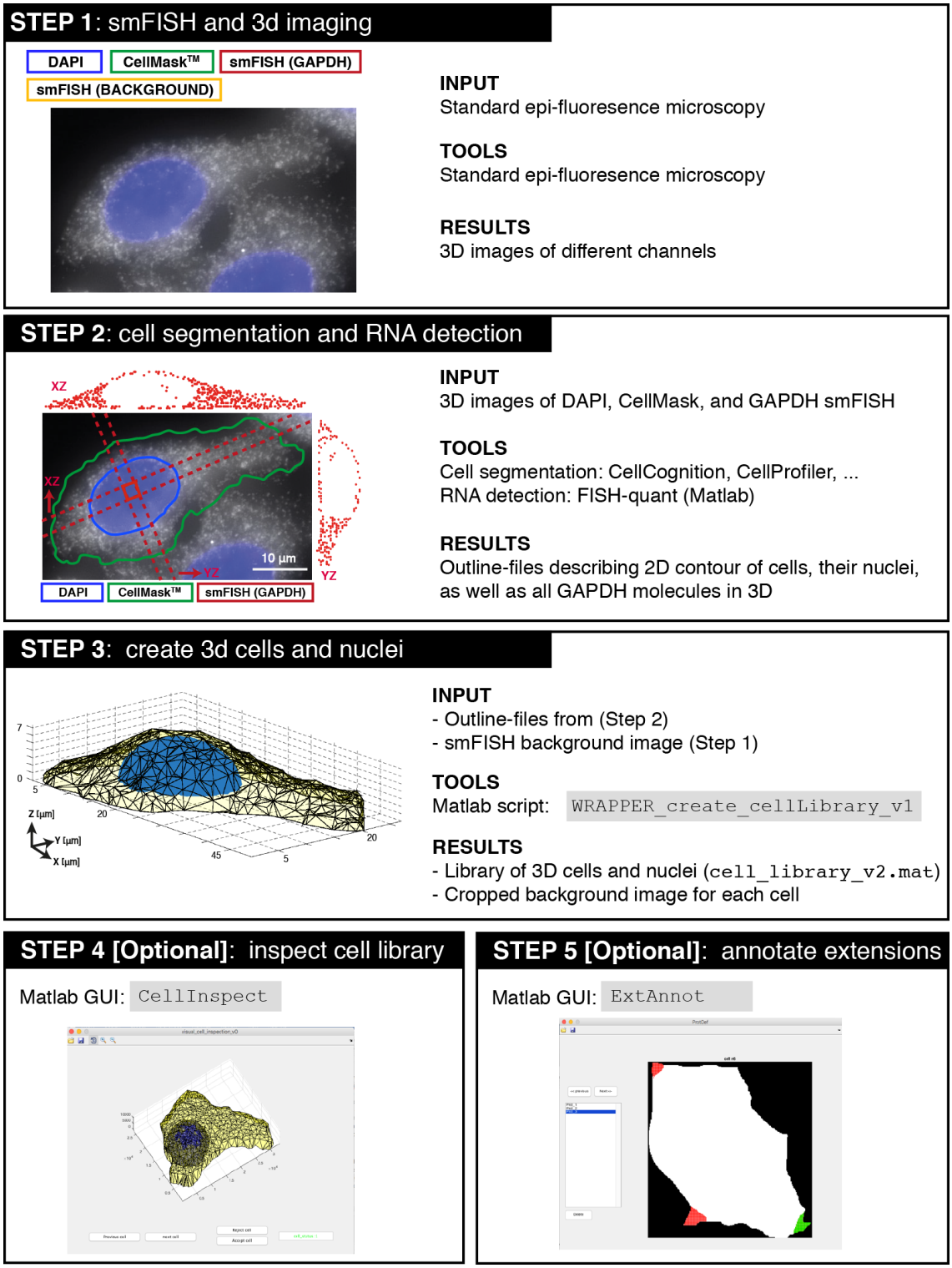


***Figure 3. Constructing a library of 3D cell shapes**. Shows main steps with input and output data, as well as the tools used.*

Step 3: create 3D cells and nuclei

We use the GAPDH mRNA detection results from Step 2 to determine the 3D shape of each cell. This can be performed with the Matlab function

`WRAPPER_create_cellLibrary_v1.m`

When running this function, the user is asked to provide the following information

- **FISH-quant spot detection** results from Step 2.
- **Folder containing background smFISH images and any additional channel** that should be considered (e.g. DAPI). The script automatically crops these channels around each segmented cell.

The script uses the Matlab function `boundary` to create a 3D polygon corresponding to the 3D **cell** around the detected mRNA molecules. The **nuclei** are not directly measured but are simulated as semi-ellipsoids. For this, the 2D nuclear segmentation (from the DAPI channel) is fit with an ellipse, and then the 3D volume is determined by defining the lower and upper border of the nucleus and as a percentage of the maximum total height of the cell. For faster simulations, the 3D polygons are then described as 3D distance matrixes, where each voxel value corresponds to the distance to the cell membrane, or the nuclear envelope.

The script **will save** for each cell a cropped image with the background and other cropped channels (e.g. DAPI). These images will be saved in a sub-folder called `cropped_img` in the folder containing the full-sized images. The actual 3D shapes will be saved in a Matlab structure called *cell_library_v2.mat*, which will be saved in the folder containing the FQ detection results.

Note: When **analyzing new data** some of the parameters in this function might need to be modified (image size, pixel-size, and position of nuclei within the cell).

Step 4: assess quality of 3D cellular shapes

By visual inspection, we exclude 3D shapes that are not accurate, e.g. due to bad segmentation in 2D or a erroneous mRNA detection of GAPDH. To exclude such shapes, we manually inspected each with a simple tool in Matlab. The Matlab command `CellInspect` will open a GUI, where you can load the cell library created as described above. In this GUI, you can then each cell and its nucleus in 3D and decide if the cell should be removed or not from the library with a few steps

1. Type `CellInspect` in the MATLAB command window and press enter.
2. Load the cell library with the load button from the toolbar and select *cell_library_v2*.mat (the file containing the cell library). The first cell in the library will be displayed.
3. The *previous cell* and *next cell* buttons allow to navigate the cell library. Alternatively, you can use the left and right arrow on the keyboard.
4. Buttons in the toolbar allow to rotate the image and zoom.
5. The *Reject cell* and *Accept cell* buttons allow to reject or accept a cell. Alternatively, you can use also keyboard short-cuts: F (reject) and G (accept). By default, all cells are accepted. The current status of a cell is shown in the upper right corner. When pressing, *Accept* and *Reject* buttons the next cell will be shown.
6. After the selection is done, save the new library with the corresponding button in the toolbar. Attention, rejected cells will be removed from the library!

Step 5: annotation of cell extensions

Some mRNAs are enriched in cell extension. We believe that finding a mathematical description of a cell extension is part of the definition of localization features. We therefore opted for a manual annotation of cell extension. We provide a Matlab tool to perform this annotation.

1. Type `ExtAnnot` in the Matlab command window and confirm with enter. This will load the GUI.
2. Load the cell library from the **toolbar on top, click on the "load file"** icon and select the file called *cell_library_v2.mat*. Loading this file can take a moment, so please be patient.
3. After few seconds, you should see in the GUI a binary image corresponding to the automatically segmented outline of the first cell in the library.

Now you can:

- **Navigate from cell to cell** with the *previous cell* and *next cell* buttons (Or **left-arrow** and **right-arrow** key).
- **Add a new extension** by clicking on **Enter**. You can see that the cursor turns into a cross when you are above the image. Now you can draw a region with the left mouse button pressed. The region will be closed as soon as you release the button. It is **important** that the defined region contains the entire

extension. You can also go outside of the image, to define protrusions faster. The program will only consider the part which is overlapping with the outline. If the protrusion touches the border, you can define a region that is partly outside the image of the cell.

- After drawing an extension, it appears in the list shown on the left. All extension are showed in red in the image, the currently selected one in the list appears in green.
- You can **delete an extension** by selecting it in the list and click on *delete* (or use shortcut 'x').
- **Save the result** by clicking on the save icon on the toolbar.

→ When you load saved results, the already defined extensions will be loaded as well. However, the interface will start with the first cell, so you have to move to the cell that you want to highlight.

5.3 Simulation realistic signal for mRNA molecules

In order to obtain a realistic signal for mRNA molecules, we consider both a realistic formation model and intensity distribution. These values can either be used as provided by default, or be newly defined by following the guidelines provided next. These parameters affect how the image will look like, but have no impact on the actual mRNA localization.

Distribution of mRNA intensities

We measure mRNA intensities from experimental smFISH data. For this, we perform mRNA detection and spot fitting with FISH-quant[5]. In the fitting step, each detected mRNA is fit with a 3D Gaussian function, and it's positions, size, background and amplitude estimated. These values are stored for each mRNA in the FISH-quant. The distribution of the fitted amplitudes can then be fit with a with a skewed normal distribution (using the Matlab functions *kurtosis*, *skewness*, *mean* and *std*). These estimates are then specified in the script `WRAPPER_simulate_smFISH_v1.m` and used simulate mRNAs with different intensities.

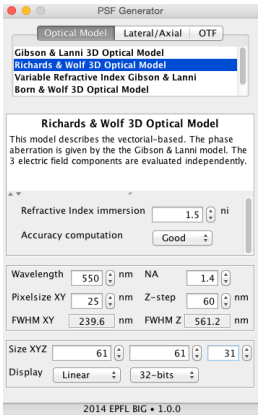
Image of individual mRNAs – the Point Spread Function

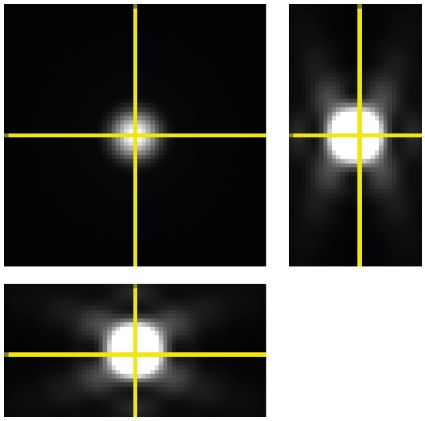
mRNAs are usually diffraction limited in size, and can thus be approximated by the Point-Spread Function (PSF)[5]. We use the Fiji plugin ***PSFGenerator***[6] to simulate such a PSF. More specifically, we used the Richard and Wolf optical model (Fig 4), with the different parameters set to match our microscope (refractive index, NA, wavelength).

In our simulation, we all allow **sub-pixel placements of the individual mRNA molecules**. To achieve this, we first simulate an image with a finer pixel-grid than the final image. We therefore need to simulated the PSF on finer grid. This can be simply done by specifying a n-times smaller pixel values, where n is the binning factor. In our case, the pixel-size in the final image is 107nm, for a binning of 3, we will need a pixel-size of 107nm/3=35.6nm.

Moreover, it has been shown **that the theoretical PSF tends to smaller than the experimentally observed**[5]. We estimate a correction factor by comparing the fitting results of the simulated PSF to the experimental PSF (from mRNAs or beads). We found that the correction factor for our imaging system is 1.3 in xy and 1.6 in z. We then further reducing the size of the pixels of the simulated PSF, which leads to a larger final image. In our case, the final pixel-size in XY is 25nm as specified below.

Figure 4. Simulating a PSF with the PSFGenerator plugin. *(Left) GUI of the plugin with the different parameters. (Right) generated PSF in different views.*





6 Analyzing simulated smFISH images

We provide multiple Matlab scripts to perform the localization of mRNA localization, feature calculation, and finally the classification of cells based on their mRNA localization.

6.1 mRNA detection and calculation of localization features

The Matlab script `WRAPPER_analyze_smFISH__SIM_v1.m` analyzes in batch mode **ALL** images in a parental folder and its subfolders (see Box for summary of analysis steps). When running the script, you only have to specify the parental folder. With the default parameters, simulated with the provided settings can be analyzed.

Analysis results are then saved in new created sub-folders in the parental folder.

- mRNA detection results: *_results_detection*

  The script reproduces the same folder structure as in the parental folder. The results for each image are saved in the same sub-folder as the image, and under the same name with an additional suffix:

  Standard detection without reconstruction of foci: suffix *_res_NO_GMM.txt*

  Detection with reconstruction of foci in individual mRNA molecules: suffix *_res_GMM.txt*

  For the latter, images with the detection results are saved as well. These allow a quick inspection of how well mRNA were detection and foci were described.

- localization features: _results_localization

  The script also saves a table summarizing the localization features of all cells, together with information about the cell names, and how the cell was simulated. This table is saved as a csv file under the file-name *localization_features.csv*. This table can be opened with any spread-sheet application and used to group cells based on RNA localization as explained in the next section.

  The script also reproduces the same folder structure as in the parental folder and saves for each image a separate csv file summarizing the localization features of this cell (suffix *_loc_features.csv)*.

Auto-save. The script automatically saves a Matlab file containing information about which files to process and which file is currently being processed. In case of a Matlab crash, this file can be used to continue the processing after a relaunch of Matlab. For this, you simply have to define the same folder to be processed, and Matlab will ask you if you want to use the auto-save file.

Box: Summary of main analysis steps

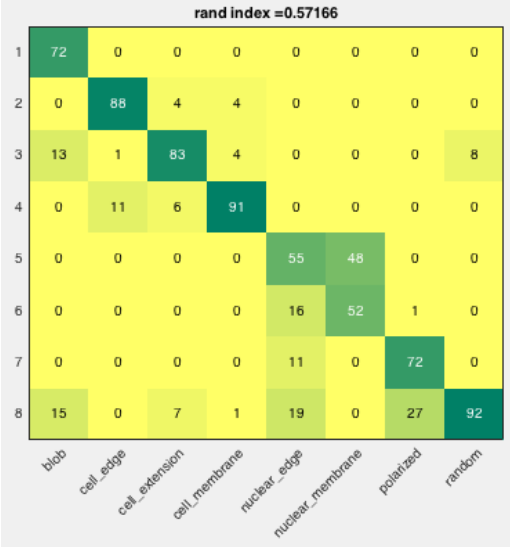
1. The script uses the **mRNA detection workflow of FISH-quant**[5], where images are first filtered, then mRNAs are pre-detected, and finally the detected mRNAs are fit with a 3D Gaussian function.
   - It uses the 2D contours for each cell stored in the cell library to define the outlines for each cell.
   - The detection settings are stored in a text file `_FQ_settings_mature__locFISH.txt`, which is provided with the Matlab code in `locFISH` folder. This settings file is generated with FISH-quant and can be adjusted for new data. The most important parameters concern the mRNA prediction, and more specifically the detection threshold. This threshold has to be adjusted when new simulations with different mRNA intensities are performed.
2. The script uses a module based on a modified **Gaussian Mixture Model** (see accompanying paper for details), to describe foci as a point cloud of individual mRNA positions. The GMM can be controlled with a number of parameters that are set to default values in the script.
3. Each cell is then described with a number of **localization features**.

6.2 Classification and visualization

We provide a Matlab GUI, `classif_look_up` to perform a classification of the cells based on the calculated localization features. It allows to generate s-SNE plots, perform k-means clustering (with the number of

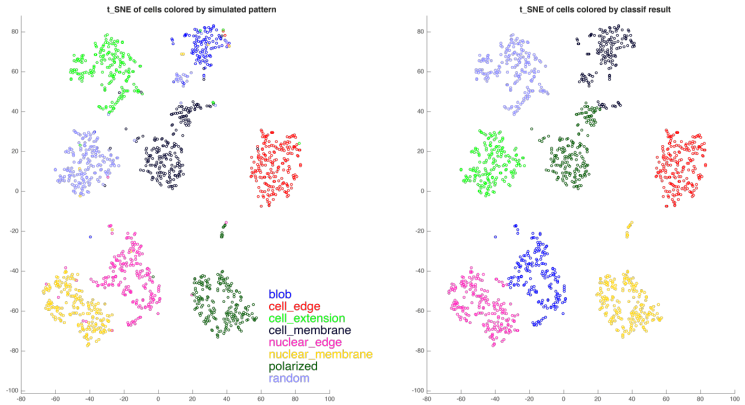
clusters being fixed to the number of present patterns), calculated confusion matrices, and to inspect which cell is in which class.
After startup, you can load the localization features calculated in the previous step (file *localization_features.csv*).

Pressing on "Classification" to perform a k-means clustering. This will then show a **confusion matrix**, where the different classes are shown as rows, and the actual patterns are shown as columns. Ideally, only the diagonal is populated. Off-diagonal elements are mis-misclassification. The overall quality of the clustering is further assessed by the **rand index,** which is displayed as the title of the confusion matrix. The rand index measures the similarity between two clustering results. Here, we compare the actual obtained clustering with the known ground-truth. A value 0 means complete disagreement, while a value of 1 means that the two clustering results are identical. The rand index has a range between 0 and 1, 0 meaning a complete disagreement and 1 two identical clustering.



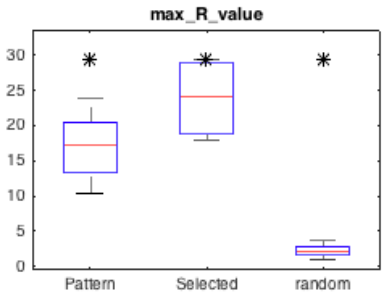
You can then change the **some parameters of this classification** and **which data are considered**.

- You can then select which **pattern levels** and **mRNA levels** will be considered. Here you can either select all, or just a sub-selection.
- You can select **which features** you want to include in the classification (button "Select features"). This opens a small dialog box, where you can select all features with Control-A, and you can remove individual ones with CMD-mouse-click.
- You can **pre-process** the features with t-SNE with a specified number of dimensions. t-SNE is a technique for dimensionality reduction, which is well suited for visualization of high-dimensional data-sets. We also found that pre-processing with t-SNE can also help to improve the classification results. The quality of the results also depends to how many dimensions the data are reduced (default is 3).
- You can **visualize the cells in the t-SNE space**, colored by the simulated pattern (plot below, left panel) and by the classification results (plot below, right panel). For this, check the "Show t-SNE results (with two dimensions) ". Note that here the t-SNE is calculated for 2 dimensions, even if the pre-processing is performed in a higher-dimensional space.



- Once you performed a classification, you **can inspect the properties in each square of the confusion matrix**. For this, press on "Select class for inspection" and move the cursor to the desired square. This will then display a figure, showing all features. For each feature, up to 3 box plots are shown

1. Boxplots for all features of the selected pattern (“Pattern”). This is ground truth and the class on the diagonal.
2. Boxplot for cells in the selected class (“Selected”). This is identical to the first boxplot if you select a class on the diagonal, and different if you select an off-diagonal class. In the latter case, this can allow to identify features where the classification failed.
3. Random control, these are all cells which show random mRNA localization



The star (‘*’) indicates the value of each feature for the currently selected cell. This can again help to identify problems with the features by comparing the class where this cell was assigned to (“Selected”) vs. the class where it should be (“Pattern”).

7 Analysis of experimental data

We provide analysis and visualization scripts for experimental data. We provide one script to analyze data organized in a simple way ("Simple analysis") to get started fast, and second script that we found useful to analyze larger data-sets. For either script, the describe data organization has to be implemented according to the instructions given.

IMPORTANT: the scripts support only images saved as **tif files as input**. Each tif file contains the z-stack for one acquired position and one channel (e.g. the smFISH images).

7.1 Simple analysis

This script uses a simple data organization, where all files needed for the analysis are saved in one folder. These files are

- **Image data**. e.g. smFISH_ch1_p01.tif

  *The script requires that **smFISH images are saved separately for each position, and each channel is stored as a separate z-stack**.*

- **FISH-quant outline files**, e.g. smFISH_ch1_outlines.txt

  *The outline files can be generated with FISH-quant, and **have to contain an outline of the cells and their nuclei.** They have to be saved with the propose default name, which is the name of the smFISH image with the suffix* `_outlines.txt`.

- **FISH-quant settings file,** e.g. smFISH_ch1__settings_mature.txt

  This file is generated by FISH-quant and contain all necessary settings of the analysis. Best practice is to save it under the proposed filename, which is the name of the smFISH image with the suffix `_settings_mature.txt`. *If you save it under a different name, make sure that the file contains the string **_settings**.*

Importantly,

- a folder should **contain ONLY one settings file**.
- a folder **can contain multiple images with the associated outline files**. Each of the outline files will be processed.

The actual analysis is performed with the script
`WRAPPER_analyze_smFISH__EXP_v1.m`

When running this script, the user gets asked which folder should be processed and where the table summarizing the localization features of all analyzed cells should be saved. The script will then search this folder for FISH-quant settings file and process all outline files. See section *"7.3 Analysis and inspection of results"* for more details on how data are saved and can be inspected.

You can also recursively search the specified folder. The script will then process each folder containing a settings files and outline files independently. This can be useful when analyzing smFISH data from different experiments. To enable this simple batch processing, change in the scrip the flag `param.flag_recursive` to 1.

7.2 Analyzing large-scale data-sets

Data organization

Folder structure – general comments

We opted for a data organization that separates the raw image data (folder `Acquisition`) from the analysis results (folder `Analysis`).

Folder structure – acquisition

For the ***Acquisition*** folder, we define a folder for each acquisition date. In this folder, the acquired images of each gene are stored in a separate folder. The name of this folder will be used as an identifier in the down-stream analysis. Each channel is stored as a separate z-stack. Below an example organization is shown. The different levels in the folder hierarchy are indicated by different columns.

Acquisition			
	170601_Nikon100X		
		HeLa_RAB13	Contains all images *NIH3T3_P53_p1_CY3.tif* *NIH3T3_ P53_p1_DAPI.tif* *NIH3T3_ P53_p2_CY3.tif* *NIH3T3_ P53_p2_CY3.tif* …
		HeLa_DYNC1H1	Contains images …
	170610_Nikon100X		
		HeLa_BUB1	Contains images …
		HeLa_P300	Contains images for slide …

Folder structure – analysis results

For the Analysis folder, we maintain the basic organization from above. For each gene, different folders containing different parts of the analysis, e.g. folders containing outline files or the **actual mRNA detection results**.

Analysis			
	170601_Nikon100X		
		HeLa_RAB13	Contains folder with results FQ_outlines FQ_results_180501 Cell_crop_MIP ZPROJ …
		HeLa_ DYNC1H1	Contains folder with results …

Analysis workflow

All analysis results are stored in the `Analysis` folder. The main steps are

1. **Cell segmentation.** We perform cell segmentation on either the non-specific smFISH background or dedicated cytoplasmic marker such as CellMask™. More details can be found in the corresponding publication[4,5].
    a. We create 2D images for segmentation with a focus-based projection approach, which are stored in the folder `ZPROJ`. We found that these images yield better segmentation results than a standard maximum-intensity projection
    b. Images are segmented with standard tools such as CellCognition[3] or CellProfiler[7].
    c. The resulting segmentation mask are then converted to FISH-quant outline files, which are stored in the folder `FQ_outlines`.

2. **Settings for mRNA detection.** We perform this analysis based on tools provided in FISH-quant[5].

a. For each smFISH experiment, a **separate detection threshold** has to be set. We implemented a simple tool that allows determining this threshold fast: `FQ_detect` . More details can be found in the FISH-quant documentation.

   In short, with this tool you can create a FISH-quant settings file with an adjusted detection threshold for each experiment (`_FQ_settings_MATURE.txt`), which is saved in a newly created folder `FQ_results_YYMMDD`, where YYMMDD is the date.

b. We provide a script for **batch processing**: `WRAPPER_analyze_smFISH__EXP_batch_v1.m`

   When running the script, you will be asked to

   I. Define the folder containing the FQ settings files that should be processed. It performs a recursive search, so a parental folder can be specified.

   II. Define which settings files should be processed. (based on a matching string). This allows to reprocess data with an updated settings file.

   III. Define the folder where the table summarizing all cells with their localization features will be saved

   The script will then analyze the data as described in the next section.

7.3 Analysis and inspection of results

The scripts perform a similar workflow as the batch processing script described in the previous section for simulated data, (1) mRNA detection, (2) application of GMM, (3) saving a MIP of the cropped cell, (4) calculating the localization features. Results are saved as described in the section *"6.1 mRNA detection and calculation of localization features"* above.

RNA detection results

The scripts will create a number of folders for each analyzed data-set.

- `results_GMM` contains the RNA detection with the GMM (to decompose foci into individual RNAs). It further contains a subfolder #plots with a PDF illustrating the results of the GMM detection. Individual RNAs are shown in green, decomposed foci with a red number indicating how many RNAs were estimated to be in each foci.
- `results_noGMM` contains the RNA detection results without the GMM.
- `localization_features` contains a CSV files with the calculated localization features.

The scripts also generate a **summary file with all the localization features** for each of the analyzed data-set. This table can then be further inspected to perform classification based on similarity in RNA localization as explained next. These tables can also be **opened and modified in a spreadsheet application** such as Excel. This allows to fuse results from different analysis, e.g. if new data have been acquired.

Analysis of mRNA localization.

We provide a simple user-interface (`exp_data_look_up.m`) that allows for a first inspection of the mRNA localization data. In this interface, you can choose which data to analyze (based on the identifier extracted from the file-name), and you can select which features to consider. You can then perform a t-SNE visualization, and k-means clustering with a user-defined number of classes.

Zoomable t-SNE plot

Inspecting larger data-sets can be a difficult task. To facilitate this, we provide the possibility to create high-resolution images of the t-SNE visualization, where the data-points are represented by the detection results. Like this, it can be easier to visually judge the biological relevance of given classification. For more details for how to generate these kind of images, please consult the separate document **FISH_loc_deepzoom.pdf**

8 References

1. Mueller, F. *et al.* FISH-quant: automatic counting of transcripts in 3D FISH images. *Nat. Methods* **10,** 277–278 (2013).
2. Piotr's Matlab Toolbox. Available at: http://vision.ucsd.edu/~pdollar/toolbox/doc/. (Accessed: 6th March 2011)
3. Held, M. *et al.* CellCognition: time-resolved phenotype annotation in high-throughput live cell imaging. *Nat. Methods* **7,** 747–754 (2010).
4. Tsanov, N. *et al.* smiFISH and FISH-quant - a flexible single RNA detection approach with super-resolution capability. *Nucleic Acids Res.* **44,** e165 (2016).
5. Mueller, F. *et al.* FISH-quant: automatic counting of transcripts in 3D FISH images. *Nat. Methods* **10,** 277–278 (2013).
6. Kirshner, H., Aguet, F., Sage, D. & Unser, M. 3-D PSF fitting for fluorescence microscopy: implementation and localization application. *J. Microsc.* **249,** 13–25 (2013).
7. Kamentsky, L. *et al.* Improved structure, function and compatibility for CellProfiler: modular high-throughput image analysis software. *Bioinforma. Oxf. Engl.* **27,** 1179–1180 (2011).