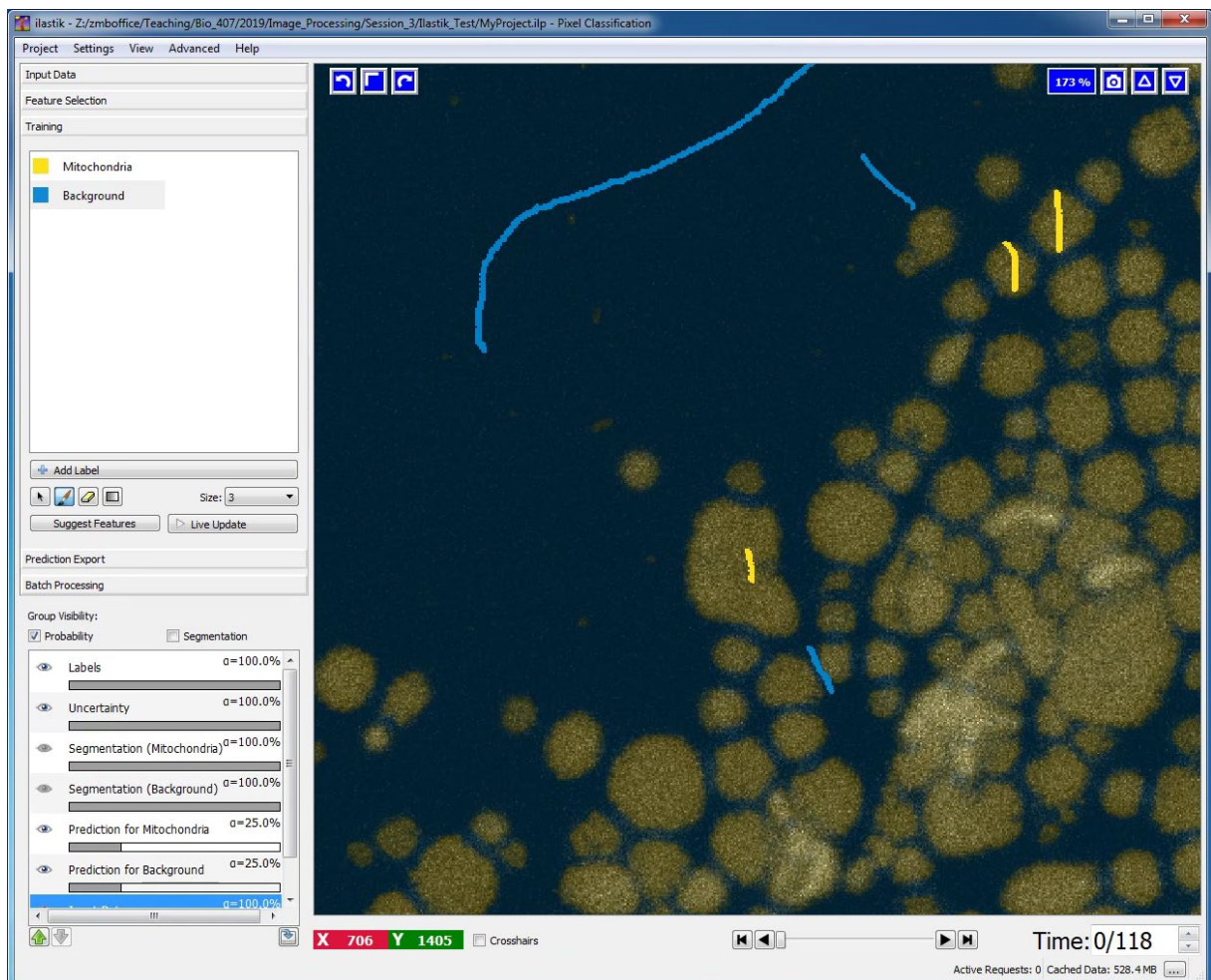


Pixel classification and segmentation using Ilastik

Practical Examples



Center for Microscopy and Image Analysis
University of Zurich

This script was written as a short summary for the users of Center for Microscopy and Image Analysis, University of Zurich, Switzerland

April 2019, Dominik Hänni and Joana Delgado Martins

Some explanations were taken from:

Haubold C., Schiegg M., Kreshuk A., Berg S., Koethe U., Hamprecht F.A. (2016) Segmenting and Tracking Multiple Dividing Targets Using *ilastik*. In: De Vos W., Munck S., Timmermans JP. (eds) Focus on Bio-Image Informatics. Advances in Anatomy, Embryology and Cell Biology, vol 219. Springer

This script was written using Ilastik 1.3.2rc2.

Contents

1. ABBREVIATIONS	5
2. REMARKS	5
1. SOFTWARE: ILASTIK	5
2. OPEN ILASTIK AND LOAD DATA	6
3. NAVIGATE ILASTIK	8
4. SELECT AND EXPLORE FEATURES	9
5. TRAIN THE CLASSIFIER	9
6. LAYERS AND VISUALIZATION	12
7. EXPORT RESULTS	12
8. OPEN AND WORK WITH THE PROBABILITIES IN FIJI	13
9. LITERATURE AND FURTHER INFORMATION	14

1. Abbreviations

Stack: Data structure with a set of related images of the same size and bit depth.

Slices: The images that make up a stack.

Voxel: In stacks, a pixel (which represents the smallest 2D image data) becomes a voxel (volumetric pixel), i.e. an intensity value on a regular grid in a three dimensional space.

Classification: In machine learning and statistics, classification is the problem of identifying to which of a set of categories (sub-populations) a new observation belongs, on the basis of a training set of data containing observations (or instances) whose category membership is known. Classification is an example of pattern recognition. (Wikipedia)

2. Remarks

This script was written for practical training purposes by the Center for Microscopy and Image Analysis, University of Zurich. Theory is kept to a minimum.

The script should be used as a guideline for training.

Highlighting styles used in this script:

File names are set in green (e.g. `Cell_Division_Pinhole_0_5AU_Nyquist_7386.tif`)

Step 1.

Practical steps are set in italic.

Keyboard shortcuts here

1. Software: Ilastik



ilastik

the interactive learning and segmentation
toolkit

Ilastik is a simple, user-friendly free open source tool for interactive image classification, segmentation and analysis. It is built as a modular software framework, which currently has workflows for automated (supervised) pixel- and object-level classification, automated and semi-automated object tracking, semi-automated segmentation and object counting without detection. Most analysis operations are performed lazily, which enables targeted interactive processing of data subvolumes, followed by complete volume analysis in offline batch mode. Using it requires no experience in image processing.

Ilastik learns from labels provided by the user through a convenient mouse interface. Based on these labels, Ilastik infers a problem specific segmentation. A random forest classifier is used in the learning step, in which each pixel's neighborhood is characterized by a set of generic (nonlinear) features. Ilastik supports up to three spatial plus one spectral dimension and makes use of all dimensions in the feature calculation.

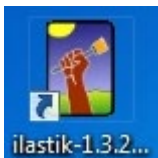
Ilastik provides realtime feedback that enables the user to interactively refine the segmentation result and hence further fine-tune the classifier. An uncertainty measure guides the user to ambiguous regions in the images. Real time performance is achieved by multi-threading which fully exploits the capabilities of modern multi-core machines. Once a classifier has been trained on a set of representative images, it can be exported and used to automatically process a very large number of images (e.g. using the CellProfiler pipeline).

Ilastik was first released in 2011 by scientists at the Heidelberg Collaboratory for Image Processing (HCI), University of Heidelberg.

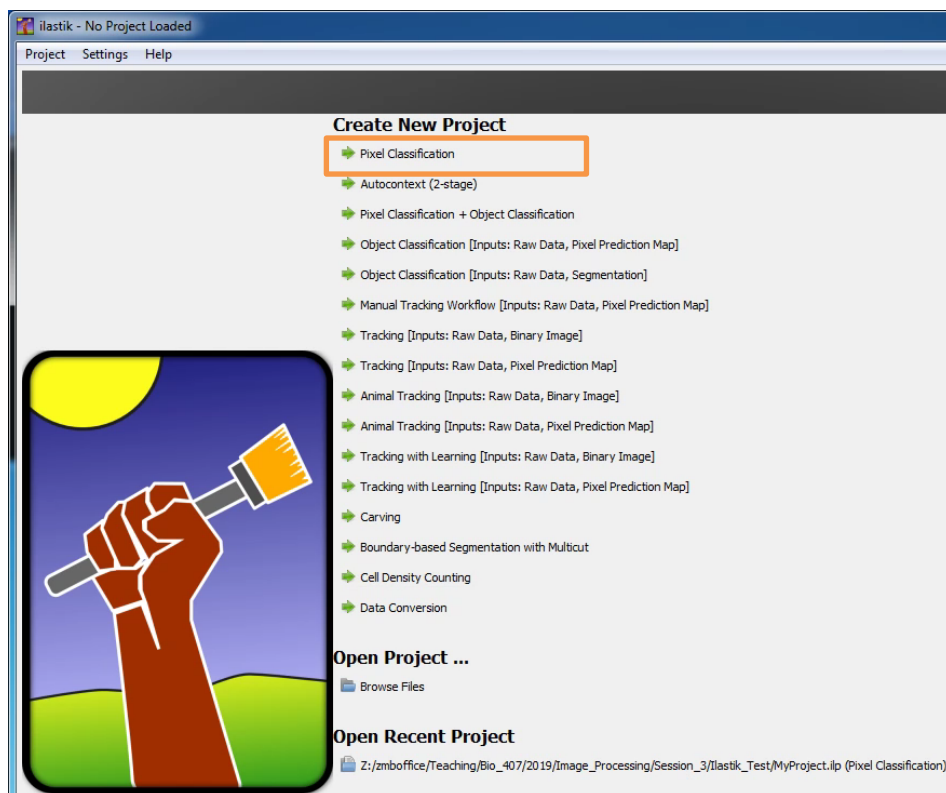
(Adapted from www.ilastik.org and “Sommer C, Straehle C, Kothe U, Hamprecht FA: Ilastik: Interactive learning and segmentation toolkit. In: *Proceedings - International Symposium on Biomedical Imaging*, pp 230–233, 2011”)

2. Open Ilastik and Load Data

Install and open Ilastik:



Under “Create New Project” choose “Pixel Classification”



Save your Ilastik project in a folder of your choice.

If the data of interest consists of only one frame or is stored as a HDF5 volume in a single file

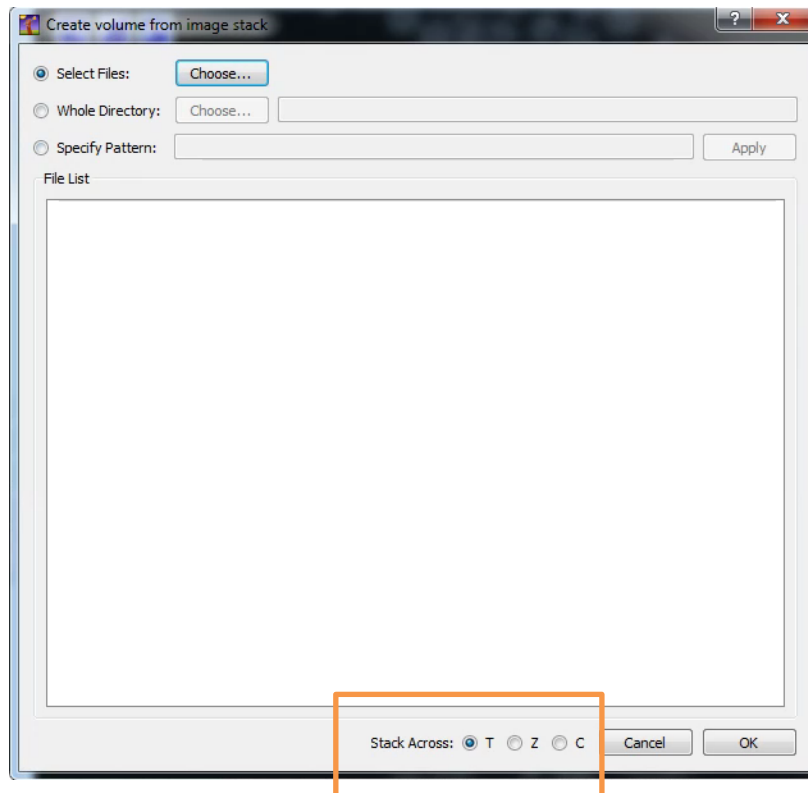
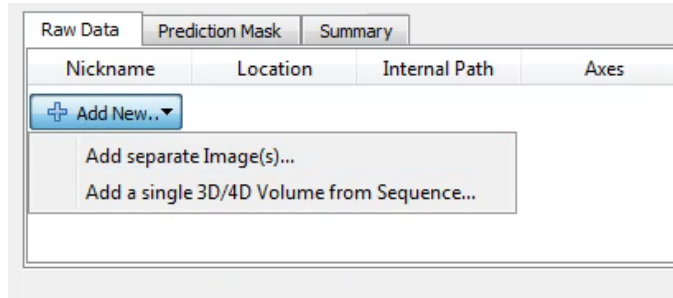
“Add separate image(s)…” should be selected,

but in the case of a stack with one file per time step or z-slice the option

“Add a single 3D/4D Volume from Sequence...” should be used.

Step 1.

Add New>Add a single 3D/4D Volume from Sequence...



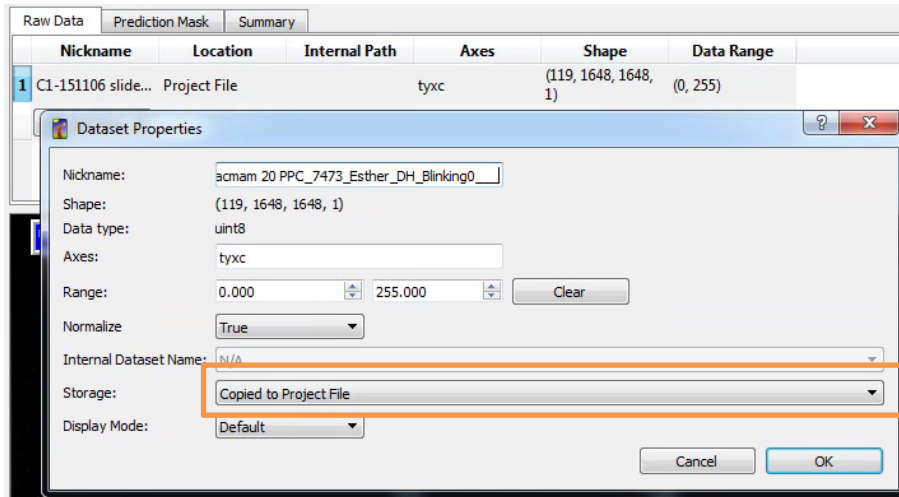
Step 2.

Select Files>Choose...

Select all files either from the demo data folder or your own data by holding down “Shift”

Also define the dimensionality by “Stack Across”

If you press OK Ilastik should load and visualize the data. If something such as the dimensions is wrong in the “Raw data” field, you can double click on the position and modify it.

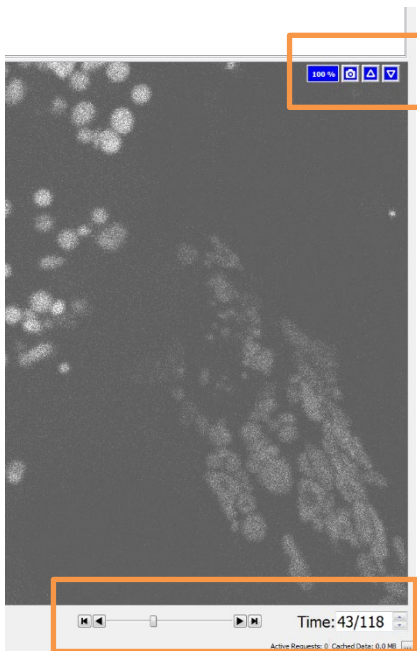


Loading a volume from a HDF5 file is faster than loading a tiff stack because HDF5 allows hierarchical blockwise access. If Ilastik runs very slowly, try converting the data into HDF5. The easiest way to do this is to select “Copied to Project File” for the Storage option in the dataset properties window.

3. Navigate Ilastik

Depending on whether a 2D+t or 3D+t dataset was loaded, Ilastik will display one timeframe as 2D image, or a split view with xy, yz, and xz slicing planes and a 3D window indicating the relative position of those slicing planes. In a 3D+t dataset Ilastik also shows a position marker that indicates which slice is shown in the other planes (note that they have differently colored GUI elements, this color corresponds to the color of the position marker).

The position marker can be disabled by a checkbox in the lower right. Double clicking inside one of the slicing views centers the other views on that position in the data.



The best user experience is achieved when using a mouse with scroll wheel. To zoom in and out of any of the views use **Ctrl + mouse wheel scrolling** (or **cmd + mouse wheel scrolling** on a Mac). Panning can be achieved by **holding the mouse wheel pressed + move**, or **dragging the mouse while pressing the shift-key**.

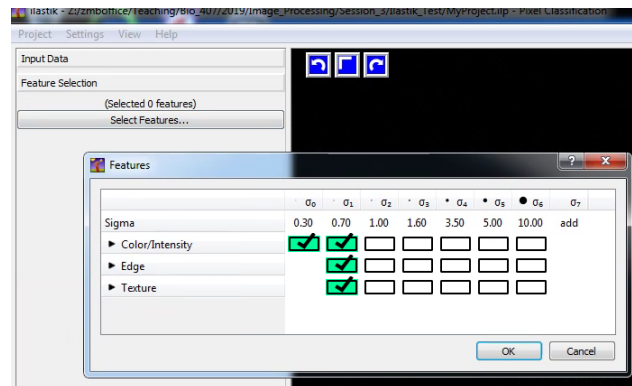
You can zoom in or out by using the keyboard **keys + and -** respectively. You can also use the arrows on the top right for zooming to fit or resetting the zoom.

To navigate through time, press **Shift + mouse wheel scroll** or use the time spin box in the lower right. Note how the position gets updated in the upper left of each view, and the timeframe in the lower right.

4. Select and Explore Features

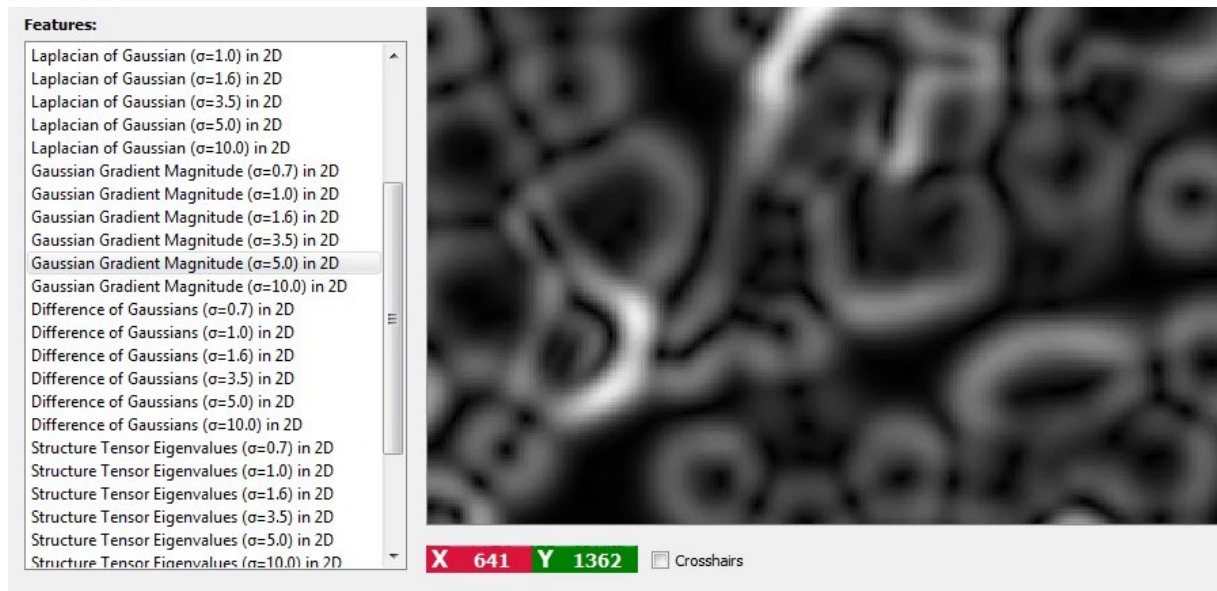
Step 3.

Next go to “Feature Selection” in the left pane and “Select Features”



Click into the boxes to select features. As a starting point you might choose all features at all given scales. However depending on your data and segmentation needs, you might need to consider the “Feature Selection” as a part of your optimization process. Also you can add custom features with a chosen scale here.

Next, take a look at the calculated features. Check how the parameters space for the multidimensional classification were augmented. Maybe you can even localize some features which might not very helpful and therefore can be removed from your feature selection. Try to remove the ones that look unnecessary but make sure that the prediction remains good. This could also be achieved by iteratively adding or removing a single feature from the list and assessing the quality of the predictions in the Live update mode during the following training step.



5. Train the Classifier

Next the classifier must be trained.

A classifier learns how to discriminate between different classes from a set of examples. For pixel classification the classes a pixel can belong to are often foreground and background, but could also

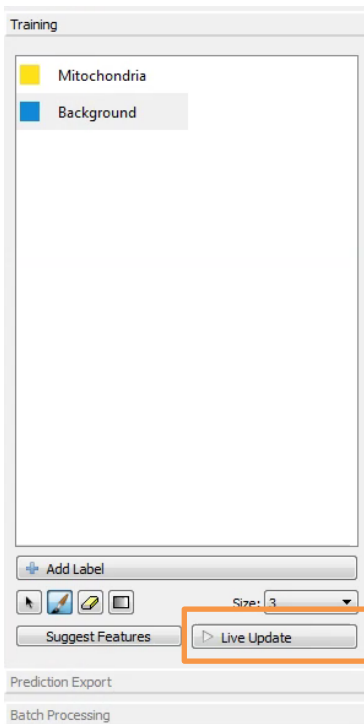
represent e.g. different tissue types. For the discrimination, the classifier considers the selected features of each pixel.

While the classifier is trained, it learns how to combine the selected features such that it predicts the desired class for the training examples as good as possible. The more informative the set of features is, the better the classifier can predict the correct classes. The classifier used in Ilastik is a **Random Forest**, an ensemble of decision trees. Each bifurcation of a decision tree represents a binary decision based on a feature, for example: is the smoothed intensity value of this pixel greater than some threshold? Each leaf of the tree is assigned the class of all pixels that it contains during training. For a new pixel p , the decision tree assigns p the class of the leaf it falls into. As a Random Forest consists of an ensemble of trees with randomly selected features at the junctions, it allows averaging this decision over all trees to yield a probability estimate per class.

Step 4.

Go to "Training" on the left pane and rename the two labels/classes by clicking on Label 1 to "Mitochondria" and on Label 2 and name it "Background".

If you have a more complex situation, you also might want to introduce additional labels/classes. This might improve your results quite dramatically.



Now select your first class "Mitochondria", chose a suitable brush "Size" and paint on some mitochondria. Next go to the next class and do the same. If you want to know how the classifier evolves and also dynamically direct the evolution, you can

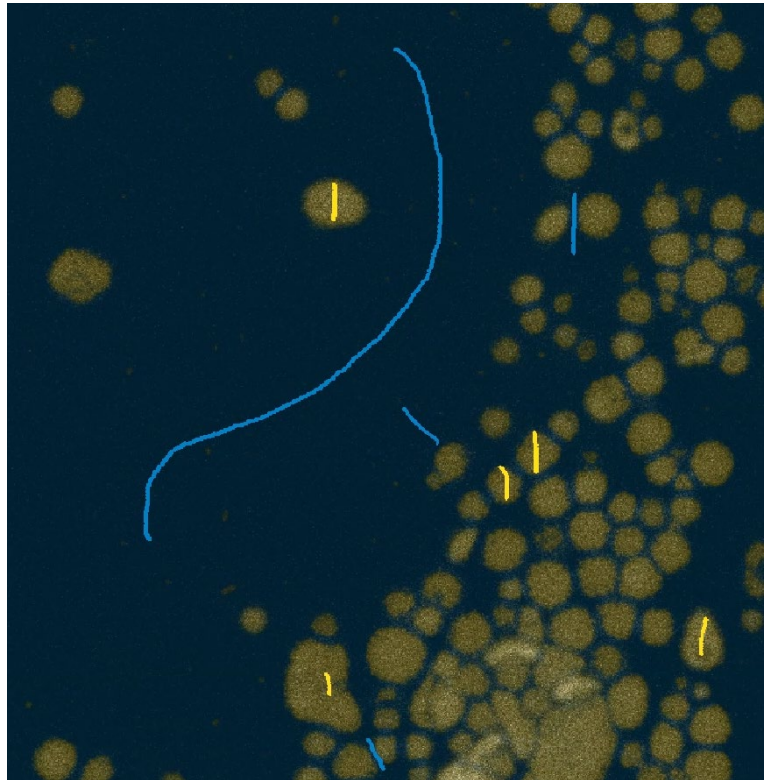
Step 5.

Activate "Live Update" Together with Group Visibility>Probability

Usually a few strokes are enough to get a good classification. You can also put some strokes in the touching region of mitochondria in order to split them.

In machine learning there is always a tradeoff between a perfect fit to the training data and good predictions on unseen data. The latter, also called generalization, usually works better if the algorithm has not been trained too specifically, which is known as overfitting. Transferring this to the labeling step means: make sure not to give too many examples, which all look the same.

Experienced users would start with a few general labels such as a thin stroke in the background, and by marking a few pixels of one or two cells. When enabling live update, the prediction should already



look sensible, but probably the boundary between objects is not properly preserved. To fix this, choose a very thin background brush, paint a line between two closely neighboring targets, and inspect the results. Leaving live update enabled, you can now navigate through the image and add labels where the prediction is wrong.



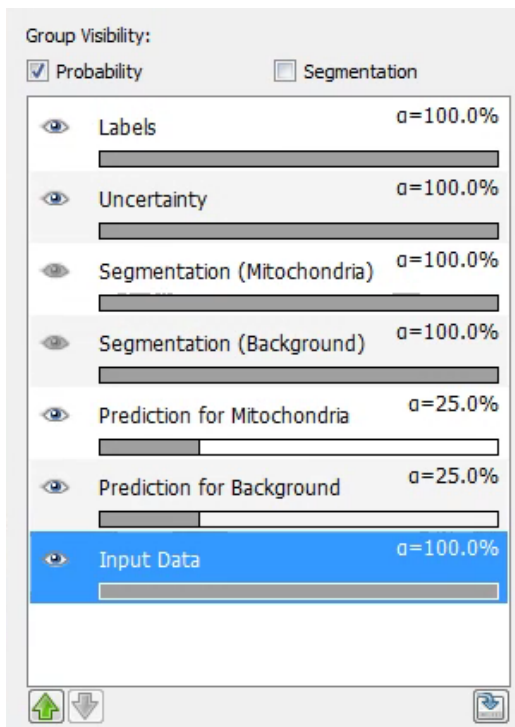
If you are interested how a possible segmentation would look like, you can turn on

Group Visibility>Segmentation

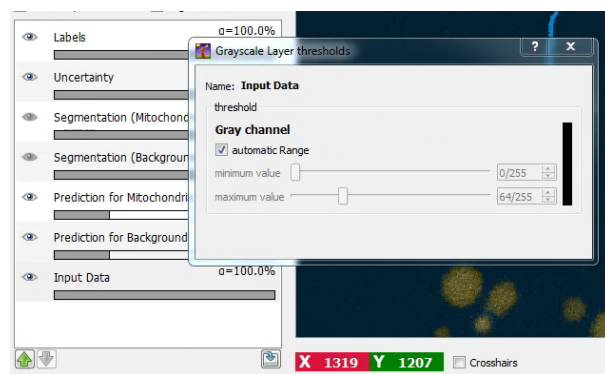
Keep in mind that Ilastik uses here a fixed probability threshold of 0.6. This means here that if a pixel has a probability of 0.6 or larger of being in the class "Mitochondria" it will be segmented yellow and otherwise as "Background" blue. Also keep in mind that other thresholds e.g. higher ones might give you better results.

6. Layers and Visualization

By clicking on the eyes in the lower left pane you can turn off and on the layers for labels, predictions etc.

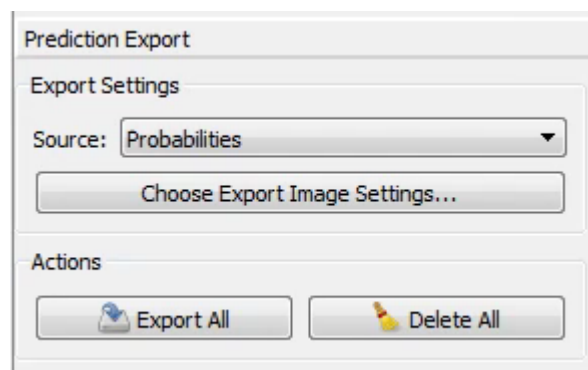


If you have difficulties to see the image intensities during training of the classifier, you can right click on the “Input Data” layer and a new window will open where you can manually adjust the visualization thresholds.

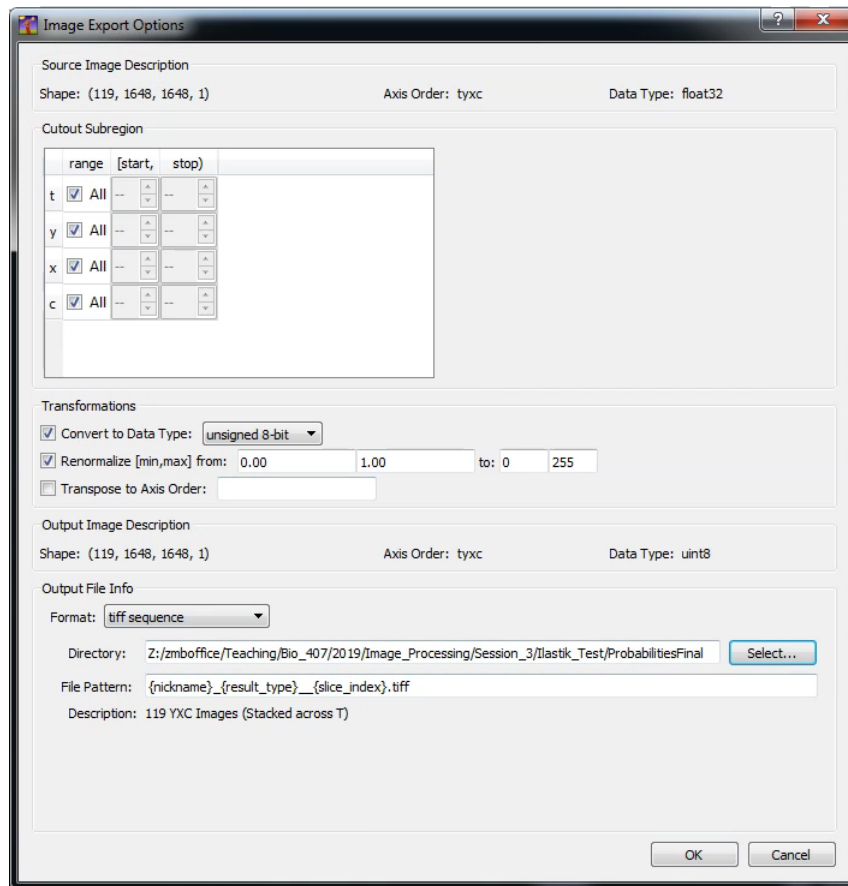


7. Export Results

Next we can go to the next step on the left pane “Prediction Export”. Here you can choose what to export



Since Ilastik uses a fixed classification probability threshold for segmentation/binarization, usually it is good practice to export **the probability images/maps** and to threshold and segment them **in another tool such as Fiji**. Adjust the export settings in “Chose Export Image Settings”



For using the data in Fiji, use the following export settings:

- Convert to Data Type: “unsigned 8-bit”
- Renormalize [min, max] from: “0.00” “1.00” to: “0” “255”
- Format: tiff sequence

For exporting the probability maps of all images press

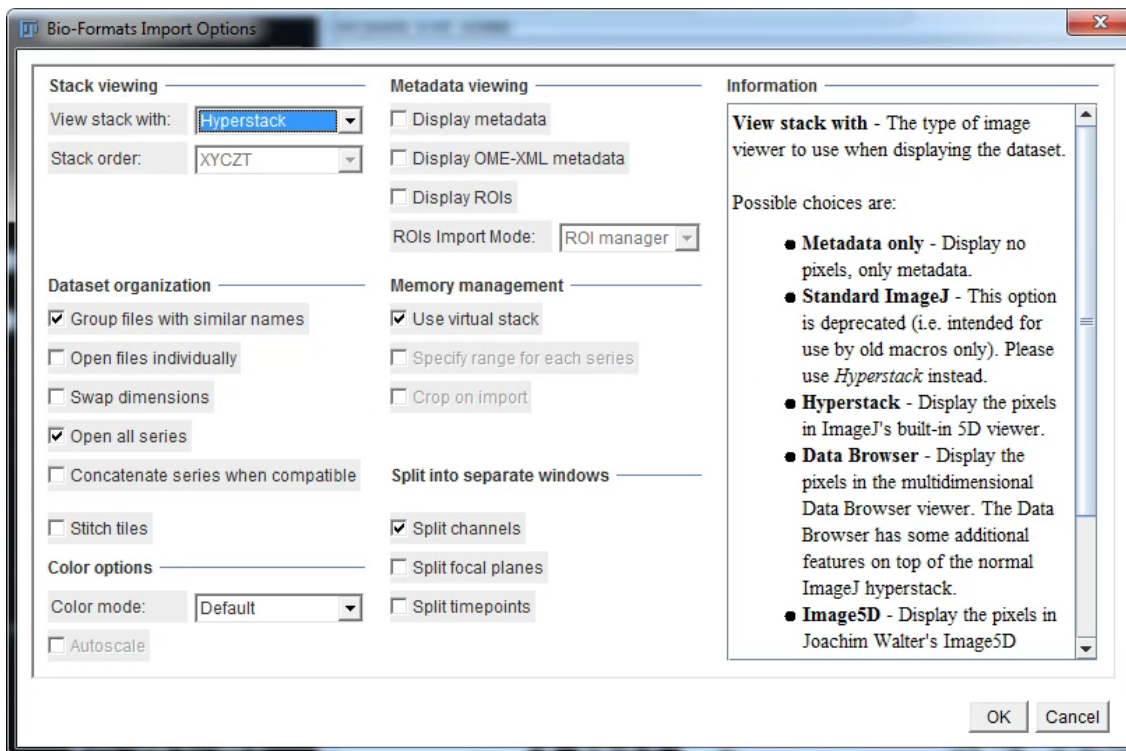
Prediction Export>Actions>Export All

Now Ilastik processes and exports all slices of your time series which can take a bit of time.

8. Open and Work with the Probabilities in Fiji

You can open the resulting tif files in Fiji. Be sure to specifically use

File>Import>Bio-Formats



Trying to open the files in Fiji using drag and drop might give you an error message.

Next, try to explore and understand the resulting files using Fiji. Also try to threshold and quantify one of your probability maps. Merge the thresholded probability maps with the original dataset for clearer visualization of the results.

9. Literature and Further Information

- Segmenting and Tracking Multiple Dividing Targets Using ilastik.
<https://www.ncbi.nlm.nih.gov/pubmed/27207368>
- Official Ilastik documentation
<https://www.ilastik.org/documentation/index.html>
- Ilastik official github page
<https://github.com/ilastik/ilastik>
- If you want to learn more about random forest classifiers
<https://youtu.be/LDRbO9a6XPU>
https://youtu.be/D_2LkhMJcfY