Combining FIJI and ilastik to provide a versatile and adaptable approach to automated Bacteria cell classification and cell volume determination - CocciVol

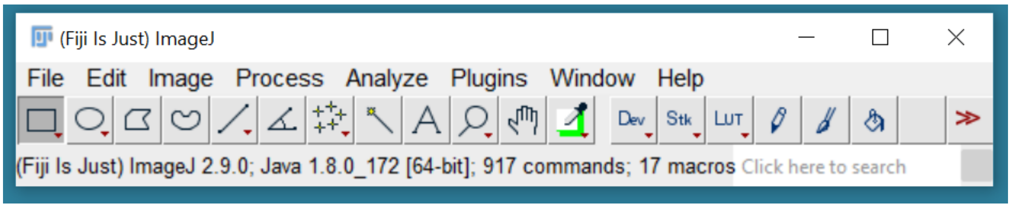
When studying bacteria cells' phenotype, one of the most common and well stablished approaches is to perform fluorescence microscopy. These images are used to determine the bacterial cells' volume as well as classify the cells based on their division stage on the cell cycle. Traditionally, this analysis process was performed manually, introducing user bias, systematic errors, and low throughput (spending many days or weeks to measure n=500 cells). Recently there has been a few examples of semi-automated or fully automated analysis for microbiology phenotyping (*e.g.* eHooke). However, these routines are highly specific to the experimental setup and therefore limited to having the same type of raw data (how many channels there are in the file and what order is the fluorescence channel versus the brightfield in the image); and the same labelling protocol (for example labelling the membrane instead of the cell wall). These differences usually vary between groups, making the current programs not versatile enough for the whole community. Here we propose a wholesome approach that can be adapted to each specific experiment. Using FIJI macros to standardise the raw data and the user-friendly machine learning approach of pixel classification provided by ilastik to segment, classify and measure the cells. This powerful combination of FIJI and ilastik (CocciVol) applied to bacterial cells provides an automated method with higher throughput of approximately n = 800 cells in 1h, which is a gamechanger for the microbiology community.

1. Open-source programs that should be installed (should work for both Windows and iOS)

This process of installing the required programs should not require previous computing skills but if there are any troubleshooting contact us (see below the contact information).

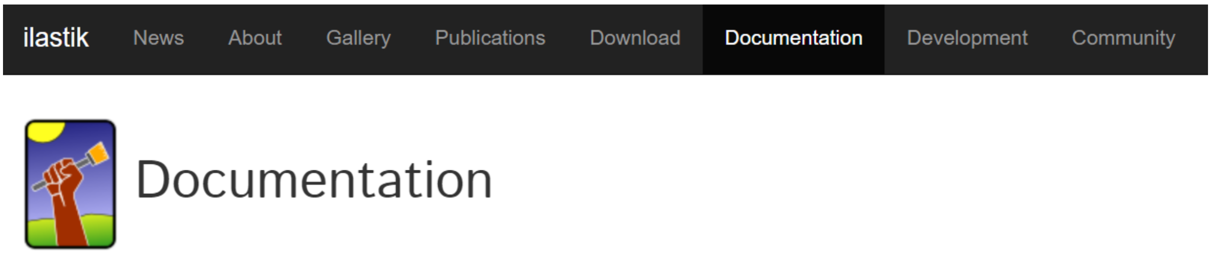
- **FIJI/ImageJ** (community-based software primarily to analyse fluorescence images)
    - ➢ Download from here following website instructions (documentation section).

This is how FIJI looks after being installed and initialised (*i.e.* started).

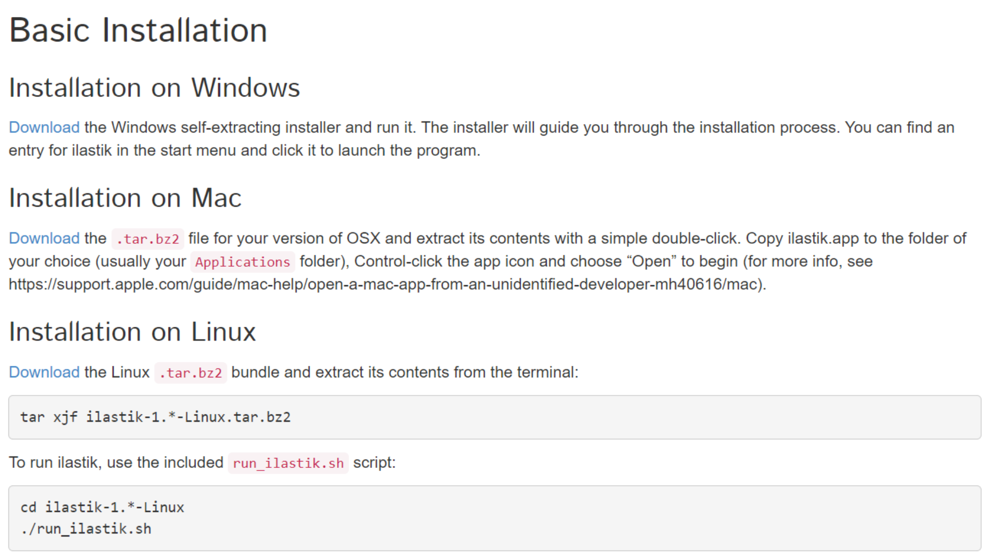


(Fiji Is Just) ImageJ 2.9.0; Java 1.8.0_172 [64-bit]; 917 commands; 17 macros

- **ilastik** (user-friendly machine learning approach to image segmentation and analysis)
    - ➢ Download from here following website instructions (documentation section).



Documentation



Basic Installation

Installation on Windows

Download the Windows self-extracting installer and run it. The installer will guide you through the installation process. You can find an entry for ilastik in the start menu and click it to launch the program.

Installation on Mac

Download the `.tar.bz2` file for your version of OSX and extract its contents with a simple double-click. Copy ilastik.app to the folder of your choice (usually your `Applications` folder), Control-click the app icon and choose "Open" to begin (for more info, see https://support.apple.com/guide/mac-help/open-a-mac-app-from-an-unidentified-developer-mh40616/mac).

Installation on Linux

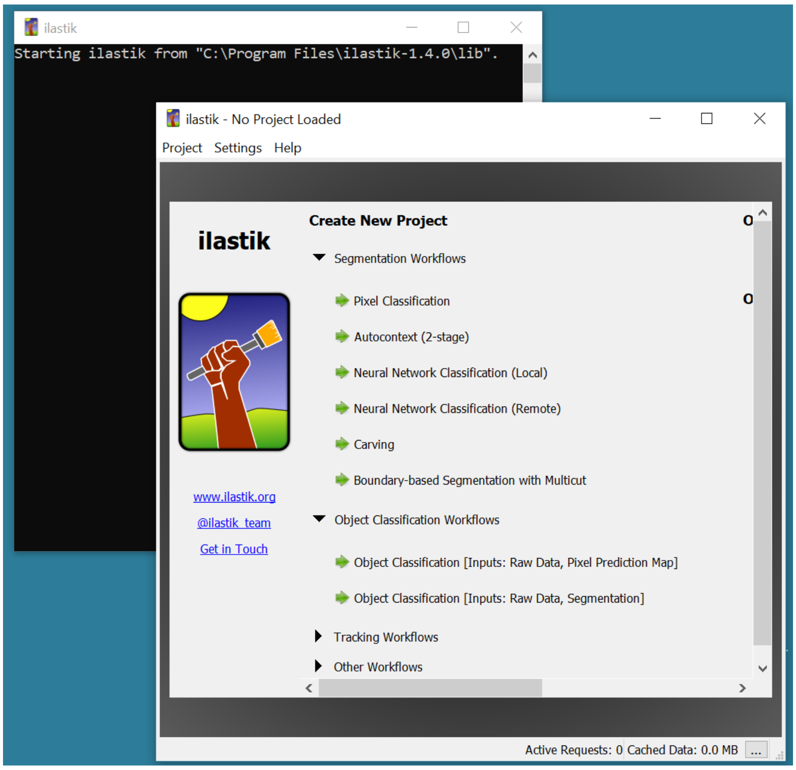
Download the Linux `.tar.bz2` bundle and extract its contents from the terminal:

```
tar xjf ilastik-1.*-Linux.tar.bz2
```

To run ilastik, use the included `run_ilastik.sh` script:

```
cd ilastik-1.*-Linux
./run_ilastik.sh
```

This is how ilastik looks after being installed and initialised in Windows.



```
Starting ilastik from "C:\Program Files\ilastik-1.4.0\lib".
```

ilastik
ilastik - No Project Loaded
Project Settings Help
ilastik
www.ilastik.org
@ilastik_team
Get in Touch
Create New Project
Segmentation Workflows
Pixel Classification
Autocontext (2-stage)
Neural Network Classification (Local)
Neural Network Classification (Remote)
Carving
Boundary-based Segmentation with Multicut
Object Classification Workflows
Object Classification [Inputs: Raw Data, Pixel Prediction Map]
Object Classification [Inputs: Raw Data, Segmentation]
Tracking Workflows
Other Workflows
Active Requests: 0 Cached Data: 0.0 MB

2. <u>Required files and plugins to perform analysis</u>

- **Dataset of microscopy images**

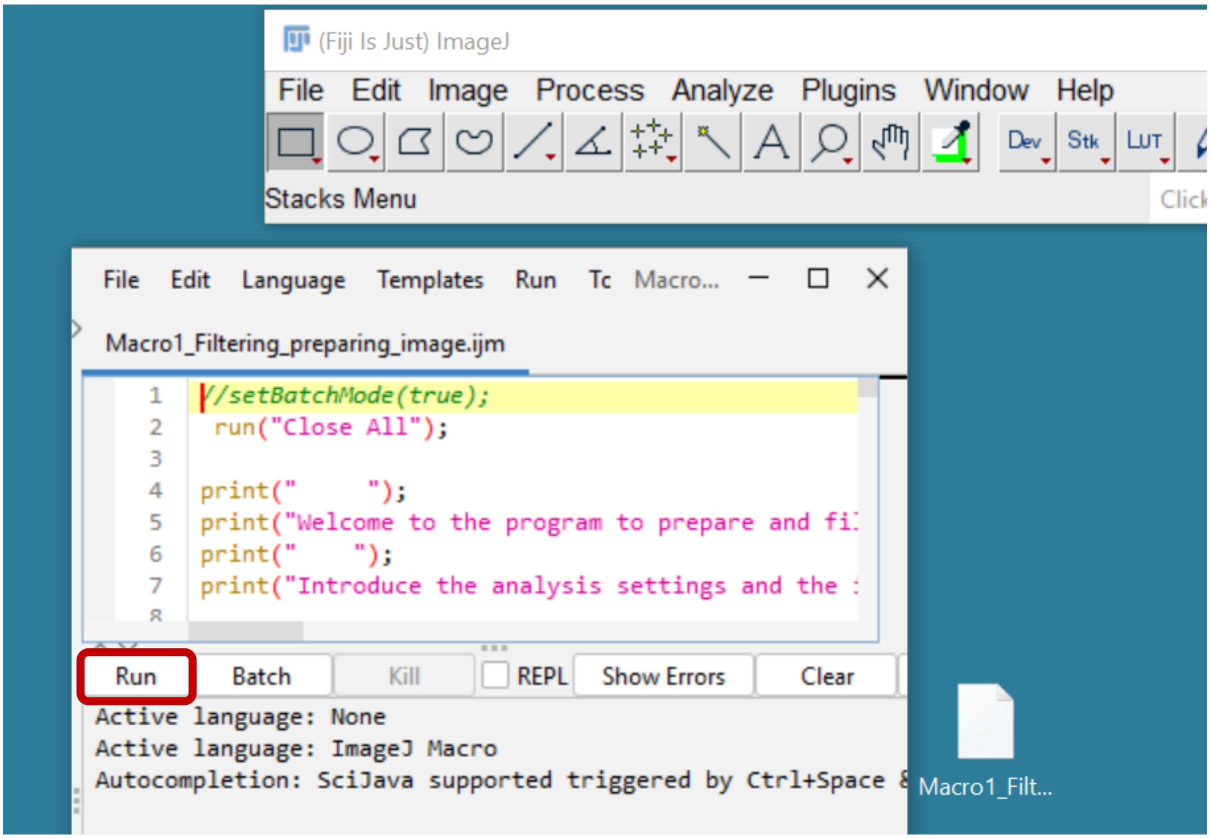
The first thing you need is a good set of images as your dataset. I recommend between 4-7 images of 160x140 um (the size can vary, see below, but this way the maximum amount of data is collected). This should be done for each sample, for example if you want to analyse a mutant with and without IPTG treatment, there should be two data sets of 7 images each to have 1 biological repeat. If a second biological repeat is required (it is recommended) then the amount of data should be doubled.

If the images are taken with a Nikon confocal microscope the format of the raw data files should be **.nd2.** However, because to read the raw data this approach uses the **Bioformats from FIJI**, any file format (for example .czi, .nd3, etc.) that is compatible with Bioformats should work.

- **2 Macros from FIJI**

The 2 macros are the following:

(*Macro1_Filtering_preparing_image.ijm* and *Macro2_Calculate_Volume_from_table.ijm*) from FIJI that are required to perform this analysis can be found in this GitHub link together with this documentation and a test sample (*image_raw_1.nd2*) to follow the video-tutorial. The macro files can be directly opened in FIJI and to run them simply click "run" as shown below:


(Fiji Is Just) ImageJ
File Edit Image Process Analyze Plugins Window Help
Dev Stk LUT
Stacks Menu
Click
File Edit Language Templates Run Tc Macro...
Macro1_Filtering_preparing_image.ijm
Run
Batch
Kill
□ REPL
Show Errors
Clear
Macro1_Filt...

```
//setBatchMode(true);
 run("Close All");

print("      ");
print("Welcome to the program to prepare and fi
print("     ");
print("Introduce the analysis settings and the :
```

```
Active language: None
Active language: ImageJ Macro
Autocompletion: SciJava supported triggered by Ctrl+Space &
```

There is no need to alter the macro code at all to perform this analysis, however if you feel confident and want to add some specific actions regarding your particular experiment, please download the code in your computer and alter it there, then upload it back on the GitHub with a different name (*e.g.* Macro1_XXX, where XXX are your initials).

- **2 ilastik projects**

There are two situations that you can find yourself when approaching this routine:

Situation 1:

You want to analyse a dataset of a bacterial strain (*e.g.* WT SH1000) that has been analysed before by someone else and the ilastik project contains the "trained" version of the program for this strain. Therefore, the analysis is much straightforward as it only requires you to perform batch analysis (see below on section 7 "Batch processing"). The two ilastik "trained" projects that you need should look like this:

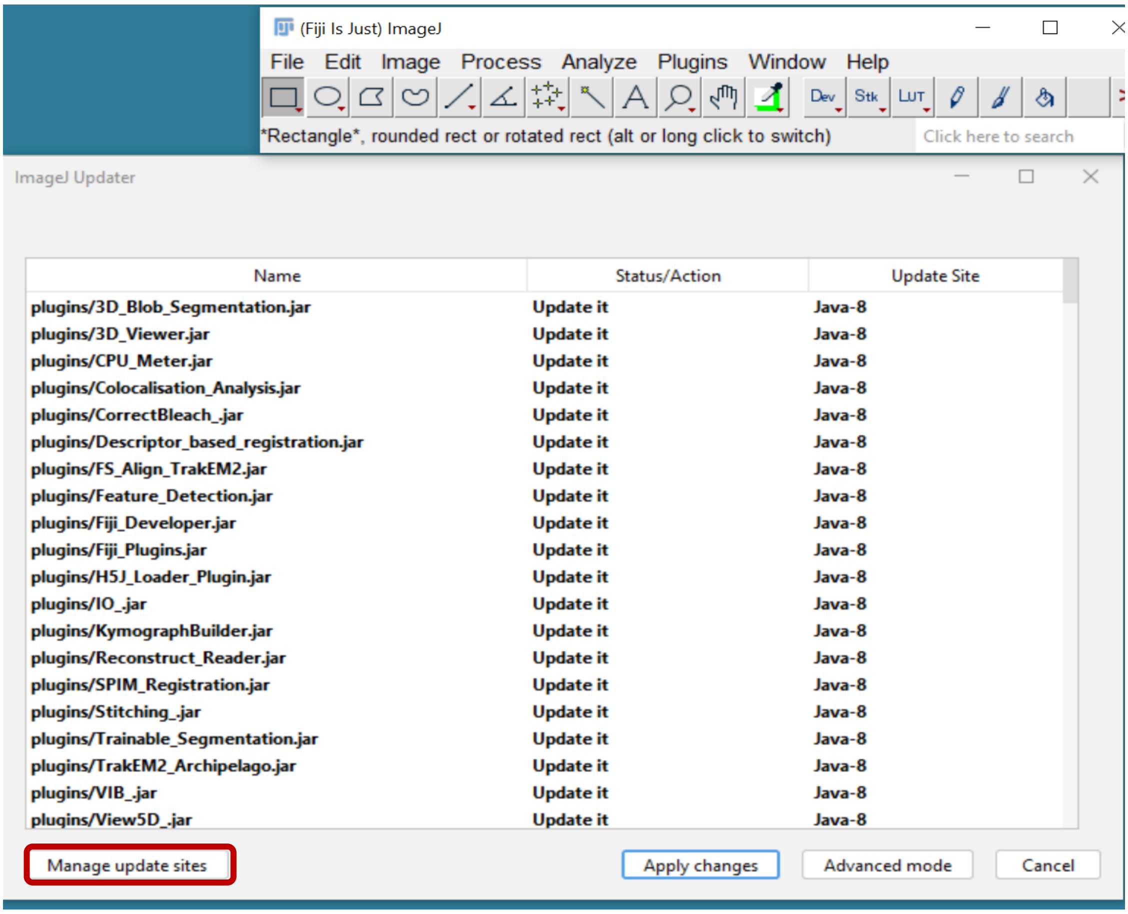
MyProject_STRAIN_XXX_object MyProject_STRAIN_XXX_pixel where STRAIN_XXX is the name of your strain.

Situation 2:

You want to analyse a dataset of a bacterial strain (*e.g.* WT SH1000 $\Delta tarO$) that has never been analysed before with this routine. This is the most common situation you are going to find yourself, because it is more likely that you are the first researcher working in your specific strains. Therefore, you need to create two projects of ilastik from scratch and train the system (see below in sections 5 and 6 "Pixel and Object classification, training ilastik 1 and 2").

- **Install ilastik plugin in FIJI (correlating both programs)**

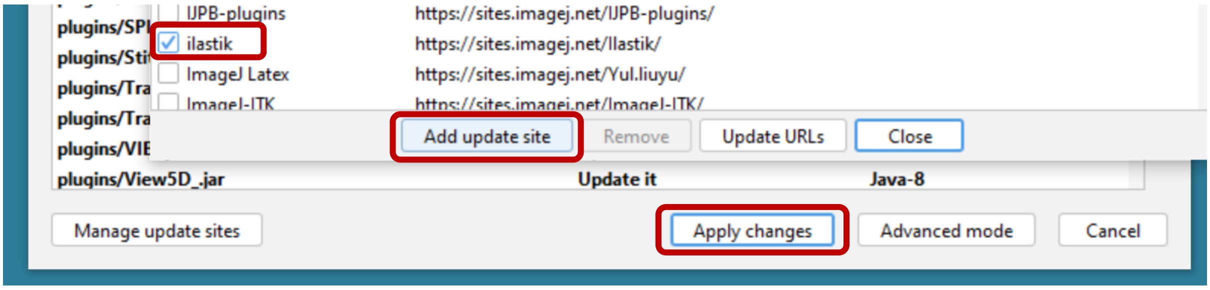
Plugins are like "tools" that can be added to FIJI. The best way to do this is going to FIJI and click in: *Help->Update…* Sometimes the system requires updating because some modification has been added and no dialog will appear. Just close FIJI and open it again and repeat the process until the dialog with a list of things appear. Then click the button "Manage update sites":



Rectangle, rounded rect or rotated rect (alt or long click to switch)

Name	Status/Action	Update Site
plugins/3D_Blob_Segmentation.jar	Update it	Java-8
plugins/3D_Viewer.jar	Update it	Java-8
plugins/CPU_Meter.jar	Update it	Java-8
plugins/Colocalisation_Analysis.jar	Update it	Java-8
plugins/CorrectBleach_.jar	Update it	Java-8
plugins/Descriptor_based_registration.jar	Update it	Java-8
plugins/FS_Align_TrakEM2.jar	Update it	Java-8
plugins/Feature_Detection.jar	Update it	Java-8
plugins/Fiji_Developer.jar	Update it	Java-8
plugins/Fiji_Plugins.jar	Update it	Java-8
plugins/H5J_Loader_Plugin.jar	Update it	Java-8
plugins/IO_.jar	Update it	Java-8
plugins/KymographBuilder.jar	Update it	Java-8
plugins/Reconstruct_Reader.jar	Update it	Java-8
plugins/SPIM_Registration.jar	Update it	Java-8
plugins/Stitching_.jar	Update it	Java-8
plugins/Trainable_Segmentation.jar	Update it	Java-8
plugins/TrakEM2_Archipelago.jar	Update it	Java-8
plugins/VIB_.jar	Update it	Java-8
plugins/View5D_.jar	Update it	Java-8

Then this other dialog will appear, it is a list of all possible plugins. Scroll down until you find “ilastik” it is in alphabetical order. Then click the box on the left of ilastik and click the button “Add update site”. Then the second dialog box will close and you can click “Apply changes” into the first dialog box and the update will proceed, installing the “ilastik”plugin. Then close FIJI and open again. You should be able to find a menu named “ilastik” at the very bottom of *FIJI->Pluggins.*



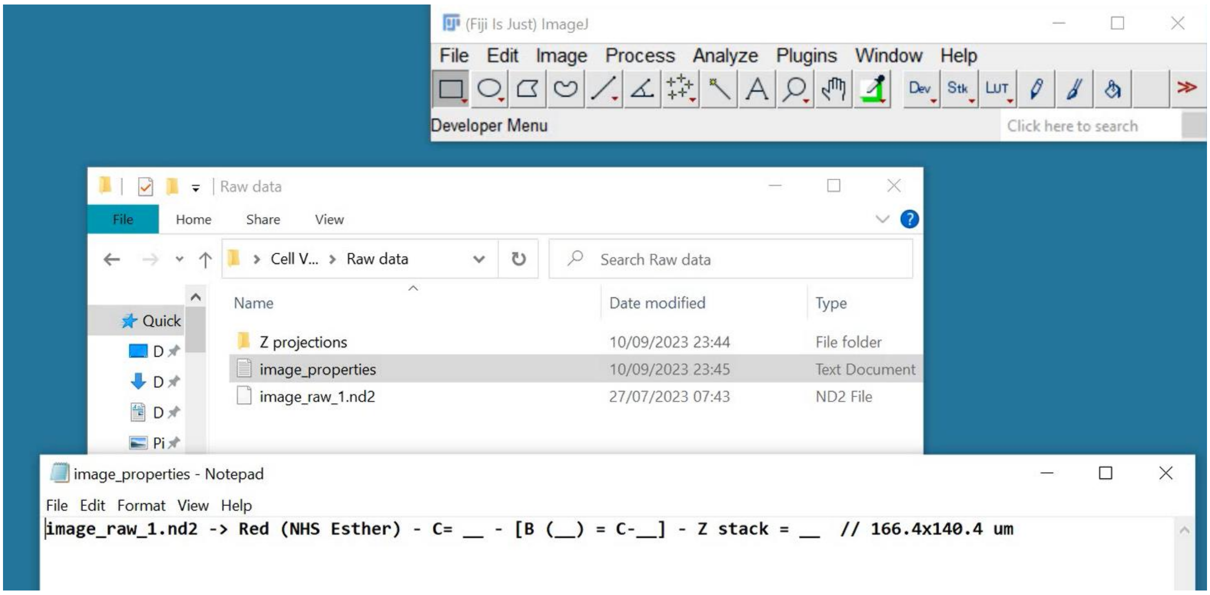
□ IJPB-plugins	https://sites.imagej.net/IJPB-plugins/
☑ ilastik	https://sites.imagej.net/Ilastik/
□ ImageJ Latex	https://sites.imagej.net/Yul.liuyu/
□ ImageJ-ITK	https://sites.imagej.net/ImageJ-ITK/

plugins/View5D_.jar	Update it	Java-8

3. Sort and prepare your files/folders

The first thing you’ll need to do when working with fluorescent images is create a Z projection of a few slices that contain the focus of your cells within your stack of several images (*e.g.* 11). However, because every experiment is different, even from image to image in the same experiment there is variability, each image from your dataset that you want to analyse will require a Z projection in a different position of the stack. Moreover, to increase reproducibility among different experiments, your dataset might contain a Brightfiel channel, or not, the dye of your fluorescent channel could be blue, green or red and the position of this channel could be different in each case. Therefore, to standardise the datasets among different experiments, we need to know all this information for each image and write it down in a text file. Then, we will apply *Macro1_Filtering_.... ijm* to obtain the same standardised image format to be used by ilastik (see below section 4 “Filtering process”).

The way I recommend doing this annotation process for all your images is the following. Create three empty folders where you want to analyse your data: “Raw data”, “Data for ilastik” and “Results”. Then inside your “Raw data” put another empty folder named “Z projections”. Inside the raw data put your dataset (e.g. image_raw_1.nd2) and a text file: “image_properties.txt”. Open each image in FIJI (click “OK” on the Bioformats window). Inside the text file make a list of all your images (*e.g.* image_raw_1.nd2, image_raw_2.nd2, ...) and next to each image put the following:



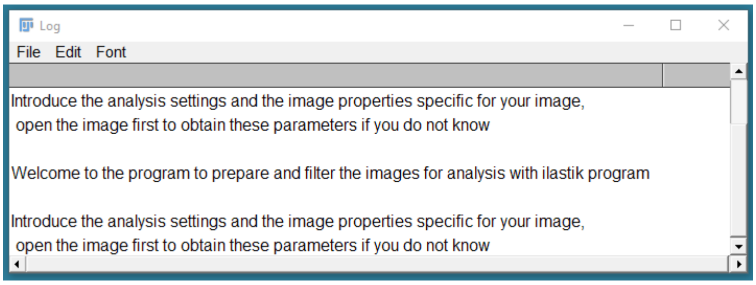
Name	Date modified	Type
Z projections	10/09/2023 23:44	File folder
image_properties	10/09/2023 23:45	Text Document
image_raw_1.nd2	27/07/2023 07:43	ND2 File

```
image_raw_1.nd2 -> Red (NHS Esther) - C= __ - [B (__) = C-__] - Z stack = __  // 166.4x140.4 um
```

The colour and type of dye (*e.g.* Red, NHS Ester or Green, HADA), then the channel where the fluorescent is found (*e.g.* C = 1). Then there is the question of if this image has a brightfield channel, put YES or NO, *e.g.* B(YES) and in which channel the brightfield is (*e.g.* C = 2). If there is no brightfield channel the place where the brightfield should be is C = 0. Then, scroll through the stack of your opened image in FIJI in the fluorescent channel and try to find the slices in the stack that appear to be more in focus (usually they should be 2 or 3 slices but it could be different depending on the settings of your experiment). Then write down in the text file the slices that you have chosen to do the Z stack (*e.g.* z stack = 5-7). Finally, at the top of the image in FIJI there should be the dimensions written in pixels and µm. Write the dimensions for later (*e.g.* 166.4x140.4 µm).

4. <u>Filtering process (preparing raw data)</u>

First we need to filter all the images to standardise them into the same format, one by one (potentially in future updates there could be a batch processing option to filter all images at once). Drag the *Macro1_Filtering_preparing_image.ijm* into the lower bar in FIJI or click *File->Open…* then just click "Run". The macro will start, and two pop-up windows will appear. The first is the "Log windows", which contains messages with instructions of what to do at each time in the program:

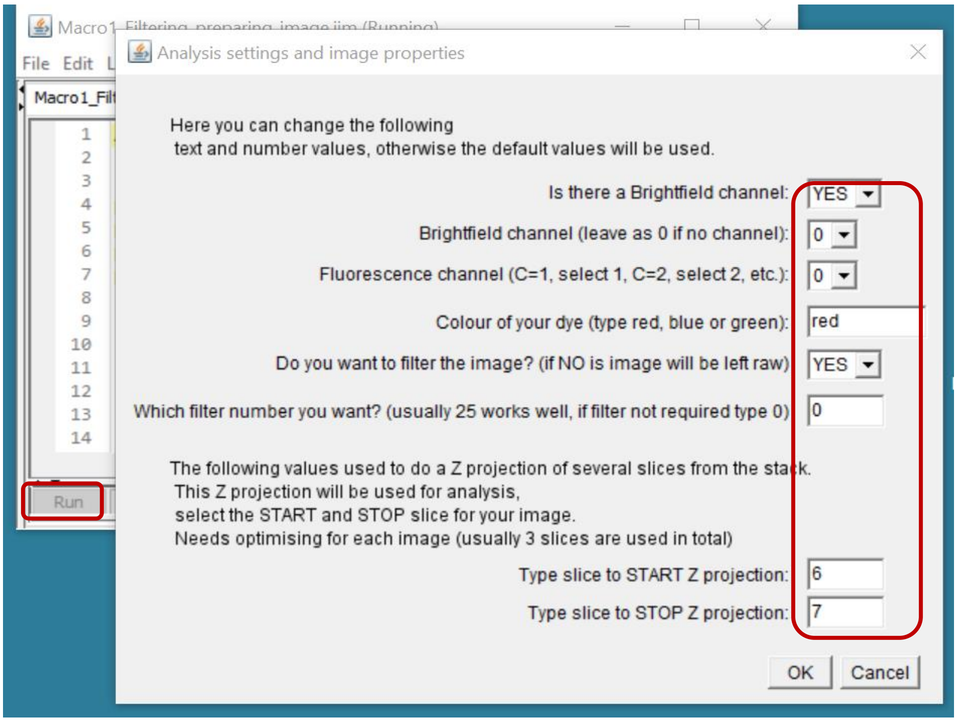


```
Introduce the analysis settings and the image properties specific for your image,
 open the image first to obtain these parameters if you do not know

Welcome to the program to prepare and filter the images for analysis with ilastik program

Introduce the analysis settings and the image properties specific for your image,
 open the image first to obtain these parameters if you do not know
```

The second Windows is a dialog where you can interact with the macro and set-up some parameters specific to your image and experiment. This window is named "Analysis and image properties":


Macro1_Filtering_preparing_image.ijm (Running)
File Edit L
Macro1_Fil
Run
Analysis settings and image properties
OK
Cancel

Here you can change the following
text and number values, otherwise the default values will be used.

Is there a Brightfield channel: YES

Brightfield channel (leave as 0 if no channel): 0

Fluorescence channel (C=1, select 1, C=2, select 2, etc.): 0

Colour of your dye (type red, blue or green): red

Do you want to filter the image? (if NO is image will be left raw) YES

Which filter number you want? (usually 25 works well, if filter not required type 0) 0

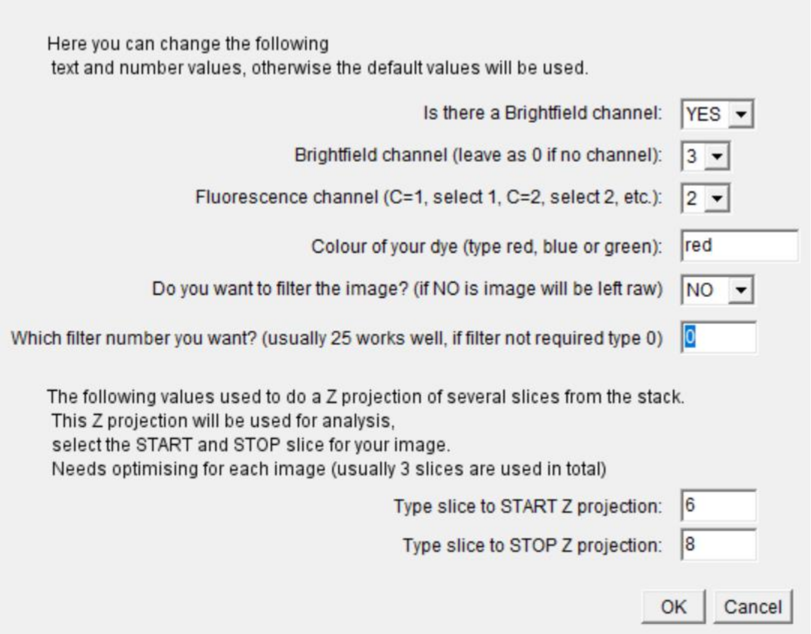
The following values used to do a Z projection of several slices from the stack.
This Z projection will be used for analysis,
select the START and STOP slice for your image.
Needs optimising for each image (usually 3 slices are used in total)

Type slice to START Z projection: 6

Type slice to STOP Z projection: 7

The highlighted column on the right is where we need to introduce all the parameters that we wrote before in our text file “image_properties.txt”. Just follow the instructions from the text in the window and click on the down arrows for a drop-down menu with the options in some of the cases.

See an example of the parameters used for the training image “image_raw_1.nd2”:



Here you can change the following
text and number values, otherwise the default values will be used.

Is there a Brightfield channel: YES

Brightfield channel (leave as 0 if no channel): 3

Fluorescence channel (C=1, select 1, C=2, select 2, etc.): 2

Colour of your dye (type red, blue or green): red

Do you want to filter the image? (if NO is image will be left raw) NO

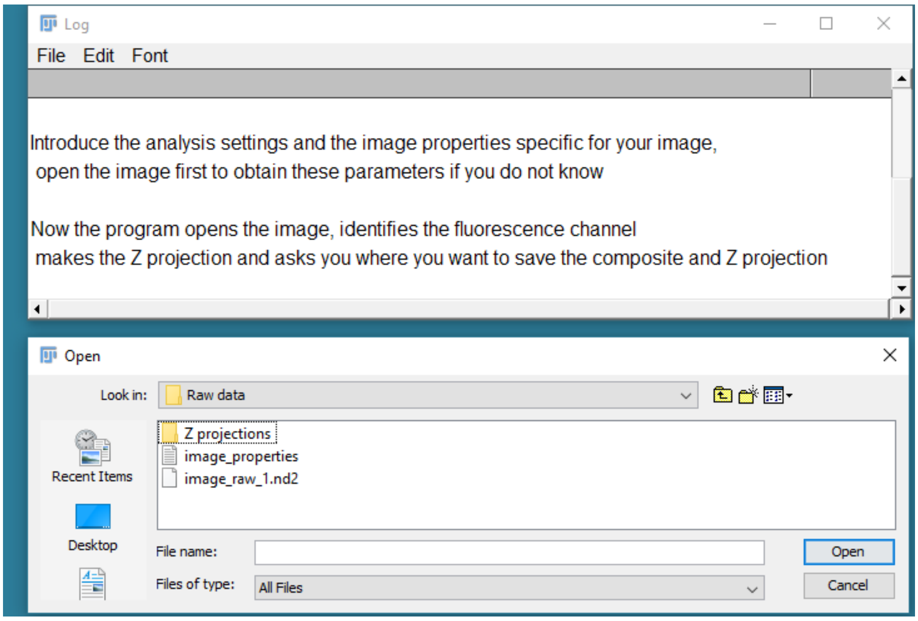
Which filter number you want? (usually 25 works well, if filter not required type 0) 0

The following values used to do a Z projection of several slices from the stack.
This Z projection will be used for analysis,
select the START and STOP slice for your image.
Needs optimising for each image (usually 3 slices are used in total)

Type slice to START Z projection: 6

Type slice to STOP Z projection: 8

The only thing not mentioned before is the filtering step. In this dialog window there is an option if you want to filter the image “YES” or “NO”. If the option of filtering is clicked “YES” it would filter the background. Only do this in case there is fluorescence signal in the background, if not the data is better without filtering. In case of the filter try numbers 20-30 (where usually 25 works better). After all the parameters are set click “OK” to continue with the code. Then this happens:

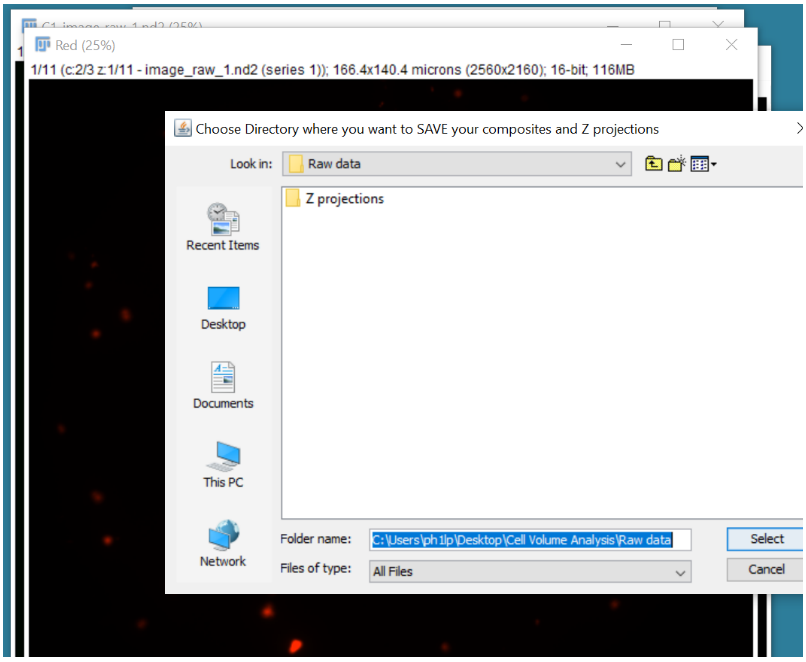


```
Introduce the analysis settings and the image properties specific for your image,
 open the image first to obtain these parameters if you do not know

Now the program opens the image, identifies the fluorescence channel
 makes the Z projection and asks you where you want to save the composite and Z projection
```

Now you need to open the image that you wanted to filter. Go through your folders, select the image and click “Open”. You can see this also explained in the new two lines in the Log Windows.

Now three images will automatically appear. One is the original image that got opened. The other two windows open are the Brightfield channel and the fluorescent channel which are now separated from the original file. Then, the program will do the Z projection of the slices you have selected (in this case 6 to 8), but first it asks the user to choose the location where the Z projection is saved:



Red (25%)

1/11 (c:2/3 z:1/11 - image_raw_1.nd2 (series 1)); 166.4x140.4 microns (2560x2160); 16-bit; 116MB

Choose Directory where you want to SAVE your composites and Z projections

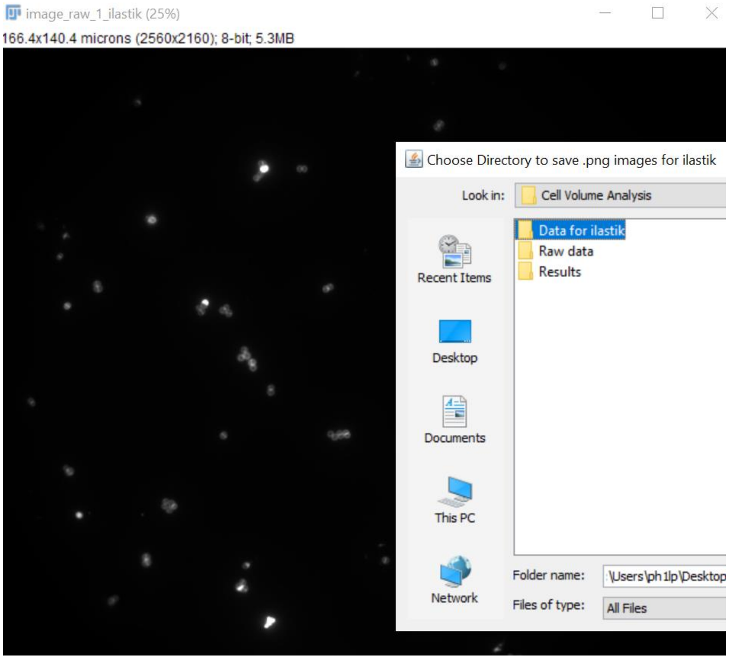
Look in: Raw data

Z projections

Folder name: C:\Users\ph1lp\Desktop\Cell Volume Analysis\Raw data

Files of type: All Files

Select the empty folder “Z projections” inside the folder “Raw data” that we did before and click “Select”. Then the program saves the Z projection, and this Z projection is converted into grey scale by splitting the channels into Red, Green and Blue. Depending on the colour of your dye, the program selects the correct window and close the rest. Then this is exported in .png and is the correct/standardised image format to be introduced to ilastik. This is why the program asks the user to save the image for ilastik:



image_raw_1_ilastik (25%)

166.4x140.4 microns (2560x2160); 8-bit; 5.3MB

Choose Directory to save .png images for ilastik

Look in: Cell Volume Analysis

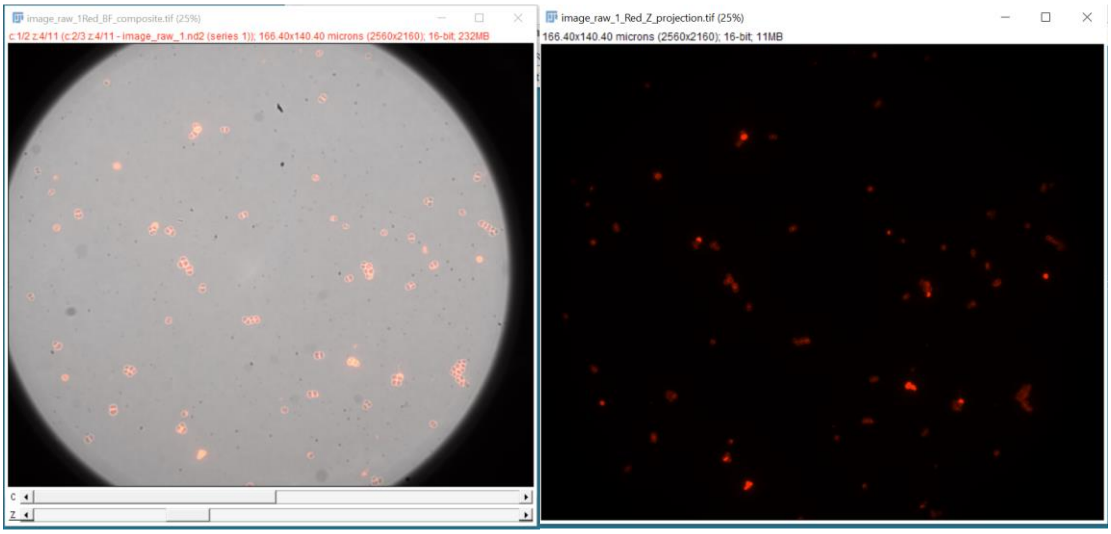
Data for ilastik
Raw data
Results

Folder name: \Users\ph1lp\Desktop

Files of type: All Files

The empty folder that was created before called “Data for ilastik” gets selected and then click “select”. The program closes all the windows and the macro 1 is finished.

If you open the files from the Z projections folder in FIJI you will see an image named: “your_image_name_Red_BF_composite.tif” which is the overlap of the Brightfield image and the fluorescence (see below on the left). If your data has no brightfield channel this file should not appear in the folder. However, if you have this image, I recommend you save it as it can be shared in an open access repository to show where the raw data from your analysis comes from. A second image is saved in the Z projections folder named: “your_image_name_Red_Z_projection.tif” which is the Z projection used for the ilastik file but still keeping the original colour of the dye (see below on the right). If your dye is a different colour than Red it should be named accordingly.



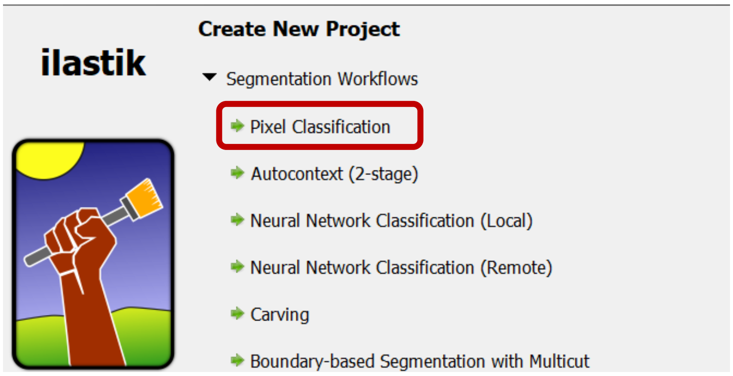
image_raw_1Red_BF_composite.tif (25%)
c:1/2 z:4/11 (c:2/3 z:4/11 - image_raw_1.nd2 (series 1)); 166.40x140.40 microns (2560x2160); 16-bit; 232MB

image_raw_1_Red_Z_projection.tif (25%)
166.40x140.40 microns (2560x2160); 16-bit; 11MB

After this process is finished, repeat for all the images in your dataset. Then, once all the images are in the standardise format .png in the folder “Data for ilastik” have a look at the files and pick an image that looks like it’s representative of the Dataset (not too clumpy, not too loose single cells). This is going to be your **training image** for this Dataset. Once you want to analyse a different strain, so that is a different dataset, you’ll have to choose another training image.

5. Pixel classification (training ilastik 1)

Open ilastik and you’ll see the window with the options of what type of project you want to create. We will first use a Pixel classification project because we need to train the program to differentiate between the pixels of the background and the pixels corresponding to the bacterial cells.



ilastik

Create New Project

Segmentation Workflows
- Pixel Classification
- Autocontext (2-stage)
- Neural Network Classification (Local)
- Neural Network Classification (Remote)
- Carving
- Boundary-based Segmentation with Multicut

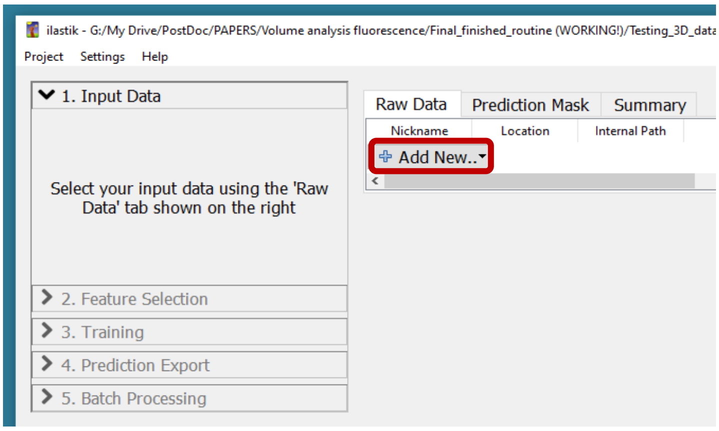
Then immediately after you must choose a name (I recommend: “MyProject_STRAIN_XXX_pixel”) and a place to save the project.

! WARNING! once you choose a place it cannot be move unless you copy the whole folder where the ilastik project and the training image are located. So be careful where you put it.

Once the project opens, you’ll see 5 steps but without completing 1 you cannot go onto step 2 and so forth. Here each step is explained in detail.

- **Input Data**

First you need to add the training image into the Raw data section. Click “Add new…” and select the image.

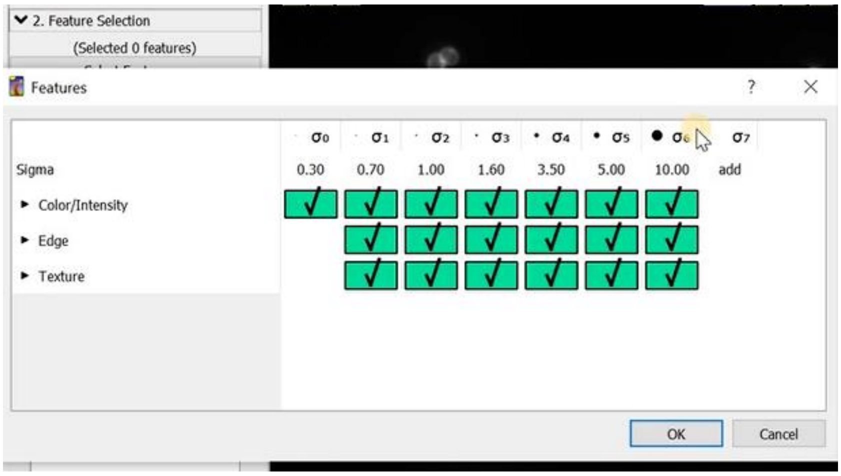

ilastik - G:/My Drive/PostDoc/PAPERS/Volume analysis fluorescence/Final_finished_routine (WORKING!)/Testing_3D_data

Select your input data using the 'Raw Data' tab shown on the right

! IMPORTANT! The training image needs to be in the same folder or subfolder where the project has been saved. Therefore, I recommend having a “main folder” where you save all the files related to a particular strain or dataset and in this main folder you save the ilastik project as well as the three subfolders we create before: the “Raw data”, “Results” and “Data for ilastik” which is where the training image will be located.

- **Feature selection**

This section is to select the features (colour, intensity, edges, texture and size) that differentiate the two types of pixels. In this case, we need to differentiate between background pixels (black, low intensity, uniform) and the pixels that correspond to cells (which are white, brighter, less uniform). It works better if you select all possible sizes. But for **data in 3D** avoid the size 5 and 10 for it to work.

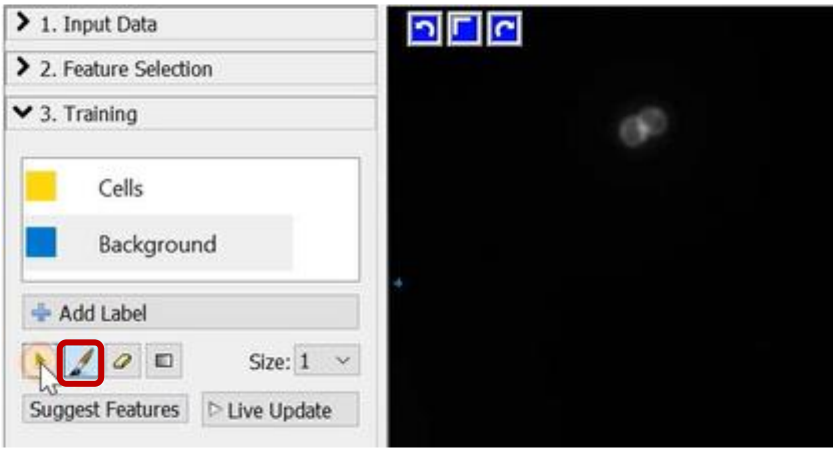

2. Feature Selection
(Selected 0 features)

	σ_0	σ_1	σ_2	σ_3	σ_4	σ_5	σ_6	σ_7
Sigma	0.30	0.70	1.00	1.60	3.50	5.00	10.00	add
▸ Color/Intensity	☑	☑	☑	☑	☑	☑	☑	
▸ Edge		☑	☑	☑	☑	☑	☑	
▸ Texture		☑	☑	☑	☑	☑	☑	

Click “OK”, then you should see roughly that “37 features” are selected. Then move to step 3.

- **Training**

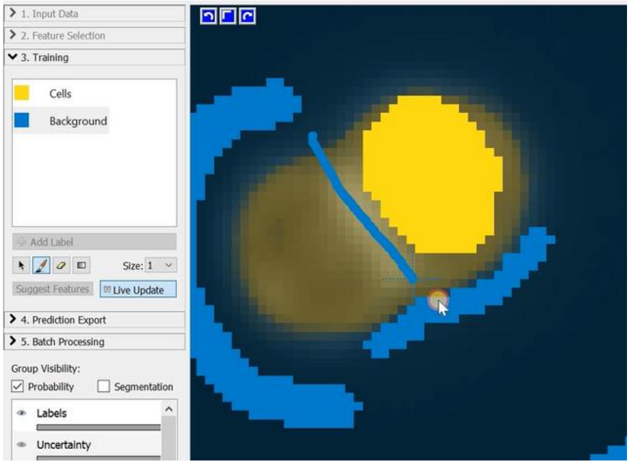
This is the bulk of the training for this ilastik project. For a new strain, depending on the variability among the cells, this part can last up to 30 minutes. This is where instead of training a Machine Learning system with code and many images, we train it by drawing our selections on which pixels are which (cells or background) and we make an **ilastik mask**. For each “drawing” the program learns and gets trained using the example of every single pixel in the image. Therefore, instead of having many images to train the system, we only need 1 representative image with many pixels (always use the maximum possible according to your microscope and objective). Write next to the “yellow colour” that these pixels will be cells and next to the “blue colour” that these pixels will be background.



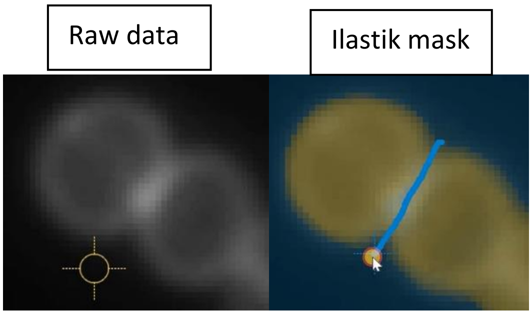
Then use the “brush tool” to paint the different areas and the “cursor button” on the left of the brush to select which colour you are drawing each time. For better understanding of this section **watch the video tutorial here.**

It is always better to click the “Live update” button so you can see how the system “learns” from each brush and changes it’s predictions for each pixel.

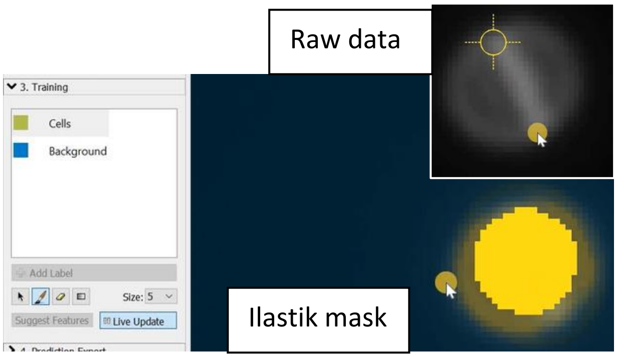
You can change the size of the brush depending on the type of feature you are selecting. See the blue brush with size 1 in the middle of the picture, next to the brush number 7 in yellow.



Because of the type of sample (bacterial cells that are dividing) sometimes two cells are divided but are still close together and we need to tell the program that that area is background so it differentiates that they are two different cells (see image in the left – Splitting cells, these are no septa cells). However, sometimes, the septa are incomplete or the cells still haven't divided even if the septa is complete, and we need to tell the program that those pixels belong to the same cell by just drawing all of it in yellow (see right – complete septa, but do the same for incomplete septa).



Splitting Cells



Complete Septa

These are contradictory messages for the program because the intensity and colour of the pixels in the middle in both situations is the same (the middle between splitting cells and the septa of one cell). But after doing this sufficient times, the program learns the difference by looking at the neighbouring pixels and realising that in one case there are two cells and in the other case there is only one cell. If this step is not perfect, not to worry because in section 6 there is an extra step that can help separate the cells that are still together after this initial training.
When working with strains that cause a lot of cell wall defects, it is important to differentiate clearly what is considered a cell or what is considered two. Therefore, once an ilastik project is trained on a strain it might be that it doesn't work on another. Also if the size of the cells change it won't work as well. In summary, it is recommended to train a project for one training image of each dataset.

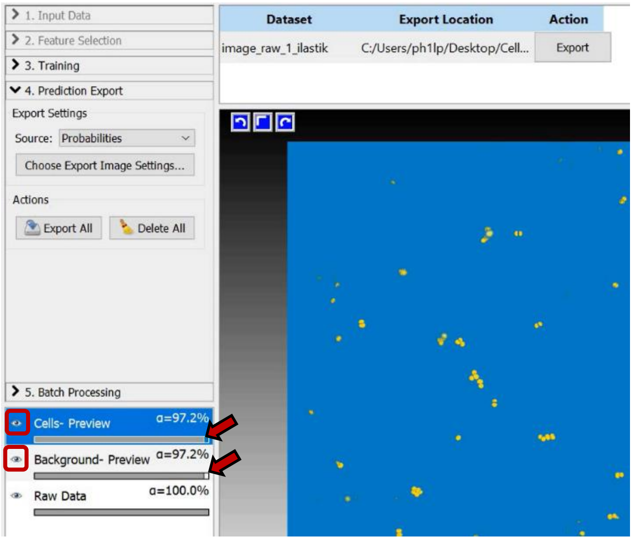
During this process there are some useful keyboard shortcuts that are crucial for a speedy process:

CTRL + "mouse click" -> Drag image to move to another place
CTRL + "mouse wheel" -> Zoom in and out of the same place
CTRL **+ "z"** -> Remove last stroke of brush in case of mistake
"i" -> To visualize raw data (which you need to do constantly to know how the mask fits).

Then once you are happy with the training part and most of the cells are correctly masked, move on to the next step.

- **Prediction export**

This is about exporting the predictions that the program made of pixel classification based on your training. First, to double check that you are happy with your training section, click on the "Cells – preview" and "Background – preview" eye icon on the left, and make sure the visibility is almost at 100%. Then, you should see your whole image with the background in a dense blue colour and the cells in dense yellow (it might take a bit to apply these settings to all the image). See the resultant image below:



Dataset	Export Location	Action
image_raw_1_ilastik	C:/Users/ph1lp/Desktop/Cell...	Export

Source: Probabilities

Cells- Preview α=97.2%
Background- Preview α=97.2%
Raw Data α=100.0%

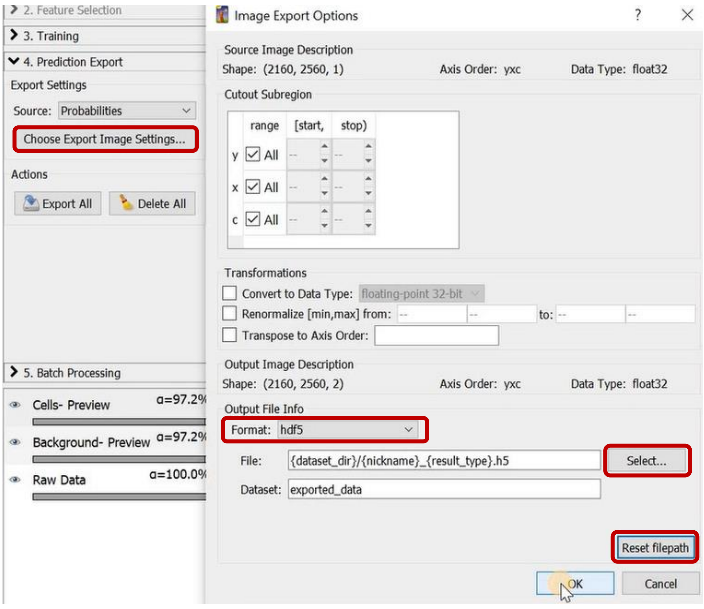
Then, once you want to save the mask and move on to the next step of the routine, you need to export the mask which is named **"Probabilities"** in the Export Settings, under the "Source:" scroll down menu. Then click the button "Choose Export Image Settings…", a pop-up window will open named "Image Export Options".

! IMPORTANT! The output file format must be in **"hdf5"** which is the format from ilastik.

Find the folder where you want to save your file, I recommend the folder that was created beforehand called "Data for ilastik", by clicking "Select". The name of the file will be: "**YourImageName_ilastik_Probabilities.h5**".

! IMPORTANT! For the future use of the batch processing tools to process all your images of your dataset, it is crucial that the folder to save your data is the correct one for this sample. Therefore, remember clicking the button called "Reset filepath".

Then click "OK" and the file will be exported. The pop-up window will close. Then finally go under the section called **Actions** and click the button "Export All". Save the project: *Project -> Save* and close it.



Source: Probabilities

Cells- Preview α=97.2%
Background- Preview α=97.2%
Raw Data α=100.0%

Source Image Description
Shape: (2160, 2560, 1) Axis Order: yxc Data Type: float32

y ☑ All
x ☑ All
c ☑ All

☐ Convert to Data Type: floating-point 32-bit
☐ Renormalize [min,max] from: -- -- to: -- --
☐ Transpose to Axis Order:

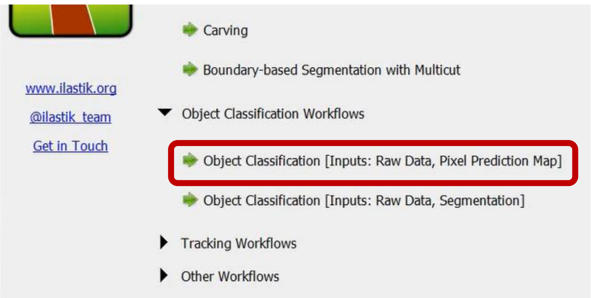
Output Image Description
Shape: (2160, 2560, 2) Axis Order: yxc Data Type: float32

Format: hdf5
File: {dataset_dir}/{nickname}_{result_type}.h5
Dataset: exported_data

6. Object classification (training ilastik_2)

After you have completed steps 1-5 the first ilastik project is completely trained and can be used further for analysis of other images from the same sample using the "Batch processing" approach in section 7 below. Now, the second ilastik project needs to be created and trained to differentiate between different types of object of interest. This is the part where the program is trained to recognise cells that correspond to different parts of the cell cycle (*e.g.* No septa, complete septa, …).

Open ilastik again and create a new project. Go under "Object Classification Workflows" and select the first option which is called "Object Classification [Inputs: Raw data, Prediction Map]".


www.ilastik.org
@ilastik_team
Get in Touch
Carving
Boundary-based Segmentation with Multicut
Object Classification Workflows
Object Classification [Inputs: Raw Data, Pixel Prediction Map]
Object Classification [Inputs: Raw Data, Segmentation]
Tracking Workflows
Other Workflows

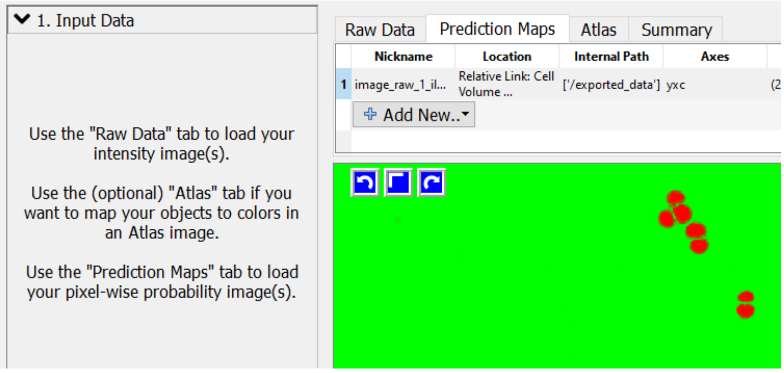
Then immediately after you must choose a name (I recommend: "MyProject_STRAIN_XXX_object") and a place to save the project.

! WARNING! once you choose a place it cannot be move unless you copy the whole folder where the ilastik project and the training image are located. So be careful where you put it.

Then similarly to the pixel classification project, there are several steps that need further explaining:

- **Input Data**

First you need to add the **"training image"** into the Raw data section (it needs to be the same one as you added in section 5 during the pixel training). Click "Add new…" and select the image. Then you need to add the probability mask that was just exported from the ilastik pixel classification project. Go to "Prediction Maps" section and click "Add new…". A green and red image appears. The file should be named: "**YourImageName_ilastik_Probabilities.h5**".



Use the "Raw Data" tab to load your intensity image(s).

Use the (optional) "Atlas" tab if you want to map your objects to colors in an Atlas image.

Use the "Prediction Maps" tab to load your pixel-wise probability image(s).

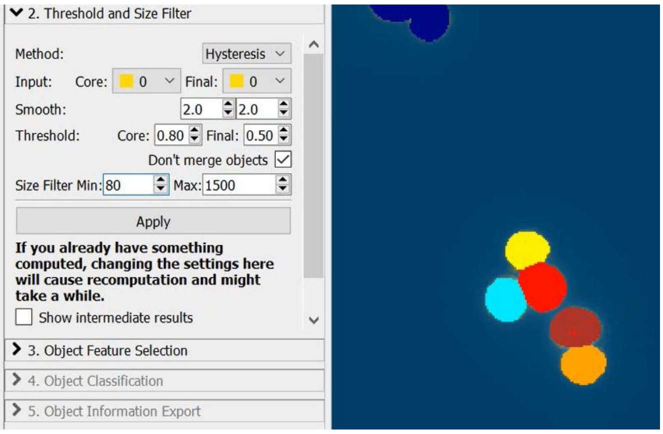
	Nickname	Location	Internal Path	Axes	
1	image_raw_1_il...	Relative Link: Cell Volume ...	['/exported_data']	yxc	(2

- **Threshold and Size Filter**

This section selects the objects to be analysed. It could be for example that during your pixel segmentation there are either too small or too big objects that you do not want to analyse. Apply the following numbers to start with. It is important you use the **Method "Hysteresis"** because that allows for the button called "Don't merge objects" to appear. Which as you can see in the video tutorial, it makes a big difference by separating those cells that are wrongfully joined together. In this stage, each cell will have a colour to identify it as being an independent object, which makes it easier to distinguish if for example a diploid (two joined cells together) has separate colours.

Modify the **Threshold values** (Core: can be 0.4-0.9 and Final: can be 0.3-0.8). These are what usually works, but for your sample it could be different.

Modify the **Size filter Min** (which is in pixels) depending on your type of data. For 2D fluorescence *cocci* shaped bacterial cells usually values: 80-3000 work well. For 3D fluorescence dataset, these values should be tripled. For other bacteria, adapt so that you select most of your cells of interest.



2. Threshold and Size Filter

Method: Hysteresis

Input: Core: 0 Final: 0

Smooth: 2.0 2.0

Threshold: Core: 0.80 Final: 0.50

Don't merge objects ☑

Size Filter Min: 80 Max: 1500

Apply

If you already have something computed, changing the settings here will cause recomputation and might take a while.

☐ Show intermediate results

3. Object Feature Selection

4. Object Classification

5. Object Information Export

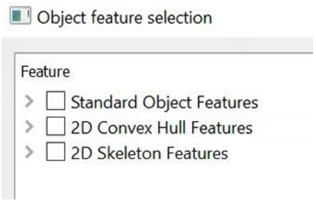
! WARNING! It is more important that you choose some parameters in this section that allow for most of the selected cells to be analysable later, rather that having a situation where you select almost all the cells but they have a bad fit and they can't be analysed. Although there is a way to remove the "bad fit" cells later, the least there are in the image the better the predictions are going to work for the batch processing of other images.

- **Object Feature Selection**

In this section you need to select the features which you plan to use later in the workflow. They are used not only as targets of the analysis but also to differentiate between classes of objects. For example if you know that the difference between two types of cells (*e.g.* no septa vs complete septa) is the intensity of the pixels in the middle of the cell, then features regarding the neighbouring pixels and mean intensity of the pixels should be included in this list. Click "Select features".

The object feature selection window pops up with three types of feature groups. Select these:

All available features (three groups):



Object feature selection

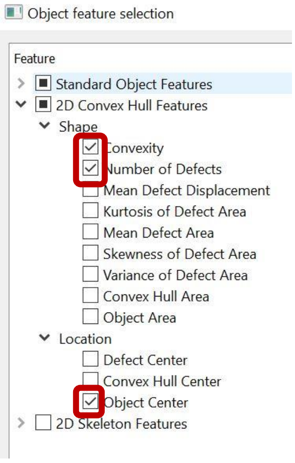
Feature

☐ Standard Object Features

☐ 2D Convex Hull Features

☐ 2D Skeleton Features

2D Convex Hull Features



Object feature selection

Feature

■ Standard Object Features

■ 2D Convex Hull Features

Shape

☑ Convexity

☑ Number of Defects

☐ Mean Defect Displacement

☐ Kurtosis of Defect Area

☐ Mean Defect Area

☐ Skewness of Defect Area

☐ Variance of Defect Area

☐ Convex Hull Area

☐ Object Area

Location

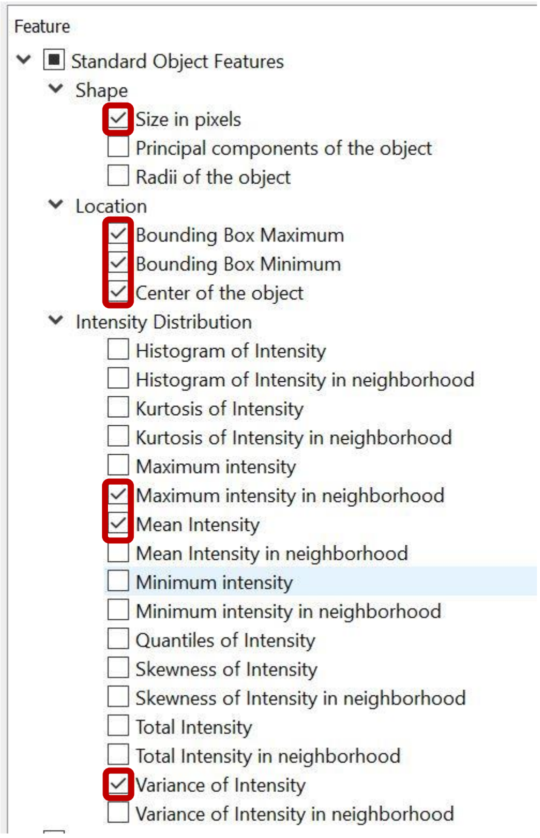
☐ Defect Center

☐ Convex Hull Center

☑ Object Center

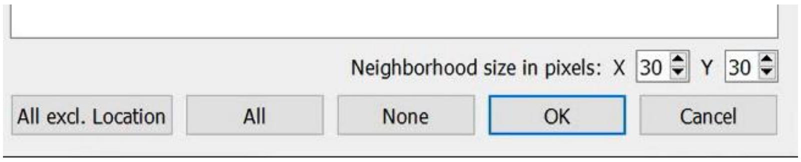
☐ 2D Skeleton Features

These values work when wanting to classify images of 2D fluorescence *cocci* shaped bacterial cells and when we want to measure the **cell volume** of the cells (therefore the Bounding Box features are selected, to be used later). However, if the experiment has different objectives (*e.g.* calculate the cell shape), change the type of feature you select at this stage. There is a summary of each feature at the bottom part of the pop-up window. For more information investigate the ilastik documentation here. Select the following values for the "Standard Object Feature" section. None of the "2D Skeleton features" are selected.



Feature
- ■ Standard Object Features
  - Shape
    - ☑ Size in pixels
    - ☐ Principal components of the object
    - ☐ Radii of the object
  - Location
    - ☑ Bounding Box Maximum
    - ☑ Bounding Box Minimum
    - ☑ Center of the object
  - Intensity Distribution
    - ☐ Histogram of Intensity
    - ☐ Histogram of Intensity in neighborhood
    - ☐ Kurtosis of Intensity
    - ☐ Kurtosis of Intensity in neighborhood
    - ☐ Maximum intensity
    - ☑ Maximum intensity in neighborhood
    - ☑ Mean Intensity
    - ☐ Mean Intensity in neighborhood
    - ☐ Minimum intensity
    - ☐ Minimum intensity in neighborhood
    - ☐ Quantiles of Intensity
    - ☐ Skewness of Intensity
    - ☐ Skewness of Intensity in neighborhood
    - ☐ Total Intensity
    - ☐ Total Intensity in neighborhood
    - ☑ Variance of Intensity
    - ☐ Variance of Intensity in neighborhood

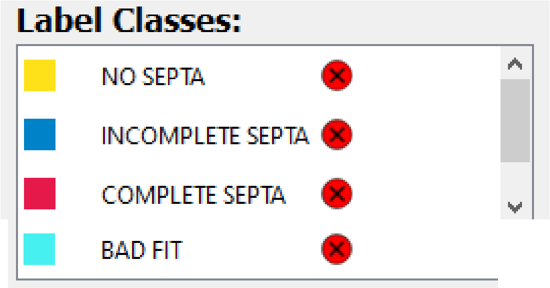
For the "Maximum intensity in the neighbourhood" and other features involving pixels surrounding the objects or inside the objects, there is an additional too choosing how many pixels in X and Y directions **define the "neighbourhood".** This will depend on the size of your objects and the image resolution. Start with 30 in each and try to move on to the next step. If the object classification prediction works poorly, try to come back and increase these numbers.



Neighborhood size in pixels: X 30 Y 30

- ○ **Object classification**

This section is finally where you can classify the cells according to different stages of the cell cycle. The three most common stages according to septa formation are: **No Septa, Incomplete Septa and Complete Septa**. The cells in the splitting stage that are already partially separated are considered as No septa cells rather than a different category because it is trickier to adjust the program to detect differences in these two situations. If you want to try, adjust the **"neighbourhood"** as seen above.

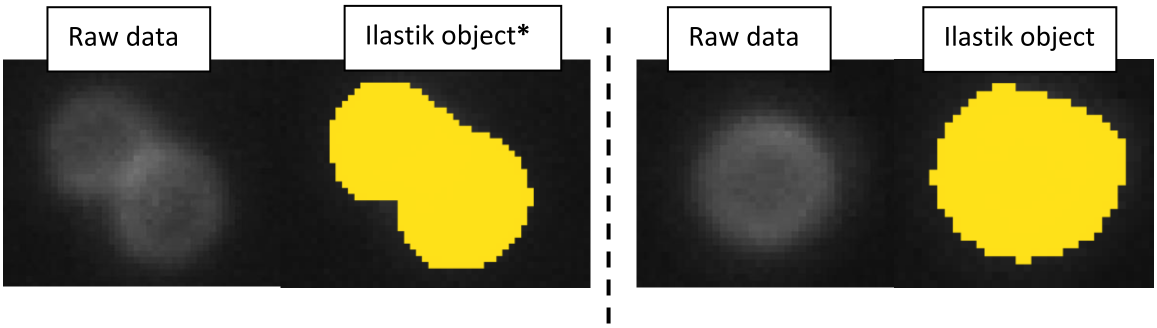


Label Classes:

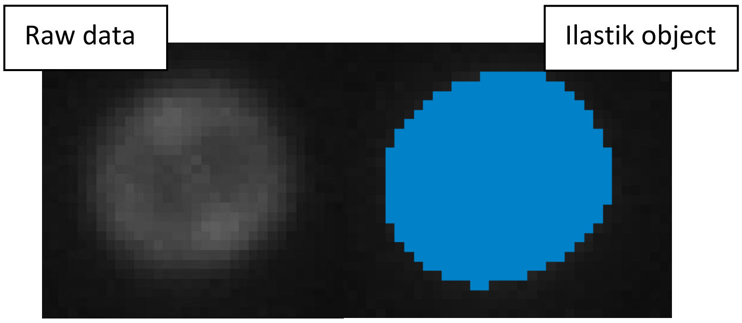
- 🟨 NO SEPTA ⊗
- 🟦 INCOMPLETE SEPTA ⊗
- 🟥 COMPLETE SEPTA ⊗
- 🟦 BAD FIT ⊗

Here are examples of each cell class according to their cell cycle:

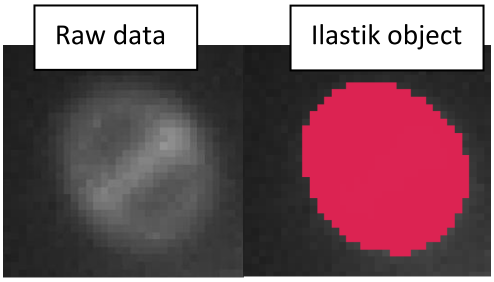
No Septa:



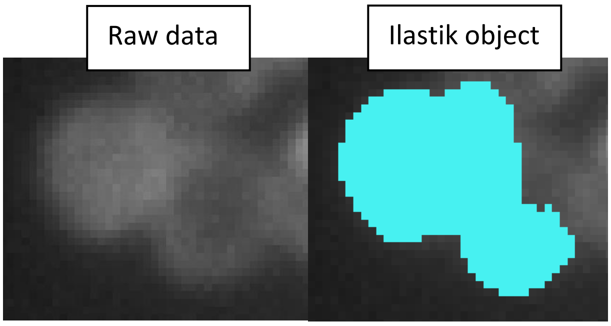
Incomplete Septa:



Complete Septa:



Bad fit:

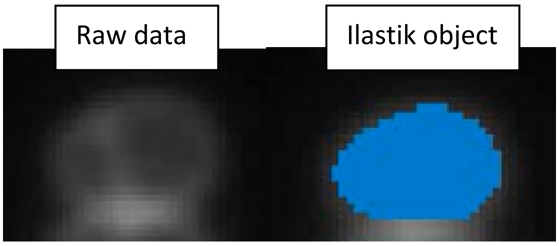


*Even thought in this case it looks like it has selected the two cells together, that is not true. The two cells are two individual objects, but because they have the same class, they have the same colour.

! IMPORTANT! The **BAD FIT** is a crucial fourth category because this is to remove the cases where the cells were not correctly masked, but they were wrongfully included after doing the **threshold**. However, this is almost impossible to avoid that some cells will not be correctly masked and their shape will be altered. If this was to be measured and included in the volume measurements, the final values would be wrong. Therefore, later in the routine in section 8, there is a way to filter these cells marked as BAD FIT and to not include them in the **final volume measurements**.

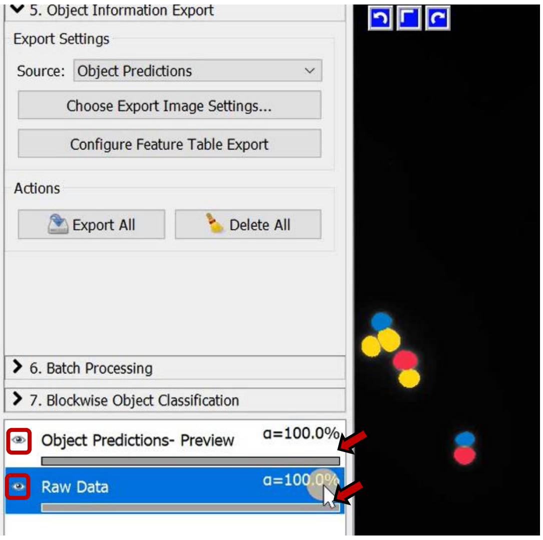
Alternative classes:

Sometimes depending on the sample, what you want to do is classify the cells according to "normal cells" or "cell shape defects", "septa defects". See one example of **cell shape defects (blue)** below.



- **Object Information Export**

This is about exporting the object classification data and the measurements (*e.g.* volume measurements) of the selected features that the program made based on your training. First, to double check that you are happy with your training section, click on the "Object Predictions – preview" eye icon on the left, and make sure the visibility is almost at 100%. Then, you should see your whole image with the raw data in the background and the classified cells with bright colours (yellow, red, blue and cyan) on top, see below.


5. Object Information Export
Export Settings
Source: Object Predictions
Choose Export Image Settings...
Configure Feature Table Export
Actions
Export All
Delete All
6. Batch Processing
7. Blockwise Object Classification
Object Predictions- Preview α=100.0%
Raw Data α=100.0%

Then, once you want to save the mask and finish, you need to export the mask which is named **"Object predictions"** in the Export Settings, under the "Source:" scroll down menu. Then click the button "Choose Export Image Settings…", a pop-up window will open named "Image Export Options".

! IMPORTANT! The output file format must be in **“hdf5”** which is the format from ilastik.

Find the folder where you want to save your file, I recommend the folder that was created beforehand called “Data for ilastik”, by clicking “Select”. The name of the file will be: “**YourImageName_ilastik_Probabilities.h5**”.

! IMPORTANT! For the future use of the batch processing tools to process all your images of your dataset, it is crucial that the folder to save your data is the correct one for this sample. Therefore, remember clicking the button called **“Reset filepath”**.

Then click “OK” and the file will be exported. At this stage another type of mask can be exported as well which is the **“Object Identities”** where each individual cell has a different colour, same as in the **Threshold** section. Repeat the same process as for the **“Object Predictions”** and save this image.

! WARNING! If you do the process above to save the two types of images (Object Identities and Object Predictions) in .hdf5 format, they won’t be useful to show in reports to highlight your data analysis process. To obtain a “.png” format of the images, it is much easier. Click on the **export button “Camera icon”** on the right top corner of the image and the “Image Export Options” window will appear. Select “png sequence” as File format, select where you want to save the image and manually give it an image name (under “file Pattern”), Click “OK” and then “Finish” and the pop-up window will close.

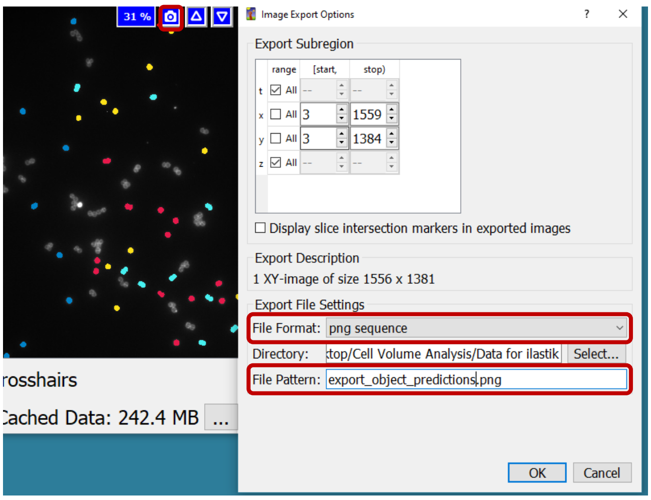


Image Export Options

Export Subregion

	range	[start,	stop)
t	☑ All	--	--
x	☐ All	3	1559
y	☐ All	3	1384
z	☑ All	--	--

☐ Display slice intersection markers in exported images

Export Description

1 XY-image of size 1556 x 1381

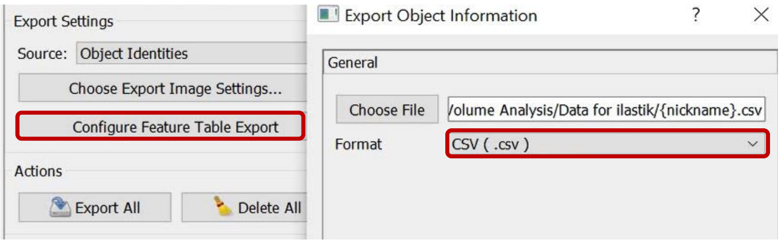
Export File Settings

File Format: png sequence

Directory: ktop/Cell Volume Analysis/Data for ilastik

File Pattern: export_object_predictions.png

Then the **measurements** need to be exported by Clicking the button “**Configure Feature Table Export**”. The “Export Object information” window will pop-up and the format of the file needs to be changed to **“.csv”.**


Export Settings

Source: Object Identities

Export Object Information

General

/olume Analysis/Data for ilastik/{nickname}.csv

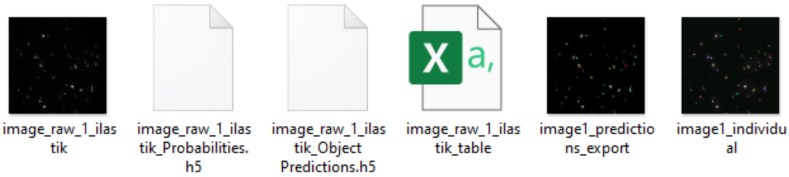
Format: CSV (.csv)

Choose Export Image Settings...
Configure Feature Table Export
Actions
Export All
Delete All
Choose File

Then Select the section **“Features”** inside the pop-up window and click **“All”.** This will export all the values of the features we have previously selected as they are the ones we want to measure.

Click “OK” and all the pop-up windows will be closed. Then finally go under the section called **Actions** and click the button **“Export All”. Save the project: *Project -> Save* and close it.**

After all this process this is how your “Data for ilastik” folder should look like this:

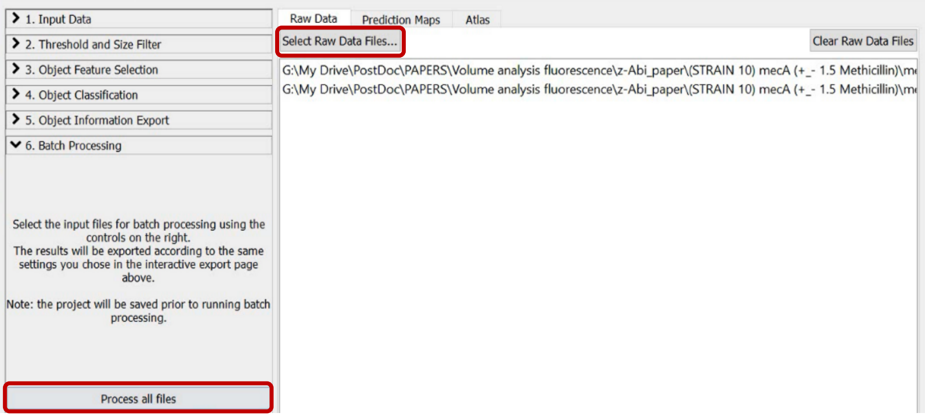

image_raw_1_ilastik
image_raw_1_ilastik_Probabilities.h5
image_raw_1_ilastik_Object Predictions.h5
image_raw_1_ilastik_table
image1_predictions_export
image1_individual

From left to right: “image_raw_1_ilastik.png” is the **Raw data** form the “training image”. Then “image_raw_1_ilastik_Probabilities.h5” is the **probability map** extracted from the pixel classification project. The file “image_raw_1_ilastik_Object_Predictions.h5” is the extracted image after the **object classification** in “hdf5” format. The table named “image_raw_1_ilastik_table.csv” is the **results table** containing all the information extracted from the measured features in the Object classification project. The last two files “image1_preictions_export.png” and “image1_individual.png” are the manually **exported images in .png** from the object predictions mask and the object identities mask respectively. These two “.png” files won’t appear when the batch processing is done for the rest of the images, but the previous 3 files aside from the raw data should appear for each individual image after performing section 7 for all the images.

7. Batch processing

As mentioned above, the sections 5-6 should only be performed for the **“training image”** of a dataset. Then, use the Batch Processing option for the rest of the images of the dataset.

First open the saved **ilastik_pixel project**, go directly to section 6 named “Batch Processing” and click on the button under the Raw data section named **“Select Raw Data Files…”,** you can select as many images as you need as long as they are the same type of sample as the “training image” you used to train this project. Then, click “Process all files”. This can take up to a few minutes, wait for all the green bars at the bottom of the page to finish loading.

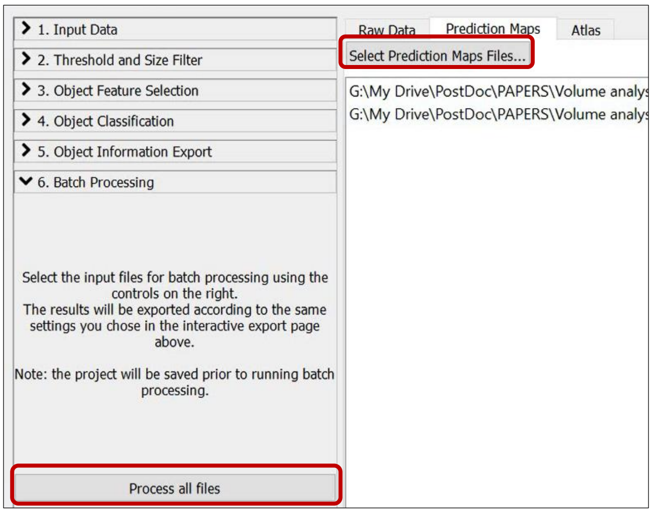


G:\My Drive\PostDoc\PAPERS\Volume analysis fluorescence\z-Abi_paper\(STRAIN 10) mecA (+_- 1.5 Methicillin)\m
G:\My Drive\PostDoc\PAPERS\Volume analysis fluorescence\z-Abi_paper\(STRAIN 10) mecA (+_- 1.5 Methicillin)\m

Select the input files for batch processing using the controls on the right.
The results will be exported according to the same settings you chose in the interactive export page above.

Note: the project will be saved prior to running batch processing.

This process will create the **probability maps** for all the selected images. The files should be named “image_name_Probabilities.h5”. Then, save the project and close it. Open the **ilastik_object project** and repeat the same process, first finding the raw data, then go to the section Prediction Maps and click “Select Prediction Maps Files…”, see below:


1. Input Data
2. Threshold and Size Filter
3. Object Feature Selection
4. Object Classification
5. Object Information Export
6. Batch Processing
Raw Data
Prediction Maps
Atlas
Select Prediction Maps Files...
Process all files

G:\My Drive\PostDoc\PAPERS\Volume analys
G:\My Drive\PostDoc\PAPERS\Volume analys

Select the input files for batch processing using the controls on the right.
The results will be exported according to the same settings you chose in the interactive export page above.

Note: the project will be saved prior to running batch processing.

This will produce two additional files: the **object classification** file in ".hdf5" format and the **results table** in ".csv" format. If you open the table, you will see the type of data that the program produces. All the measurements are in pixels. The results need to be process further to be converted into µm and to calculate the Volume from the "Bounding box" coordinates.

! WARNING! The prediction maps in the batch processing step need to be selected in the same order as the raw data has been selected, or else the resultant files will not make sense.

The cell classification and measurements for all the images are finished. Move on to section 8 to obtain clear and user friendly result tables containing the cell number, the classification label, the minor and major diameter of the cell and the Volume calculated assuming the cells are prolate spheroids.

8. <u>Volume measurement from table.</u>

Needs finishing (but no need for the 3D application of this routine to obtain the cell shape).