University of Exeter

**Ilastik Mother Machine Python**

****

Optimization Guide & Protocol for ilastik

Mother Machine Python

College of Engineering, Mathematics and Physical Sciences

Project Supervisor: Dr Stefano Pagliara

Study Programme: Natural Sciences

9th December 2020

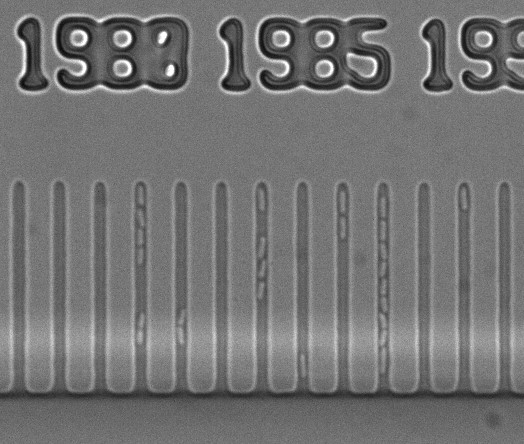
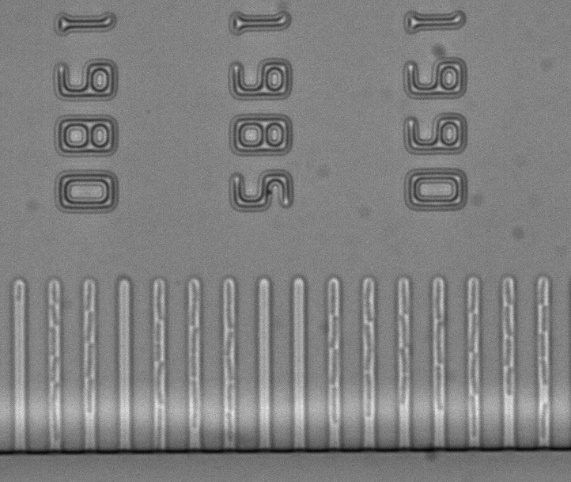
**Section 1 – Mother Machine Imaging Optimisation:**

Image quality will play a large role in not just making the upcoming steps easier and quicker but improving the quality of the data. Here are some points to look out for when gathering or filtering out images to analyse via ilastik and iMMPY:

**~Brightfield Contrast:**

1. Aim to ensure a similar level of contrast throughout as many of your images as possible. Subtle variations are fine. However, any distinct variations in contrast between consecutive frames or portions of the time lapse will require more care and extra training to be undertaken during the pixel classification stage to compensate for changes.
2. The default colour for the bacteria and channel can be either white and grey, or grey and white respectively, both will work equally well providing it remains consistent across all images.

See examples below:

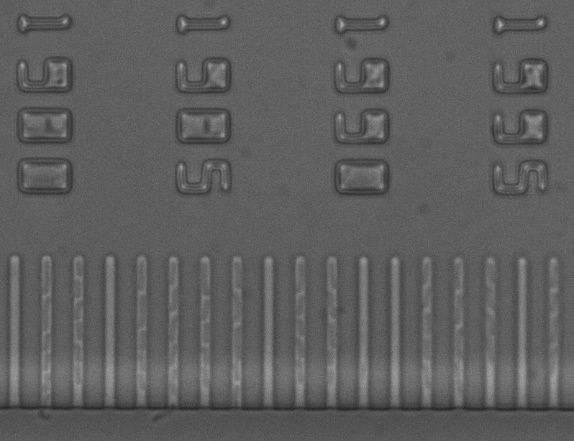
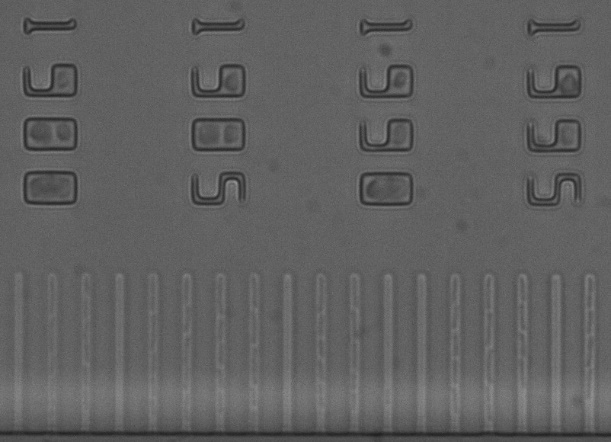
**** 

**Section 2 – Training machine learning via ilastik:**

Make sure that you have downloaded ilastik <https://www.ilastik.org/> and a python interpreter of your choice (anaconda and its included Spyder interpreter are recommended: https://www.anaconda.com/products/individual).

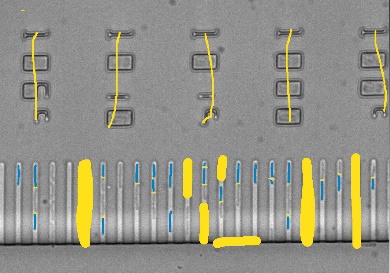
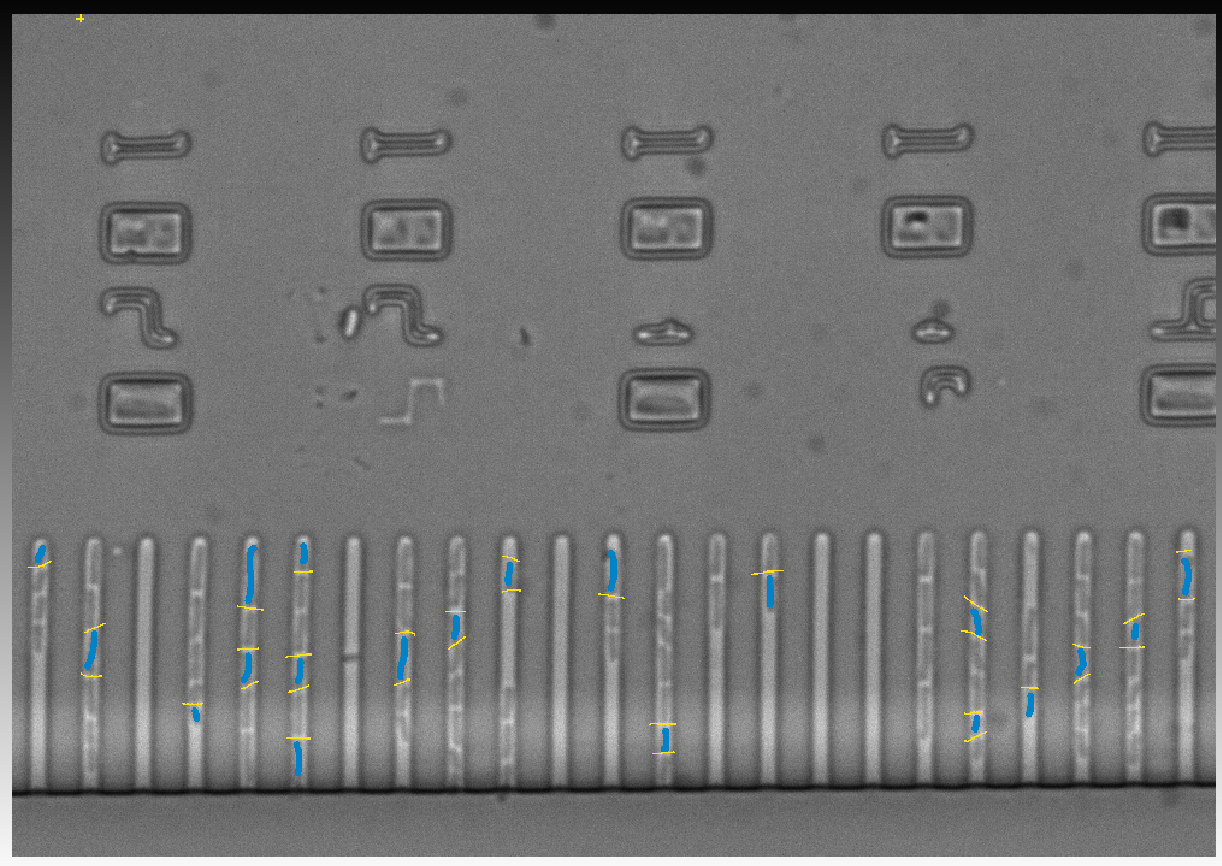
**~Preparation:**

1. Remove all time lapse images that aren’t in a brightfield-fluorescence pair (e.g. any initial brightfield-only images taken during calibration).
2. Separate the brightfield and fluorescence images into different folders (thanks to step 1, both folders should have the same number of images). Ensure that they are named such that they show up in the same order (1st brightfield image is paired with 1st fluorescence image, 2nd with 2nd etc.).
3. View the brightfield images using ImageJ by going to: file, import, image sequence and selecting all images in the brightfield folder.
4. Look through these images and check for any sudden variations in contrast or loss of focus. If there are any distinct changes, note down the frame numbers and/or time codes for each image. Remove these images and their corresponding fluorescence images.
5. The images below represent such examples of a sudden change in contrast or loss of focus:

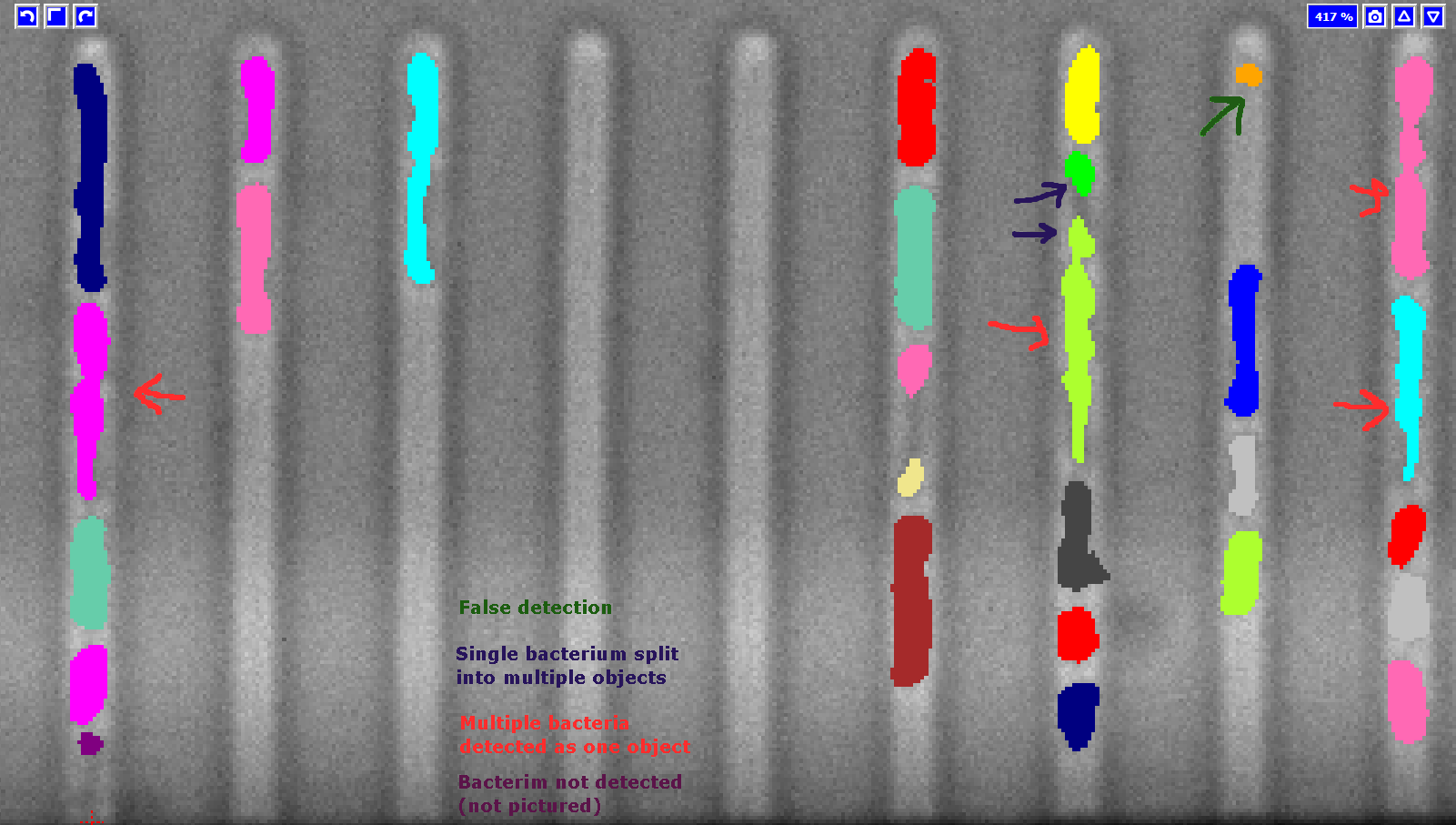
1. While these examples may still be usable, they will require a large amount of special consideration during the later training stages and can reduce the overall quality of tracking - thus, it is advisable that they are removed during preparation.

**~ilastik – Pixel classification:**

1. Start a new pixel classification project.
2. Import images:
   1. “Add a single volume from a sequence”
   2. “Select files”
   3. Select every brightfield image
3. In “Feature Selection”, select every feature.
4. In “Training”, work through the image set: Ideally the first frame, some frames scattered mid-way through the data set e.g a couple frames around 2/5th 3/5ths and 4/5ths of the way through, and the final few frames. Additionally, follow the below steps to give special attention to any frames showing variations in contrast or frames where any abnormalities may appear.
5. If this is your first frame – or a frame requiring special attention – use the “background” colour to cover the background making sure to hit all of the points of interest (e.g. numbers, empty columns, the central channel).
6. In a different colour, carefully draw over some of your bacteria with the “bacteria” colour [suggestion: size 5].
7. Draw a thin line separating the bacterium and its neighbours with your “background” colour (this helps the program with fine separation [suggestion: size 3]). Repeat step 4 a-c until the desired number of bacteria have been labelled. As a suggestion, try to get at least 100 total bacteria. An example of this is shown below:
8. Note not to worry about discriminating between bacteria that you want to and don’t want to measure, unless they have very different shapes/sizes. At this point we simply want to train the model to best detect any bacteria.
9. You should end up with a portion of your frames looking similar to below:
10. Save the project! This is important as the next step can crash if you’ve drawn too much.
11. Press “live update” to see the program’s current segmentation model.
    1. Be patient, but if you’ve left the computer alone for a while and the program has been stuck at 80% for a long time, you may need to close and reopen ilastik and use the eraser tool to remove a bit (or remove some of the options in feature selection)
    2. Scroll through the frames and check that the prediction quality seems generally reasonable (doesn’t need to be perfect). If not, label more bacteria.
    3. Note that if some of your images are different from the rest for whatever reason (changed contrast, momentary obstruction of the microscope…), that image will likely be given special care. You may need to label more bacteria on that frame specifically (although this could muddy the model as a whole so be careful), or cut it out of the dataset.
    4. If high levels of bacteria prediction (blue overlay) occur around the edges of the growth channel (usually at the top or around glare regions – see the bright streak from left to right in 4.e) draw a U shaped line in the background colour around the edges of the growth channel to help define the bacteria from the channel edge.
12. In “Prediction Export”, “choose export image settings”. There, change the export format to “multipage tiff”, set the output file name and location, then return to the main ilastik screen and “Export”.

**~ilastik – Tracking with Learning:**

1. Start a new “Tracking with learning” project (with pixel prediction map)
2. Import images the same way
3. Import the pixel classification export from the previous project (In “input data”, switch tabs to “prediction maps” and “add new separate image(s)”).
4. In “Threshold and Size Filter”, change the thresholds so that the bacteria look as well defined as possible:
5. Input: change to the colour of the bacteria’s label.
6. Smooth: changes how much artificial smoothing is applied to the bacteria [suggestion: 1; 1.5]
7. Threshold: How certain the model must be before labelling a node as part of the bacterium [suggestion: 0.6. This means that only pixels that the model is 60%+ sure belong to bacteria will be labelled as such. Higher numbers lead to smaller bacteria and sometimes causes them to split into two, but also leads to better separation between those that are almost touching. Lower will do the opposite.]
8. Size filter: The maximum and minimum size of what the model will label a bacterium [suggestion: min 30 max 2000. Only bacteria that cover more than 30 pixels or less than 2000 will be counted]
9. Adjusting these will affect the model in real time, so manually play around with these numbers to get the most accurate model (clicking the eye next to “final output” will show the underlying images).
   1. Here are some problems to look out for:



1. Use “live update” to check across other frames that the prediction seems mostly accurate. Add more manual labels to problem areas if needed to give the model additional training.
2. In “Object count classification”, go through a couple of frames and label objects that ilastik has detected. Pressing “add label” will create new colours for you to label outlines that contain two bacteria or more.
3. Use “live update” to check across other frames that the prediction seems mostly accurate. Add more manual labels to problem areas if needed to give the model additional training.
4. In “Tracking”, you can leave everything as default and press the big “Track!” button (note if you’ve forgotten a step, you’ll be prompted here).
5. In “Tracking Result Export”, change the export format to “multipage tiff”, set the output file name and location, and export.
6. If an error occurs e.g. ‘dtype does not have matching OME-XML type’ go into in the image export options, select “convert to Data type” then select ‘unsigned 16-bit’ before exporting.

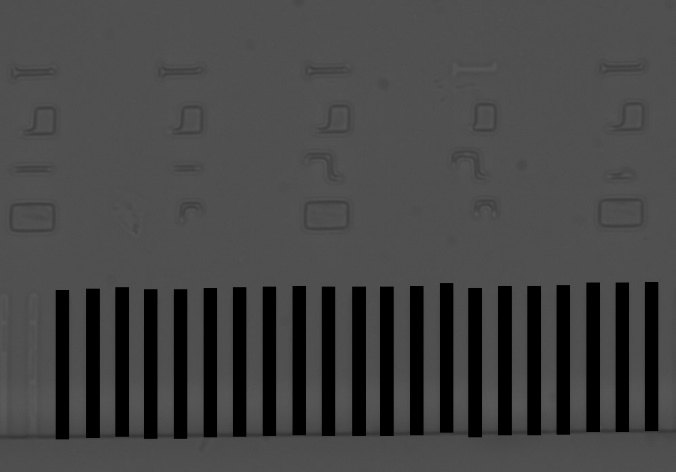
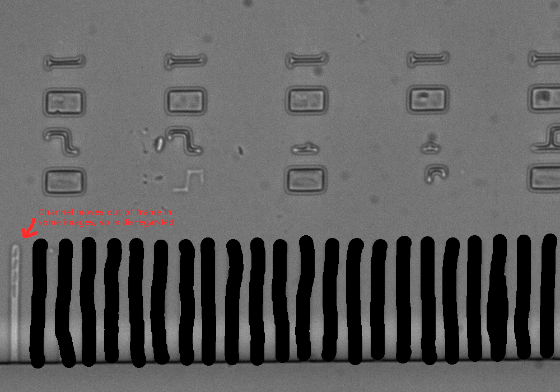
Collating the images with ImageJ/Fiji:

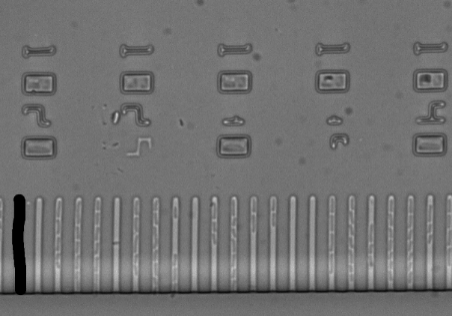
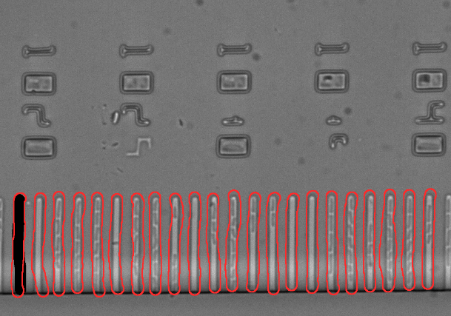
1. File → Import → Image sequence. In the popup window, select one picture in the brightfield images folder, and the program will open up a dialogue box which will import the others.
2. At this step, look through your images and make a note of which channels get cut off/fall of the side of some images, in the next step, you’ll only want to label the channels that are consistently present in the whole time lapse
3. Save as → Tiff.
4. Repeat for fluorescence images.

**Section 3 – Making the channel masks:**

Use an imaging software of your choice to make the masks. Here I will detail the process for using the free software Gimp:

1. Open the stack of brightfield images created above (Open → select image stack → import).
2. Click on the first layer (probably automatically labelled “background”) and a new layer (the defaults options are fine).
3. Repeat this step until each layer has a corresponding new empty transparent layer, making sure to name them alphabetically (Ctrl-Shit-N makes a new layer).
4. Look through the images using ImageJ and identify which frames stay in frame for the full duration of the time lapse, these are the channels we wish to create masks over.
5. Now is the time to discriminate between desired and undesired bacteria: Make sure to only label channels that are present throughout the entire experiment and ignore any that touch the edge / get cut off in a later image / are undesirable for any other reasons [but do leave in empty channels unless they are undesirable for other reasons]. The code will only analyse bacteria that are contained within the channels that are labelled here.
   1. Pick the top transparent layer and draw a box around the first channel that remains in the frame using the rectangle select tool. Aim to make this box slightly larger than the growth channel.
   2. Fill it in black by selecting the bucket fill tool and clicking inside the rectangle.
   3. From here, select the move tool and click on the black rectangle, use ctrl + C to copy the and ctrl + V to paste the rectangle, drag and drop it over the next growth channel. Continue pasting until the final growth channel present for the duration of the time lapse.
   4. Don’t worry about being too precise. However, make sure the rectangles do not intersect, and try to cover the whole channel (small slips are fine).
   5. We can additionally create masks using the paintbrush tool as shown below in the left image, however this method generally takes longer to complete.



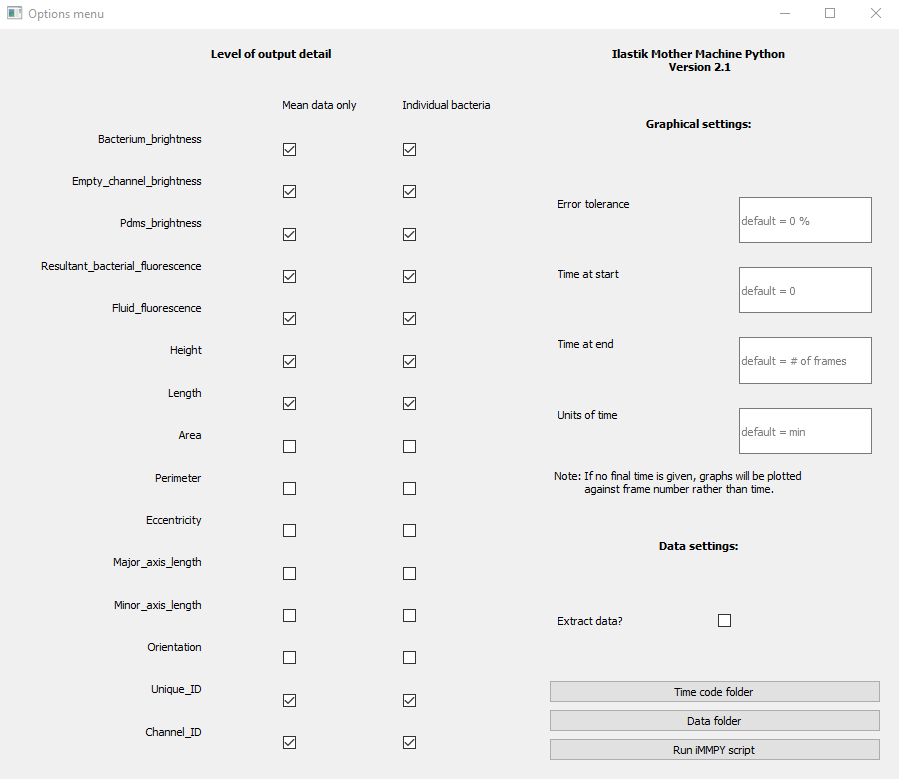
1. Once you have created a mask for all desired channels, click the green anchor under the layers tab in the bottom right corner to form one mask layer.
2. Copy this mask layer and paste it on the transparent layer below it, uncheck the eye symbol to toggle the visibility of the two layers new above. Drag the black boxes (note that all of them should now move together) and align them to the next frame. Toggle the visibility of the mask layer to see just the outline of the boxes if you wish to better align the mask over the channels, when aligned click the anchor and repeat this process for all frames. Note that some frames will drift more than others.
3. Once all frames have a complementary mask layer, you may notice that the yellow dashed box around each mask layer doesn’t align with the image boundaries. Next, we want to align these masks to the image boundary, this can be done by selecting the mask layer, clicking on Layer in the tool bar at the top, then clicking Layer to Image size. To make this process easier I suggest using the ‘ofn-layers-to-image-size’ extension which can be found here: https://sourceforge.net/projects/gimp-tools/files/scripts/?fbclid=IwAR1rKZM6btQltMdzWVmiCScBIWU\_vmDpqL6Wi4tLmDfi9vaIiA\_J503pNso
4. Repeat this step for all mask layers.
5. Alternative method: This feature may work, but often leads to an error during stage 4:
   1. Click on the eye next to the two previous layers, this will make them disappear.
   2. Use the mouse or arrow keys to select the next empty transparent layer.
   3. Either create full masks for each layer as in step 4
   4. Or only draw the leftmost channel for each frame. The code will duplicate the first frame’s channel mask onto other layers, using that one drawn-in channel as a guide (an example of this is shown here in red).
6. Once each frame’s channels have been labelled, export each of the transparent layers. I suggest using this extension to make the process easier: <https://github.com/khalim19/gimp-plugin-export-layers>
7. Go into the folder containing the masks export and delete the brightfield images.
8. Import the masks into ImageJ, change the file type from ’RBG’ to ‘16bit’ by going into ‘type’ in the ImageJ options bar and clicking on 16bit. Save as a .tiff as done in the previous section.

**Section 4 – Running iMMPY:**

1. Make sure these files are all in the same folder, and rename them as such:

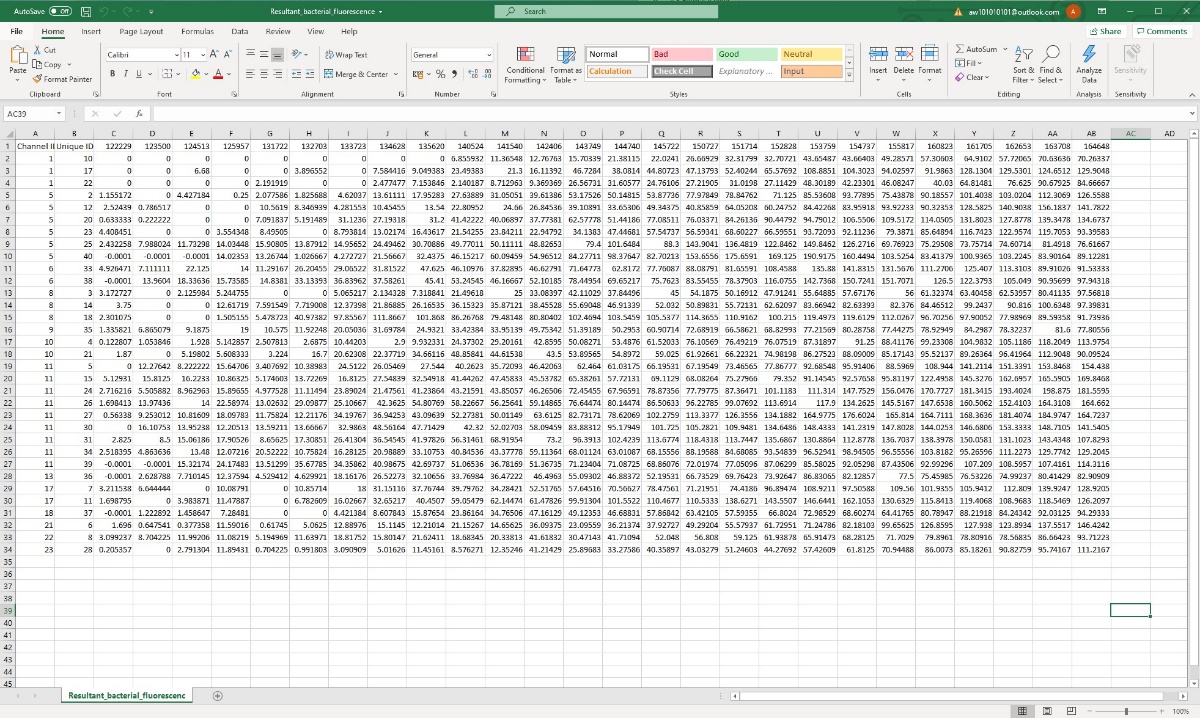
* Stack of brightfield images: [something]\_Brightf
* Stack of corresponding fluorescence images: [something]\_Fluo
* Stack of manually drawn channels: [something]\_Channels
* Stack of ilastik-created masks: [something]\_Ilastik

1. At this point you should be able to run the code. It’ll pull up an options window in which will allow you to change a couple of settings, as well as set the location of the folder in which the files discussed above are stored. Additionally, a time code folder needs to be selected - this is the folder containing either the brightfield or the fluorescent images.

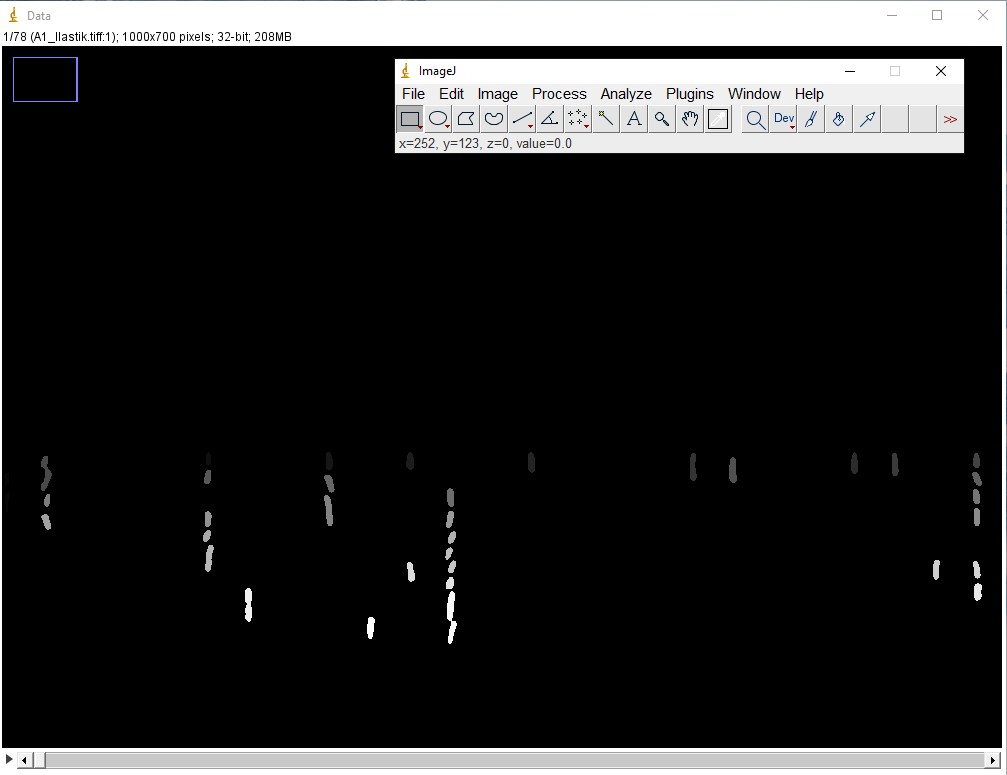
****

**Section 5 – Interpreting the output files:**

The main export files (Bacterium brightness, Resultant bacterial fluorescence, Height, Length, Area, Perimeter, Eccentricity, Major/minor axis length and Orientation) will be exported with the following format:



Here the “Channel ID” column corresponds to the channel number - as determined by your manually drawn mask layer from left (channel 1) to right (channel 23 for this example set). The “Unique ID” column corresponds to a unique identifier value given to each bacterium. This unique ID can be checked for a given bacterium by opening the ilastik-generated mask using ImageJ and looking at “value = “ option in the ImageJ bar while hovering over a bacterium as shown below:



Additional context: Value = 0 corresponds to the unique identifier for the background.