

FISH-quant v3

Generate FQ outlines from automated cell segmentation

MUELLER Florian¹ muellerf.research@gmail.com

SAMACOITS Aubin¹ aubin.samacoits@gmail.com

WALTER Thomas² thomas.walter@curie.fr

¹Institut Pasteur Paris, *Computational Imaging & Modeling Unit, France*

² Institut Curie, Paris, France

1. Introduction	3
Installation instruction for the Matlab tool <i>FQ_seg</i> Matlab tool:	3
Workflow	3
Image-acquisition	4
2. Z-projections	5
3. Workflow for CellProfiler	7
Segmentation	7
Analyze Images	8
Test mode	9
Generate FQ outlines files from CellProfiler results	9
Specify experimental parameters	9
Define naming scheme of original images and segmentation results	9
[Optional] Generating outline files for a second color	10
Specify folder where the results will be saved	10
Create outlines	10
4. Workflow for CellCognition	11
Segmentation	11
Naming convention of CellCognition: position, well, subwell	11
Importing of images	11
Cell segmentation	13
Saving the results	14
Analysis and organization of results files	14
Generate FQ outlines files from CellCognition results	15
Specify experimental parameters	15
Define channel and segmentation method identifier	15
Specify folder where the results will be saved	16
Define folder	16
Create outlines	16
5. Inspect outline files	17

1. Introduction

Cell segmentation can be performed with many specialized software packages.

- *CellProfiler*² described as a “*free open-source software designed to enable biologists without training in computer vision or programming to quantitatively measure phenotypes from thousands of images automatically*”.
- *CellCognition*¹ described as a “*computational framework dedicated to the automatic analysis of live cell imaging data in the context of High-Content Screening (HCS)*”.

Either package allows the automated detection (segmentation) of cells and nuclei. This can then be used to automatically generate outline files that can be analyzed in FISH-quant.

FISH-quant is a freely available Matlab package to automatically analyze smFISH images in 3D,

CellProfiler is available at <http://www.cellprofiler.org/>

CellCognition is available at <http://www.cellcognition.org>

FISH-quant is available at <https://code.google.com/p/fish-quant/>

Installation instruction for the Matlab tool *FQ_seg* Matlab tool:

- **Copy the Matlab code** from dropbox on your computer. Do not let it in the “Download” folder, but copy it dedicated folder for Matlab source code.
- **Set the Matlab path.** The Matlab path is a set of folder that Matlab uses to look for each of functions you run. If you try to run a function which is not present in one of the folder on the Matlab path, you will have a “function not found” error message. You can change the path Matlab in the menu, selecting *File > Set Path*. You need now to add the folder which contains the code to the path with the button ‘Add with subfolders’. Before closing this interface, click on ‘save’ to save the updated path settings.
- You can now type *FQ_seg* in the command window of Matlab to launch the user-interface.

Workflow

Workflow requires CellProfiler, FISH-quant (Matlab), and a special Matlab user-interface (*FQ_seg*) that allows controlling many of the steps described below.

1. Image acquisition in 3D.
2. Generate z-projections to obtain 2D images
3. Segmentation of 2D images in *CellProfiler* or *CellCognition*
4. Conversion of segmentation results in FQ outline files.
5. Check quality of segmentation results

Image-acquisition

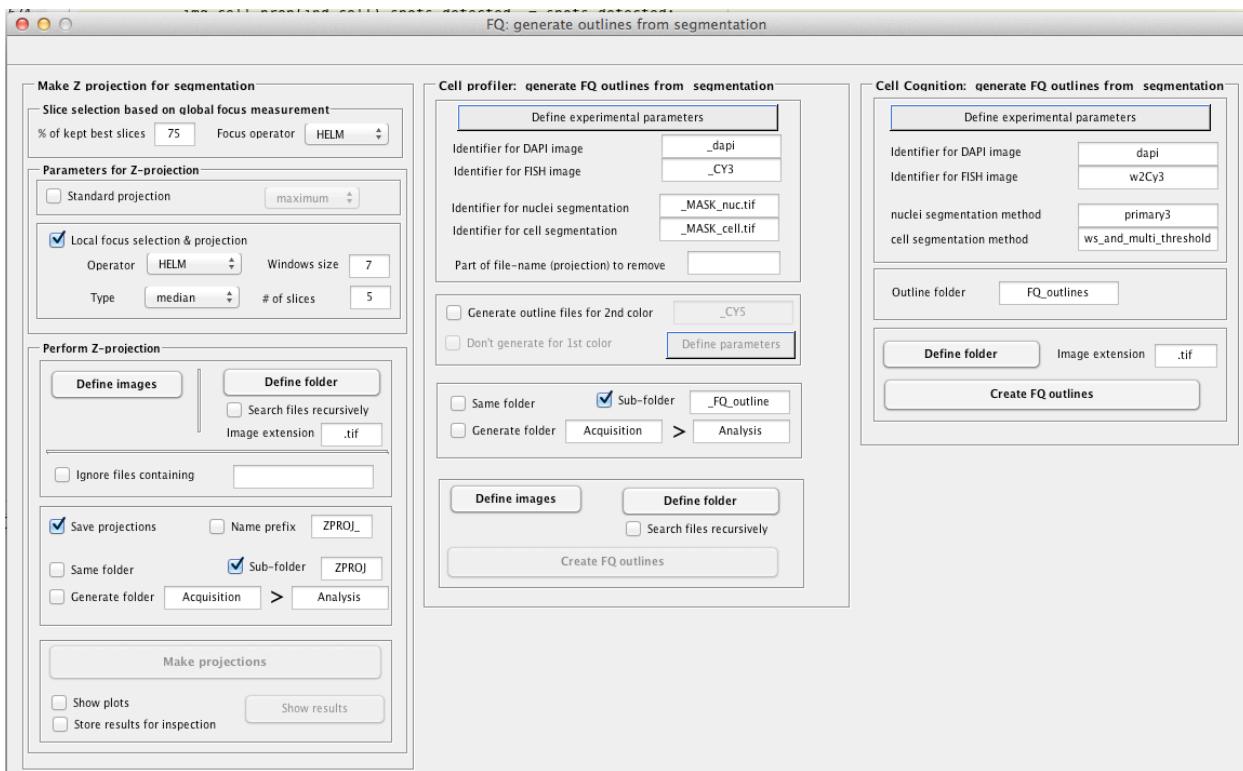
- Images have to be 3D and saved as a separate stacked TIF for each channel.
- Images have to follow a strict naming convention and start with a unique name, e.g. *img_001* where **001** is a running index. Subsequent identifiers specify the image type, e.g. **_DAPI** for DAPI images and **_CY3** for FISH. The complete file-name for different FISH image would be *img_001_CY3.tif*, *img_002_CY3.tif*, ... and similar for DAPI *img_001_DAPI.tif*, *img_002_DAPI.tif*,

2. Z-projections

CellProfiler and *CellCognition* work only on 2D images. Frequently, for the segmentation a **maximum intensity projection** of each channel is used. However, we found that better segmentation results can be obtained by performing a more advanced projection method. Specifically, we use focus measurements to obtain ‘sharper’ images to increase the segmentation accuracy. This is achieved by a 2-step process:

1. Calculation of the total focus of each z-slice. With this measurement, blurry slices, which have mainly out-of-focus signal, can be removed.
2. Local focus projection. Here the focus is measured locally for a user defined window size. Then the z-projection is performed for each pixel over the best-identified slices.

These projections can be obtained with a Matlab tool which can be opened by typing **FQ_seg** in the command window. This will open the user interface shown below. The projection is performed with the panels on the left side with the title ‘**Make Z projections for segmentation**’.



Slice selection

In the first panel you can select the focus projection method (See³ for details, default ‘HELM’). You can then select which percentage of slices will be kept (default 75%). To use all slices, defined a value of 100%.

Z-projection

You can either perform either a **standard global projection** based on a mean, median, or maximum projection or a projection based on local focus measurements.

For the local focus, you can again select the focus metric. In addition, you can define the window size that will be used to calculate the local focus. You can specify how many slices with the respective best focus values will be projected. Finally, the same projection methods as for the global approach are available if more than one slice will be used.

Define images / Define folder

In this panel, you can specify the images that will be analyzed. Several features allow a general selection of images

- You can define manually images in a given folder with the button '*Define images*'.
- Alternatively, you can specify a folder in which all images will be analyzed. In this case, you have to define additionally the image extension, e.g. *.tif*. Note, that the image extension is case-sensitive.
- You can also define folders recursively, i.e. images will be searched in the defined folder and all subfolders.
- You can also specify that certain files will not be processed. Here you have to specify parts of the file-name that uniquely specifies them, e.g. 'CY3'.

Save images

The generated images can then be saved. By default, they are saved under the same file-name as the original image in a **subfolder** of location of the original image called 'ZPROJ'.

There are different options to change this behavior.

- You can add a pre-fix to the file-name. Important: if you add a prefix, you have to specify during step where the FQ outline files are created, that this prefix has to be removed. Otherwise, the outline files will point to an image that doesn't exist (more details below).
- You can save the files in the same folder as the original files.
- You can save the files in a different folder. This option allows you to replace parts of the folder-name by another string. For instance, files are saved in */Users/FM/img/raw*. You can now say that 'raw' should be replaced by 'ZPROJ'. The images will then be saved in the folder */Users/FM/img/ZPROJ*

Note 1: If a file with the same name is present, the image will not be saved. You have to manually delete this image first.

Note 2: it is possible to combine saving in a subfolder and replace parts of a name of the folder.

Additionally to the images, the program will also save a small text file with the settings used to create the z-projections.

Show plots

If you enable this option, you will obtain a plot for each image that summarizes the results for each analyzed image. We recommend doing this only for a small number of images, to avoid having too many open images.

Show results

With this option enable, the program will save the standard maximum intensity projection and the advanced projection for each image. When the processing is done, you can then inspect these in a separate user interface by opening 'Show results'. Enable this option when processing only for a few images, otherwise the resulting data might be so large that Matlab crashes.

Perform z-projection

Pressing the button 'Make Projections' will then apply the Z-projection to all specified images. Images with fewer than 3 Z slices will not be considered in the analysis.

3. Workflow for CellProfiler

Segmentation

Below we list a typical workflow in CellProfiler for the segmentation of nuclei and cells. The results are saved in masks that can be used to generate outline files for FISH-quant. Such workflows are saved in so-called pipelines, which are provided with this toolbox (files ending with `.cppipe`).

1. Open CellProfiler
2. Import analysis pipeline from Menu *file > Import Pipeline > from file*
3. Then you have to **specify the images** that you would like to analysis. These are the 2D projection obtained above. You can drag and drop them into the big window in the '**Images**' module (part of the Input modules).
4. The next step is very important. Here the **channels** belonging to the same image (e.g. DAPI for the nuclei and smFISH for the cellular outlines) are assigned to each other. This done with two other modules. First, the '**Metadata**' module where the channel and position information is extracted. Second, the '**NamesAndTypes**' module where images obtain names for the further analysis steps.
 - The extraction method for the meta-data depends on how your images are named and organized. We usually use a method that extracts the information from file-names. To guarantee largest flexibility so-called regular expression are used. This is a relatively simple syntax to automatically extract different parts of a text. For more details, you can consult any website about regular expression such as <http://www.regular-expressions.info/quickstart.html>

For the example data, the image name follows the convention `gene_pos_channel.tif`, where *gene* is the analyzed gene, *pos* the position of the image (i.e. a value that will be increased for each acquired multi-color stack), and *channel* the color channel (e.g. Cy3 or Dapi). To extract the relevant information, we use the following regular expression

`^(?P<FileID>.*_*)_?(?P<Channel>.*).tif$`

This might look a little bit confusing at the beginning. A good website to test your regular expression is <https://regex101.com/>. It not only allows you to test your expressions, but provides information about the different terms. When we test this expression on an example file-name (*BGH_01_CY3.tif*), we see the following

The screenshot shows the regex101.com interface. In the 'REGULAR EXPRESSION' field, the pattern `^(?P<FileID>.*_*)_?(?P<Channel>.*).tif$` is entered. The 'TEST STRING' field contains `BGH_01_CY3.tif`. The 'EXPLANATION' panel provides a detailed breakdown of the regex components:

- `^(?P<FileID>.*_*)_?(?P<Channel>.*).tif$`
- `^ assert position at the start of the string`
- `(?P<FileID>.*_*)` Named capturing group `FileID`
 - `* matches any character (except newline)`
 - `Quantifier: * Between zero and unlimited times, as many times as possible (greedy)`
 - `_ matches the character – literally`
 - `(?P<Channel>.*)` Named capturing group `Channel`
 - `* matches any character (except newline)`
 - `Quantifier: * Between zero and unlimited times, as many times as possible (greedy)`
 - `.tif matches any character (except newline)`
 - `TIFF matches the characters TIF literally (case sensitive)`
- `$ assert position at end of the string`

In the 'MATCH INFORMATION' panel, the results are shown:

MATCH 1
<code>FileID [0-6] 'BGH_01'</code>
<code>Channel [7-10] 'CY3'</code>

With this regular expression, we simple extract the first part of the string and a name it *FileID* (e.g. BGH_01) and the second par of the string and name it *Channel* (e.g. CY3). For the corresponding DAPI image BGH_01_DAPI.tif, we obtain the same FileID but a different channel (DAPI). So with these two extracted strings, we have unique identifiers to properly assign for each image the different channels.

If your image names don't follow this convention, you have to adapt the regular expression. See *CellProfiler* for more help.

To apply the settings to extract the Metadata press the button '**Update metadata**'. This might then take a little while. If everything works fine, a table in the lower part of the panel will populate. Containing information about the channel and position. If that's the case you can continue

- To now assign the different channels to a position, go to the '*NamesAndType*' module. Here you tell CellProfiler how to refer to the different images (e.g. DAPI vs CY3). This will be important for the subsequent analysis steps, where for instance the nuclear segmentation is done based on the DAPI image. In the example data, we simply use the channel information in the file-name to decide if a image is DAPI or CY. Simply click update. This will populate a table in the lower part, where the different images of the different channels are correctly assigned to each other and names are attributed. For other images, you will likely have to update these naming rules. Please consult the CellProfiler help if this doesn't work.
5. You can then either simply apply the segmentation settings to all images (Button 'Analyze Images') or start a test mode (Button 'Start test mode').
 6. We recommend saving the CellProfiler project in the folder with the Z-projections. This allows later inspecting the settings used for the segmentation.

Analyze Images

This will perform the automated segmentation of all images and save the results automatically.

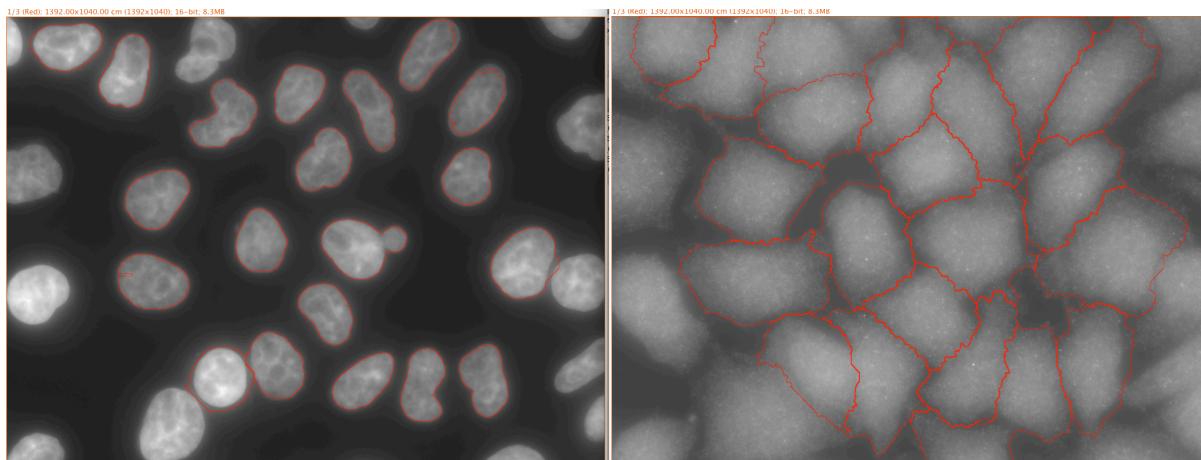
FOR FASTER PROCESSING DISABLE THE DISPLAY OF THE SEGMENTATION.

This can be done by clicking on the icon with the eye next to the different processing steps.

The **segmentation results** are saved in the folder of the MIP

- a. Masks for cells and nuclei. These are images with the same name as the MIP, followed by '_MASK_cell' or '_MASK_nuc'. In these images each identified cell or nucleus has a different gray-scale value. They will then serve as an input to automatically generate FQ outline files.
- b. Images that allow to check how well the segmentation worked, they also have the same file-name as the MIP and end by '_CONTOUR_cell' or '_CONTOUR_nuc'

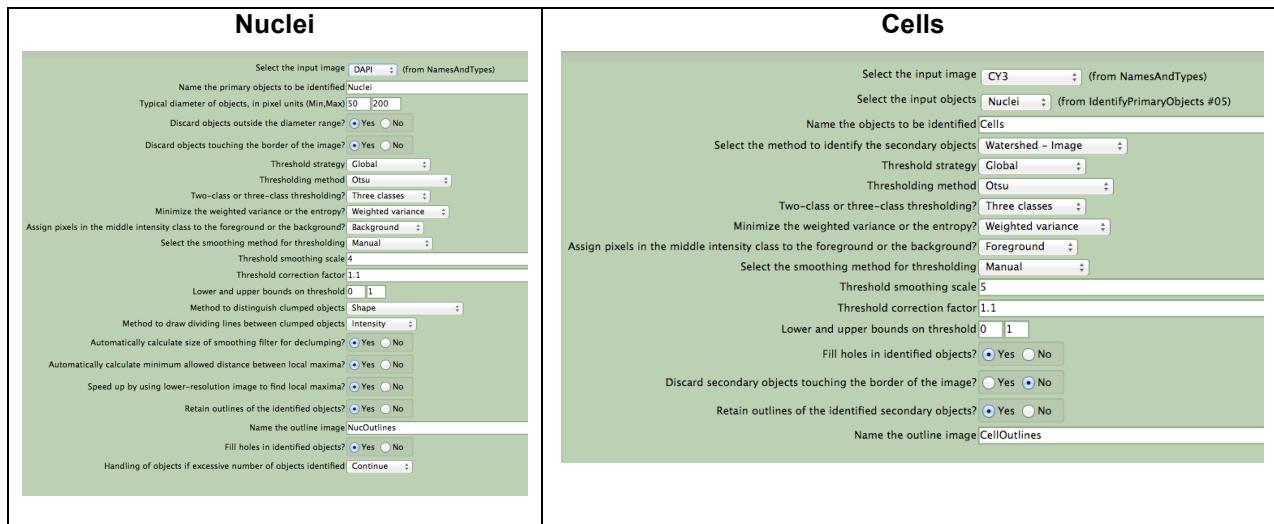
Below are an example for segmentation nuclei (left) and cells (right).



Test mode

This allows you to change settings and look at their impact at the individual steps of the pipeline. More information can be found in the *CellProfiler* help.

Below typical settings used for nuclear and cellular segmentation are shown. CellProfiler provides an excellent help function that explains the different parameters.



Generate FQ outlines files from CellProfiler results

The segmentation masks generated by *CellProfiler* can be used to automatically create FQ outline files with the middle part of the user interface.

Specify experimental parameters

First, you have to define the '**Experimental parameters**'. These are the same parameters that have to be defined for the images in the main FISH-quant interface when pressing the 'Modify' button in the panel 'Experimental parameters'. Simply adjust them to your data by changing the corresponding values.

Note 1: You have to **ALWAYS** define the parameters – even if the default parameters are good. Only then the button to generate the outlines will be enabled.

Note 2: you can change the default parameters in a text-file called 'FISH-QUANT_default_par.txt' which is located in the same directory as the *FQ_seg* Matlab file.

Note 3: If you added a prefix to the file-name of the 2D projections, then you have to specify in the dedicated box that you want to remove this prefix when generating the outline files. This is important, since the outline files point to the original image files (whose file-name does not contain this prefix).

Define naming scheme of original images and segmentation results

Second, you have to define a few parameters regarding the **naming convention of your files**. You have to define the unique identifier for the FISH and DAPI images. These identifiers have to be defined in a way that when you take the full file-name of the FISH image and you replace the identifier for FISH, e.g. CY3, by the identifier of the DAPI, e.g. 'DAPI', you get the DAPI file-name. Then, you have to define the

identifier of your segmentation results using the suffix that *CellProfiler* added in the name of your masks (MASK_nuc and MASK_cell for example).

You also have to specify the **file-extension of the original (3D) images**. By default the extension *.tif* is used, if your files have another extension (*.tiff*, *.stk*, ...), you have to accordingly change the text. This is important to guarantee that these files are correctly referenced in the FISH-quant outline files.

[Optional] Generating outline files for a second color

This option allows to generate outline files for a second color, e.g. for a dual-color FISH experiment. The outlines for this color will be based on the segmentation results of the first color and the exact same cells will be used. This allows a simple comparison between the detection results.

As above, the identifier for the second color has to be specified, e.g. 'CY5'. You also have to redefine the experimental parameters (most often to adjust the excitation and emission wavelength).

You can also choose that only the outlines of the second color are saved. This option can be useful if the first color doesn't contain actual smFISH data but results of a dedicated cell segmentation stain.

Specify folder where the results will be saved

The same options as for the projections are available. More details can be found there.

Create outlines

Lastly, press the button '**Create FQ outlines**'. The script will then automatically search for the files describing the segmentation of cells and nuclei. For each image an outline file with the reference to the ORIGINAL 3D image will be generated and nuclei assigned to their respective cells.

4. Workflow for CellCognition

Segmentation

CellCognition recursively scans user provided folders for images, performs the segmentation, and saves the segmentation results as masks. Such workflows are saved as **settings files**, which are provided with this toolbox (files ending with `.conf`).

You can load the settings file from within *CellCognition* using *File>Open Settings*. Once the settings file is loaded, you just have to set the input and output folder and the segmentation is ready to start.

Two different settings file are provided¹:

- “Global method settings” which combines a global threshold with a watershed segmentation.
- “3-Otsu method settings” where a Otsu thresholding with 3 classes is performed. The middle class can be assigned to the background or the foreground. We assign the middle class to the foreground, i.e. the cells. The thresholded image is again analyzed with a watershed segmentation.

→ In our experience, the “3-Otsu method settings” gives best results for segmentation of smFISH images.

Naming convention of CellCognition: position, well, subwell

CellCognition was developed specifically for high-content screen. This affects the data organization and a few terms are important to understand and how they translate to a simple imaging experiment. In high-content screens, images are often acquired on multi-well plates. This means a given plate, each well corresponds to a specific experiment (a specific treatment, a specific gene, etc). For each well, images at different position can be acquired. These positions are referred to as sub-wells. The position of an image is a unique value for each plate and is defined by combination of the well and the subwell. In a given plate, the position will be for example « 001_04 », where 001 is the well and 04 is the subwell.

Important: Plates are separate folders (as explained below in more detail). Well and subwell is defined by the file-name (explained below). However, the well data can be (but most not be) in separate folders.

Example. smFISH experiments are often not performed in a multi-well format but this notion can be adapted and allows for a flexible organization. We perform FISH experiments against 5 different genes on two different days. We can consider each day as a “plate”. And each of the FISH experiments as a “well”. We have therefore two folders corresponding to the different dates. Each can contain separate folders for each gene with the different acquired positions (corresponding to sub-wells). What is important is how the smFISH images are named – they have to contain a part that can be interpreted as a well (the same for a given gene) and a part that is the sub-well (the actual different acquired positions).

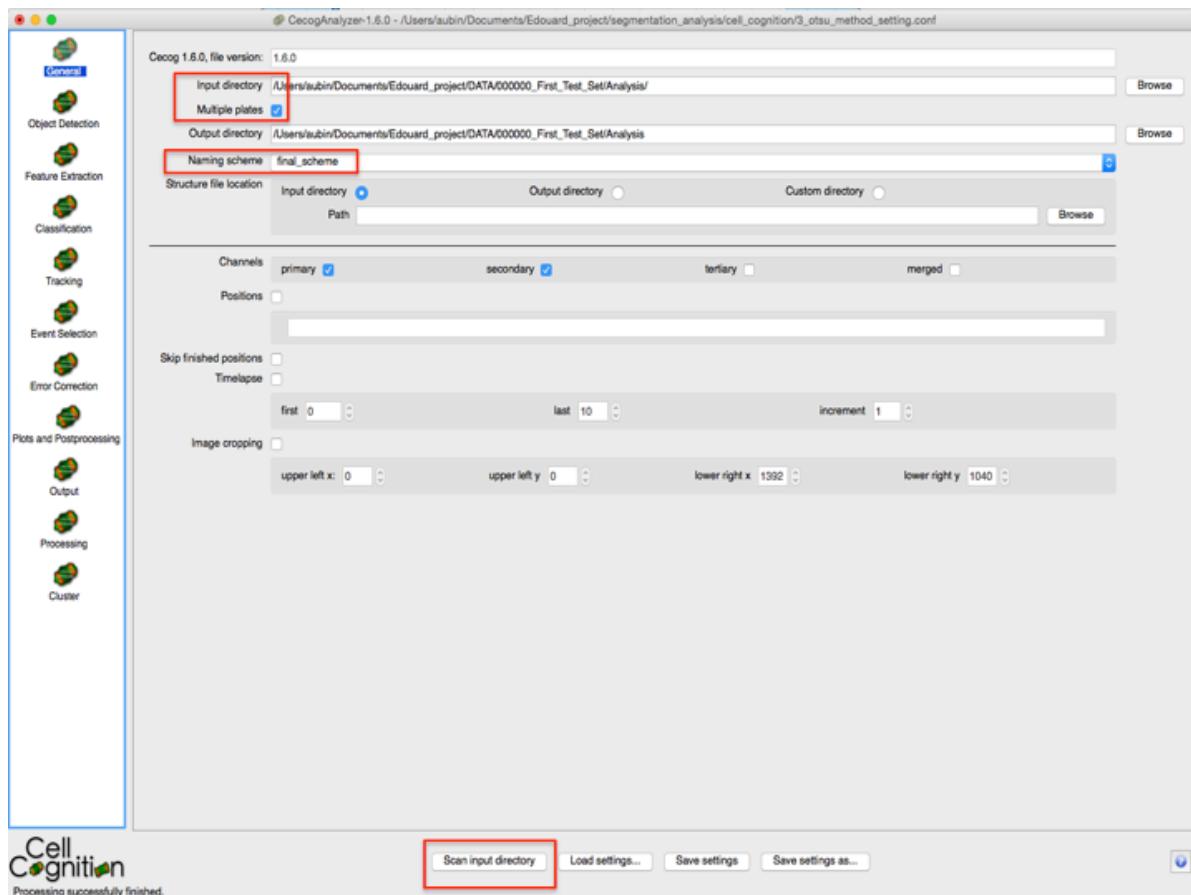
Importing of images

You have to specify an input folder, which is scanned by CellCognition at the beginning of the analysis to detect all images. It performs a recursive search of the folder, looking into every subfolder of the input

¹ These methods are available on the GitHub version of CellCognition, and will be part of the official release 1.6.0 which is due in Fall 2015.

folder.

An important option is ‘multiple plates’ (see below). When enabled this indicates that the input folder contains subfolders corresponding to different plate. If the option is enabled, the organization of the results will be in a per-plate basis (more details below).



The import of images is based on a **regular expression** that allows extracting basic information about the images from their file names (well, time, position etc....). These regular expressions are stored in the “naming_schemas.ini” file, which can be found with *Cell/Cognition* code package (On a Mac, right-click on the *Cell/Cognition* icon and chose “Show package contents”) at “content\Resources\resource\naming_schemas.ini”. We provide an updated file, with a naming scheme called *FISH_HCS* adapted to the file-names with the naming convention:

well_cell_gene_pos_channel.tif.

For instance, *Cell/Cognition* will interpret w3_HeLa_DYNC1H1_P4_DAPI.tif as

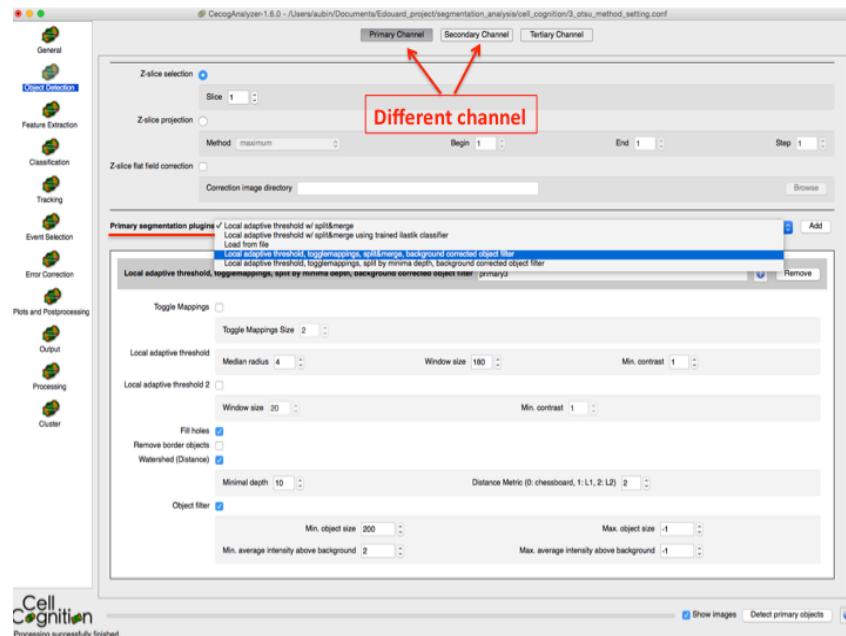
Plate :	Determined by the folder with the images
Well :	3
Subwell :	4
Channel :	DAPI

→ Position (as defined by CellCognition): 3_4

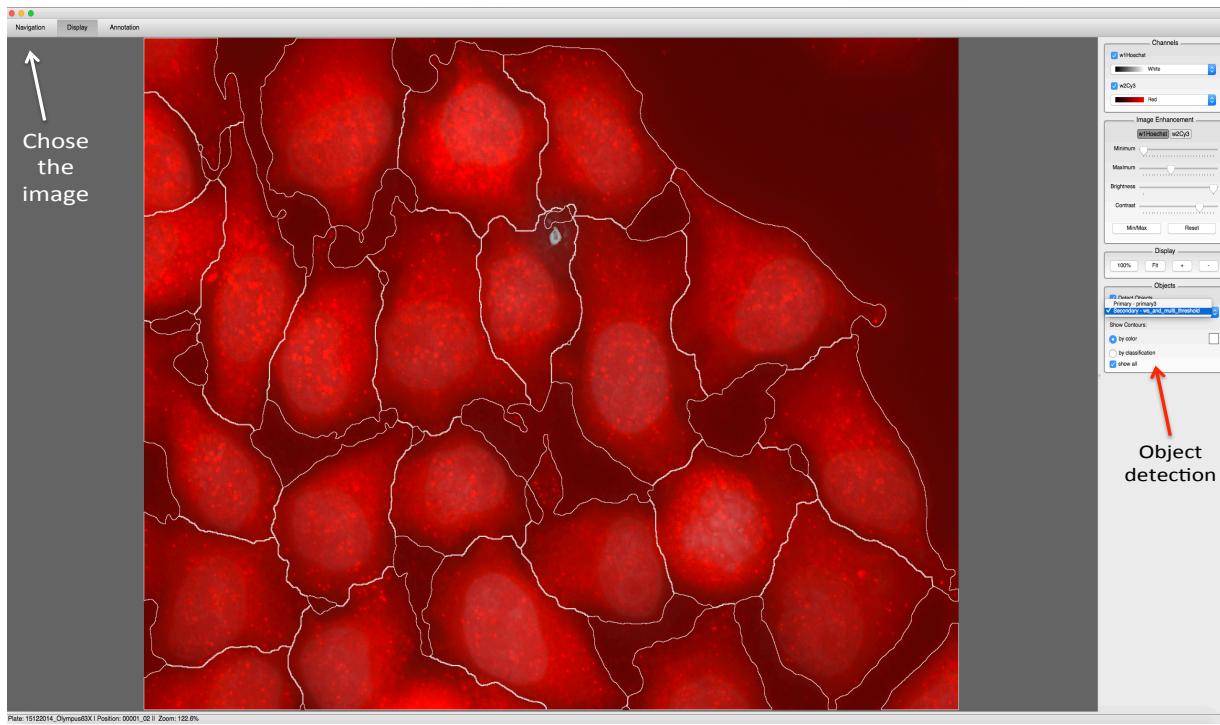
Once the input folder is defined and the correct naming schema is selected, you can click on “**scan input directory**” and *CellCognition* displays how many plates/images have been found.

Cell segmentation

The segmentation of cells is a two steps process. First, nuclei will be segmented with a global thresholding strategy. Second, each nuclei will serve as a ‘seed’ for the segmentation of the cells. These two detection steps have to be defined by the user. Once in the “object detection” tab, it is possible to navigate between the different channels, and, for each one, to select a primary/secondary plugin in the list box and add it. The set of parameters are now visible and can be adjusted.



The easiest way to adjust the parameter is to use the browser (**CTRL + B**). It is possible with the browser to select an image in the list, to adjust the contrast of the several channel, and to perform and visualized the segmentation with the settings specified in the main interface of *CellCognition*.



Saving the results

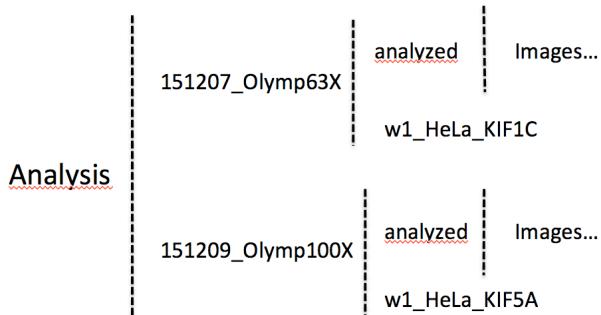
In the **output panel**, we can specify what segmentation result we want to export and save

- “**Contour images**” are the original images with the outline of cells segmented; it allows to visualize the segmentation results compared to the real images.
- “**Label images**” are simple images, with pixel value of 0 in the background, and a different value for each cells; these are the results we use to create FQ outlines.

Analysis and organization of results files

To launch the analysis on the whole image data set, use the tab “processing”. Press the “start processing” button to start the analysis.

The results will be saved in the output folder defined at the beginning of the analysis. *CellCognition* automatically creates 4 folders called ‘*hdf5*’, ‘*log*’, ‘*plots*’ and ‘*analyzed*’. If the *multiple plates* option is not checked, the set of 4 folders is created directly in the output folder defined at the beginning of the analysis. Otherwise, the 4 folders will be created in each subfolder corresponding to a plate.



The folders 151207_Olymp63X corresponds to a plate (an acquisition performed on a given day on a certain microscope), containing different subfolders such as w1_HeLa_KIF1C which corresponding to wells.

The segmentation results can be found in:

analyzed/images/labels/channel/segmentation_method/

The label images are named based on the information extracted with the regular expression:

P01_04_T00.tif

Where *P* is the position as explained above and *T* the time (always 1 in our case since we don't have time-resolved data).

Generate FQ outlines files from CellCognition results

The segmentation masks generated by *CellProfiler* can be used to automatically create FQ outline files with the middle part of the user interface.

Specify experimental parameters

First, you have to define the '**Experimental parameters**'. These are the same parameters that have to be defined for the images in the main FISH-quant interface when pressing the 'Modify' button in the panel 'Experimental parameters'. Simply adjust them to your data by changing the corresponding values.

Note 1: You have to ALWAYS define the parameters – even if the default parameters are good. Only then the button to generate the outlines will be enabled.

Note 2: you can change the default parameters in a text-file called 'FISH-QUANT_default_par.txt' which is located in the same directory as the *FQ_seg* Matlab file.

Define channel and segmentation method identifier

Second, you have to define the names of the two channels and the names of the segmentation method for the first and for the second channel. When saving the results, Cell Cognition creates a folder per channel, and a subfolder per segmentation method used to segment the channel. You have to indicate to the *FQ_Seg* which channel result you want to use, and from which segmentation method. The names have to correspond to these folder names.

Specify folder where the results will be saved

You only have the possibility to save the outlines file in a subfolder of the well folder of the projection. You can specify the name of the subfolder from the “Outline folder” of the GUI. Your outline will always be saved in “analysis→w1_HeLa_KIF1C→”Outline_folder”.

Define folder

You have to define one or multiple folder that **corresponds to a plate**.

Create outlines

Lastly, press the button ‘**Create FQ outlines**’. The script will then automatically search for the files describing the segmentation of cells and nuclei. For each image an outline file with the reference to the original 3D image will be generated and nuclei assigned to their respective cells.

5. Inspect outline files

The next few steps summarize a simple workflow for a manual inspection of the outlined cells

1. Open FQ and define the folders with images and the new outlines.
2. Open 'List directory content' from the tools menu. Here navigate to the folder with the outlines. Double-click of an outline will display it on the outline designer.
3. Click on the list with the different cells. You can use the arrows to navigate through this list (see shortcuts below).
4. Delete each cell that's not good with CTRL-X
5. Save outline with CTRL-S

The following short cuts help to speed up the analysis

CTRL-S	Save outline
CTRL-X	Delete currently selected cell <i>Note: works only when you clicked on the list with the cells before</i>
Arrow UP	Previous cell <i>Note: works only when you clicked on the list with the cells before</i>
Arrow DOWN	Next cell <i>Note: works only when you clicked on the list with the cells before</i>

6. References

1. Held, M. *et al.* CellCognition: time-resolved phenotype annotation in high-throughput live cell imaging. *Nat. Methods* **7**, 747–754 (2010).
2. Kamentsky, L. *et al.* Improved structure, function and compatibility for CellProfiler: modular high-throughput image analysis software. *Bioinformatics* **27**, 1179–1180 (2011).
3. Pertuz, S., Puig, D. & Garcia, M. A. Analysis of focus measure operators for shape-from-focus. *Pattern Recognition* **46**, 1415–1432 (2013).