A flexible image segmentation pipeline based on pixel classification for heterogenous multiplexed tissue images

Bodenmiller Lab

Universität Zürich[UZH]

Vito Zanotelli, Nicolas Damond

www.bodenmillerlab.com

October 2nd, 2019

User group meeting workshop

Goal: Segment raw IMC files to generate cell masks

- Why image segmentation ?
- Why using a pixel classification-based approach ?
- Hands-on session: IMC image segmentation pipeline

  - Concepts

  - File format and metadata requirements

  - A practical walk through the pipeline

Why and when does image segmentation make sense for multiplexed tissue image analysis?

Biologically relevant entities:

- Organism
- Tissue
- Cell
- Cell compartments
- Molecules

→ A 1 x 1 x 5 µm pixel

does not represent any of these !

Issue: Pixel data is noisy and pixel size arbitrary

→ Partitioning the images into relevant regions (= segmentation) is a solution

Segmentation:

- Averages pixel over “relevant” regions
- Decreases noise
- Summarizes data in a relevant way

“Relevant” is question dependent: cells or tissue structures can be relevant, e.g.:

- Intensity levels within structure
- Intensity distribution within structure
- Distribution of these structures in the images

Major caveats

IMC resolution: 1 x 1 µm resolution, 5 µm thickness

- Subcellular compartments can physically overlap in a pixel
- Cells can physcially overlap in a pixel
- 5 µm only captures a slice of a cell (not always representative of overall cell marker profile)

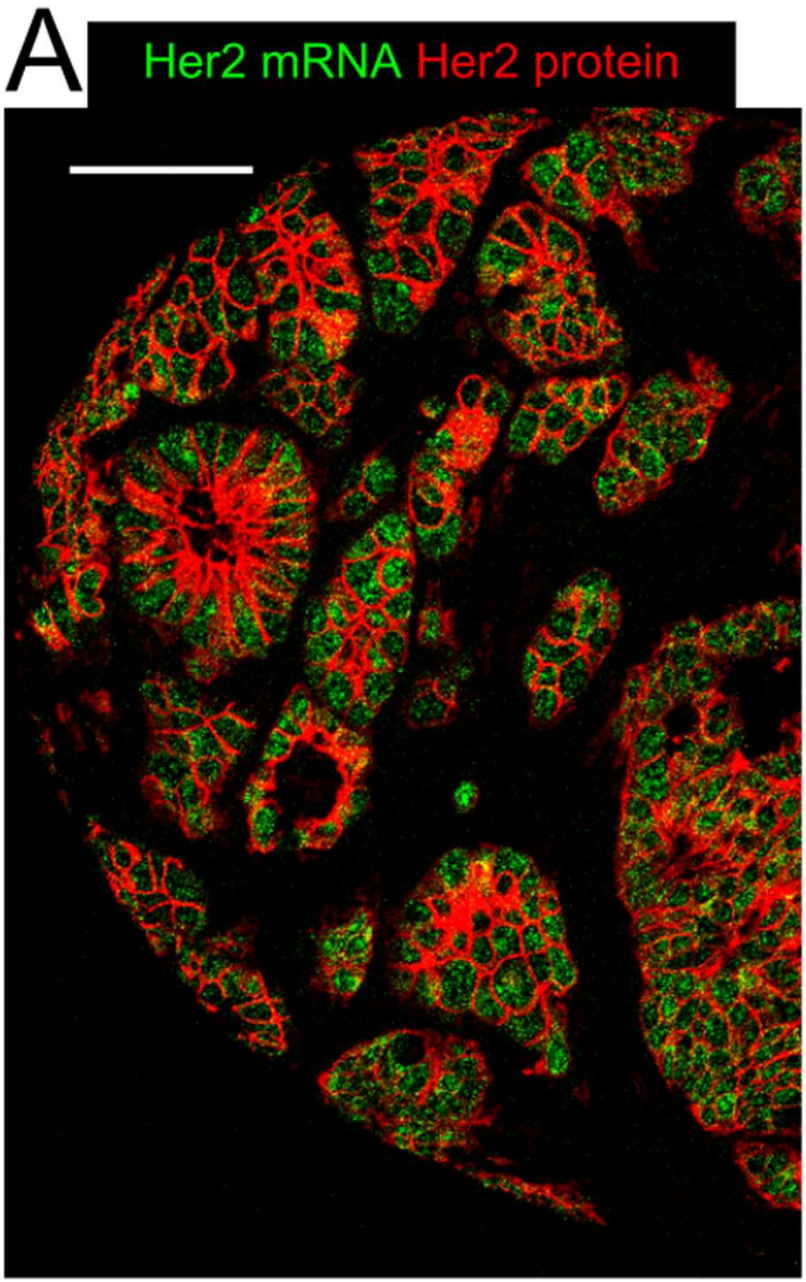
⇒ IMC-based cell phenotyping is more noisy than e.g. Flow Mass Cytometry, even with 'optimal' segmentation

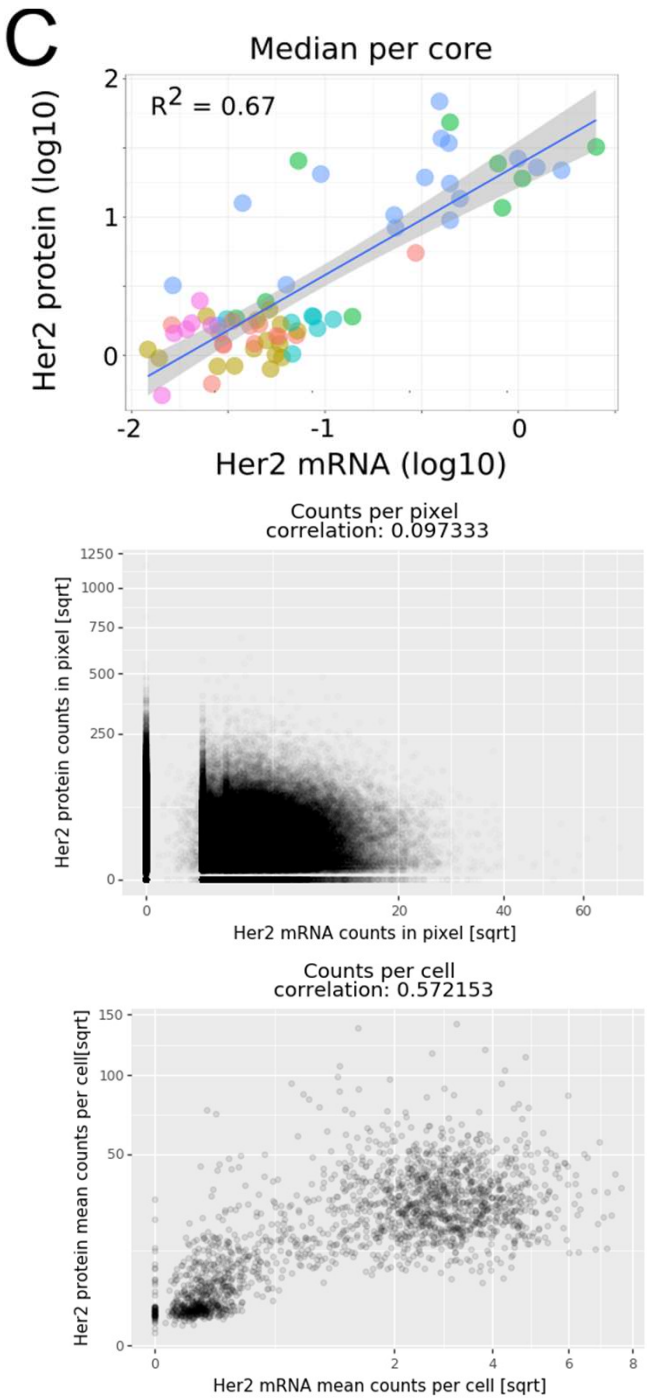
But,

→ Still good enough for cell phenotyping

→ Spatial context and subcellular information

Example | Her2 mRNA vs protein in breast cancer





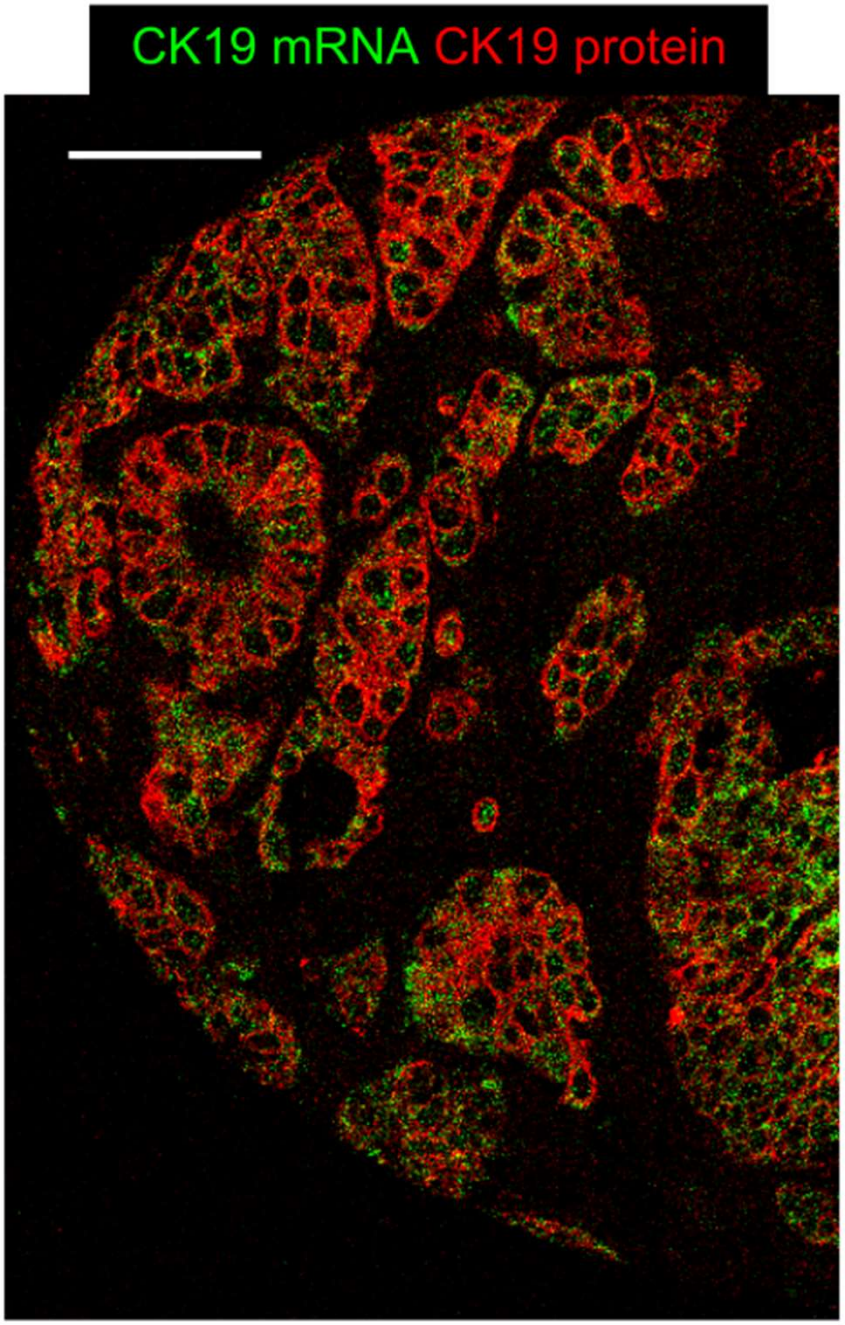
Good correlation of median signal per image over patients

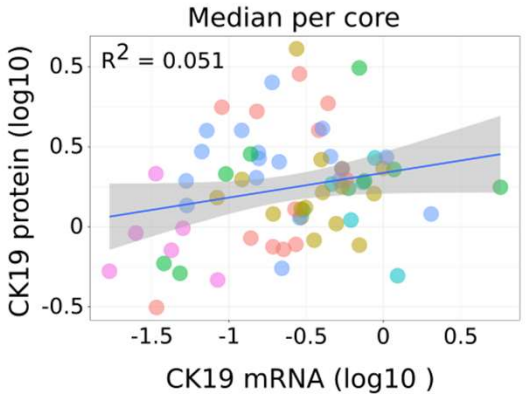
No or very weak correlation within pixels of tissue area of an image

Strong correlation between mRNA and protein between segmented cells of a core

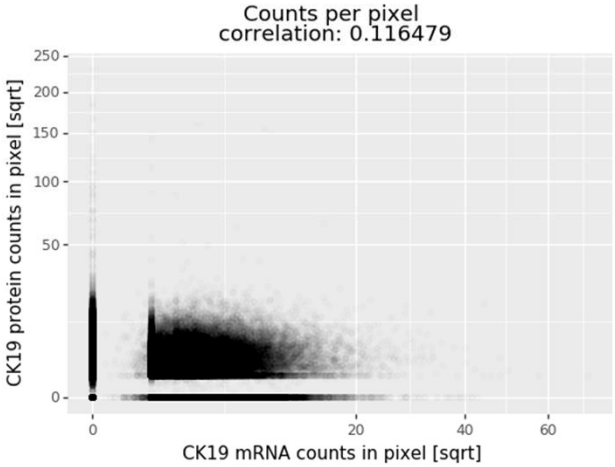
Bodenmiller Lab, 2019

Example | CK19 mRNA vs protein in breast cancer

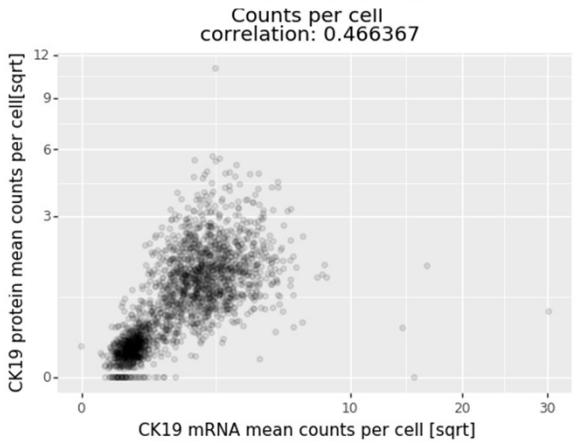




No correlation of median signal per image over patients



No correlation within pixels of tissue area of an image

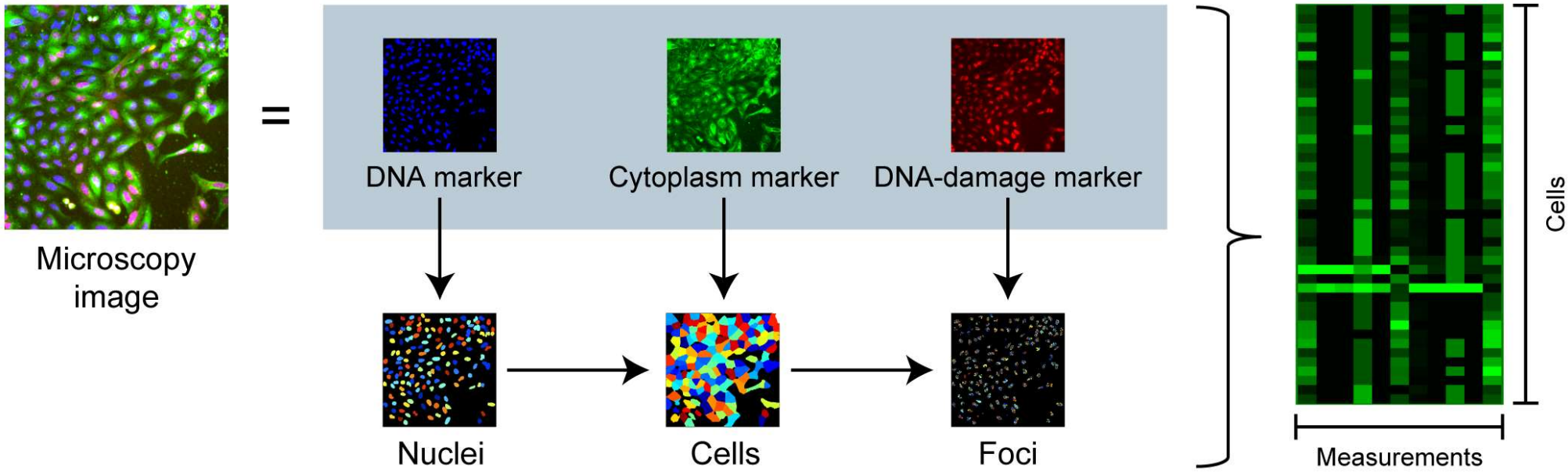


Strong correlation between mRNA and protein between segmented cells of a core

Bodenmiller Lab, 2019

Why using a pixel classification-based approach ?

Classical cell-segmentation pipeline:



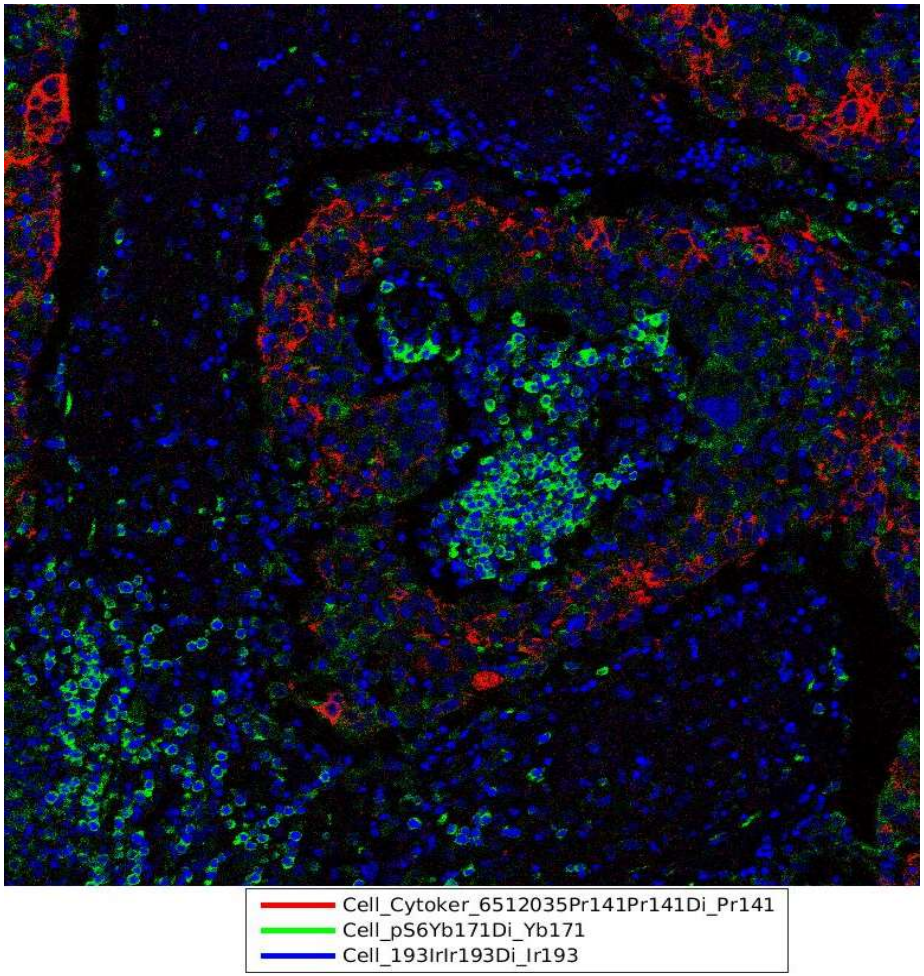
(Introduction to the Quantitative Analysis of Two-Dimensional Fluorescence Microscopy Images for Cell-Based Screening, Ljosa & Carpenter, 2009)

1. Find cell nuclei by using a nuclear marker → **nuclear mask**

2. Expand nuclear mask within cytoplasm until the membrane using a cytoplasmic or a membranous marker → **cell mask**

But: only uses information of 1-2 markers, dependent on actual channel intensities

Caveats of intensity-based approches



- Cytoplasmic/Membranous markers are strongly cell type-specific

→ Intensity varies from cell to cell

- Even Histone / DNA intercalator (iridium) show strong cell type-dependent intensity

→ Purely intensity based segmentation challenging

- Staining variability across an image / across images

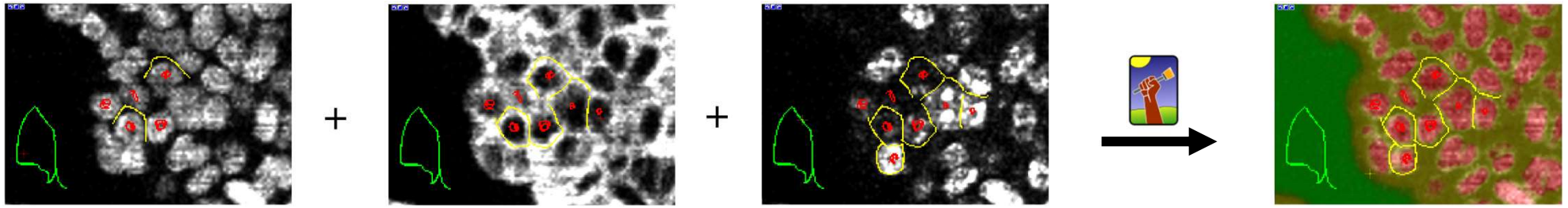
- Multiple markers contain information about the nature of a pixel: nuclear/cytoplasmic/membranous

- Marker distribution, textures, gradients matter to visually identify subcellular structures

Bodenmiller Lab, 2019

1. Tell the computer if you think a pixel is nuclear or cytoplasmic/membranous

2. Let the computer learn what you actually mean

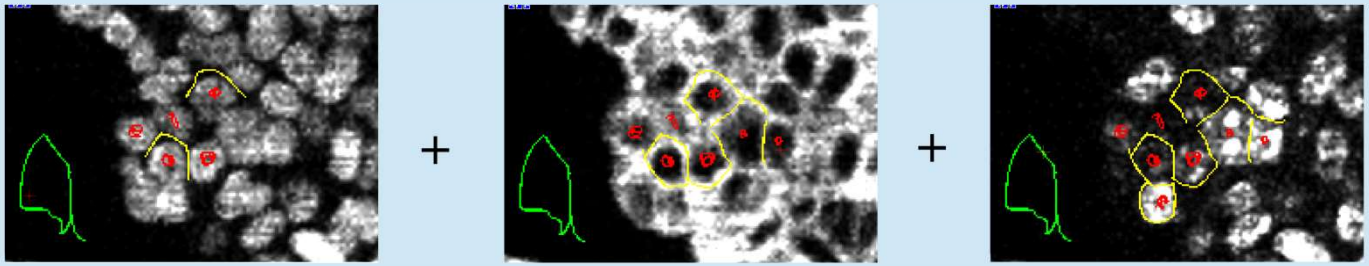


- Developed for low-dimensional images – but works even better for high dimensional images in our hands
- Based on manual input of human export knowledge
- Automatically integrates all channels as well as gradient and texture information of these channels
- Generates a highly normalized, easy segmentable “probability map” images
- Once well trained, works nicely even with heterogeneous tumor tissue

  → more than 800 images classified with one classifier

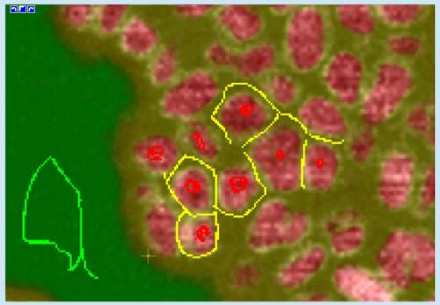
  → the classifier can be easily transferred to new images

Channel-based painting: nucleus, cytoplasm/membrane, background



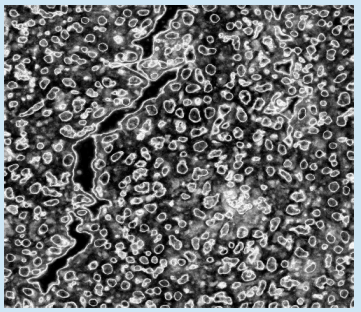
Classify

Train uncertain regions



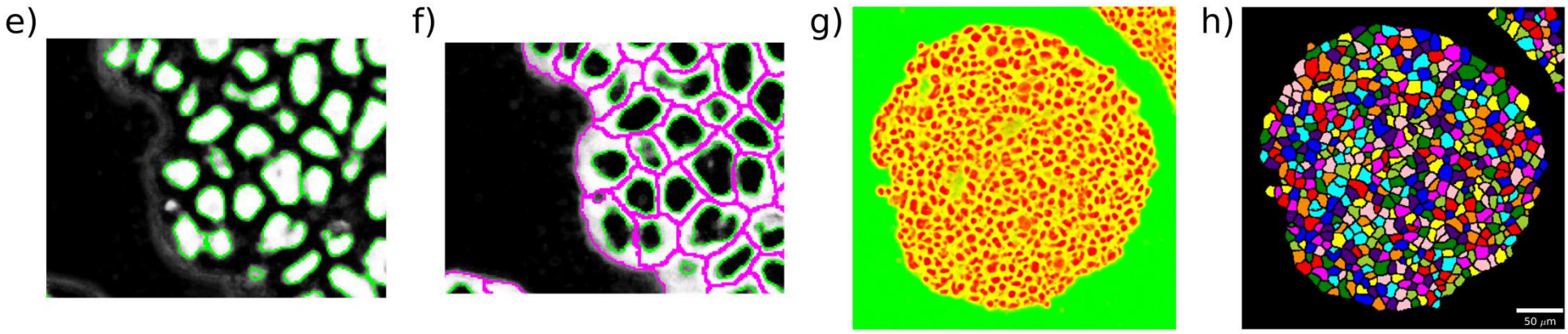
Check probability maps

Export uncertainty maps



Identify regions that need additional training

Segmentation using CellProfiler



- Segmentation of well trained probability maps using CellProfiler is robust

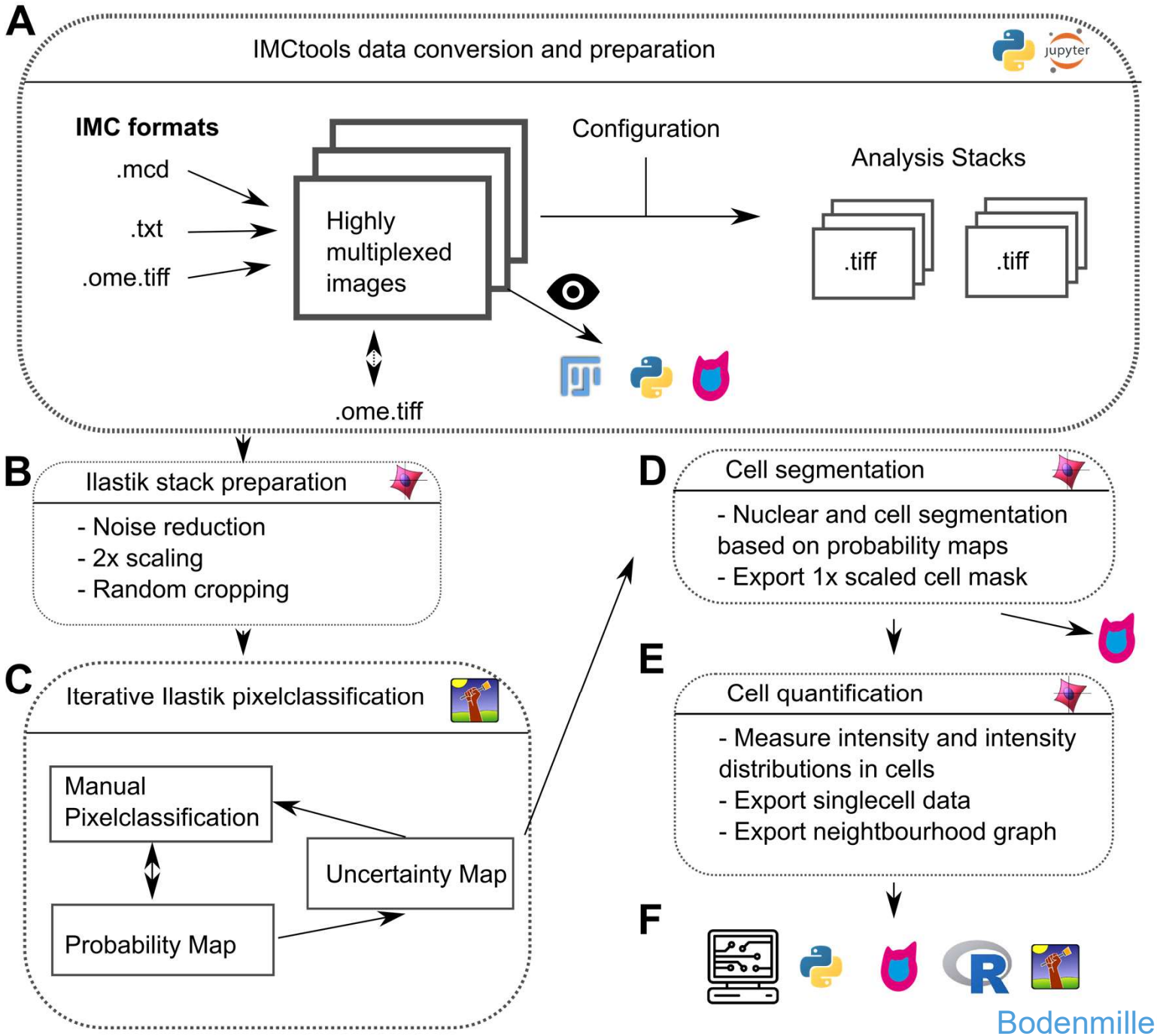
  → check uncertainty!

- One pipeline is applicable to the probability maps of very large datasets

- The pipeline generates single cell masks

  → Can be used for HistoCAT

  → Can be used to extract single-cell data from IMC images


A
IMCtools data conversion and preparation
jupyter
IMC formats
.mcd
.txt
.ome.tiff
Highly multiplexed images
Configuration
Analysis Stacks
.tiff
.tiff
.ome.tiff
B
Ilastik stack preparation
- Noise reduction
- 2x scaling
- Random cropping
C
Iterative Ilastik pixelclassification
Manual Pixelclassification
Uncertainty Map
Probability Map
D
Cell segmentation
- Nuclear and cell segmentation based on probability maps
- Export 1x scaled cell mask
E
Cell quantification
- Measure intensity and intensity distributions in cells
- Export singlecell data
- Export neightbourhood graph
F
R

- **IMC data**

  - .mcd file(s)

  - associated .txt files

One zipped folder should contain one .mcd file

+ (optional) the associated txt files

Example in *.../2019_IMCWorkshop/data/zipfiles* Do not unzip!

- **Antibody panel**, csv file containing the following column:

  - "Metal tag": metal isotope, format: In113.

  - "Full": boolean (0,1): which channels to use for single cell measurements.

  - "ilastik": boolean (0,1): which channels to use for ilastik training.

Do not modify!

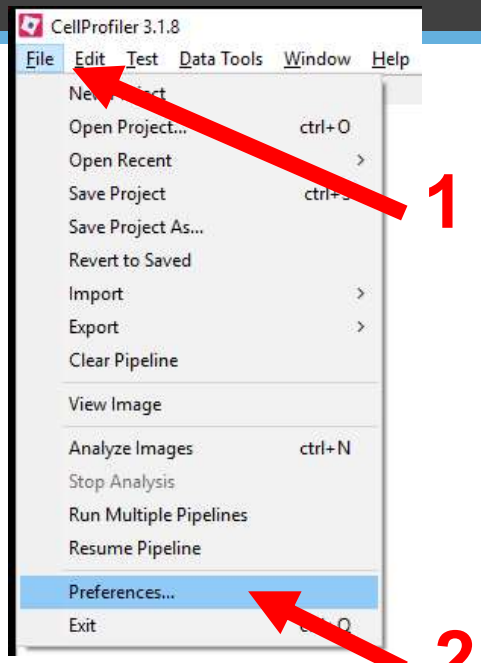
Example in *.../2019_IMCWorkshop/ImcSegmentationPipeline/config/example_panel.csv*

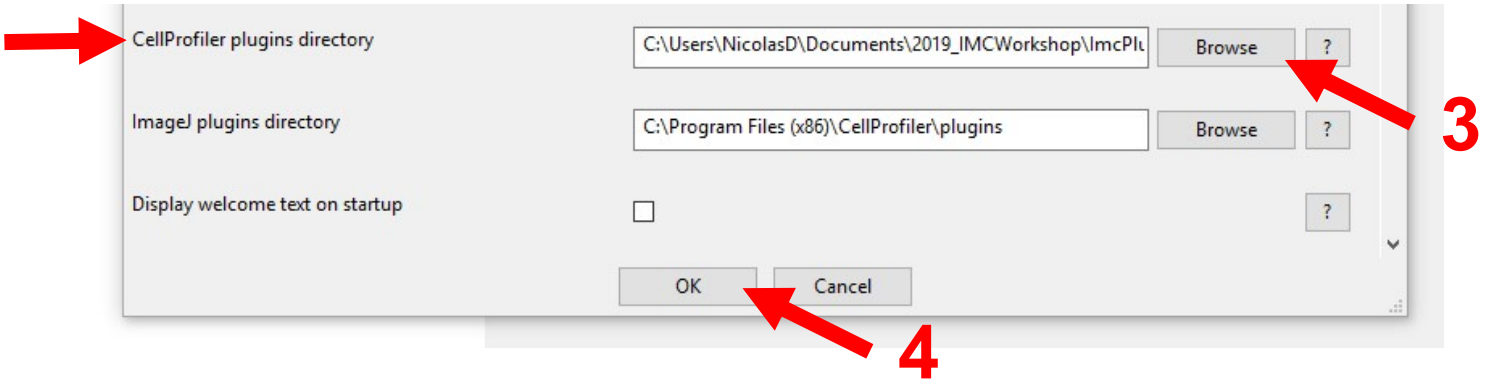
CellProfiler 3.1.8

- Try to open it

- Go to File (1) –> Preferences (2)

- CellProfiler plugins directory: Browse (3)





- Select this folder: *…/2019_IMCWorkshop/ImcPluginsCP/plugins*

- Click OK (4) and close CellProfiler

Anaconda (Python 3)

- Start the Anaconda Navigator

- In Applications on, select imctools (1)

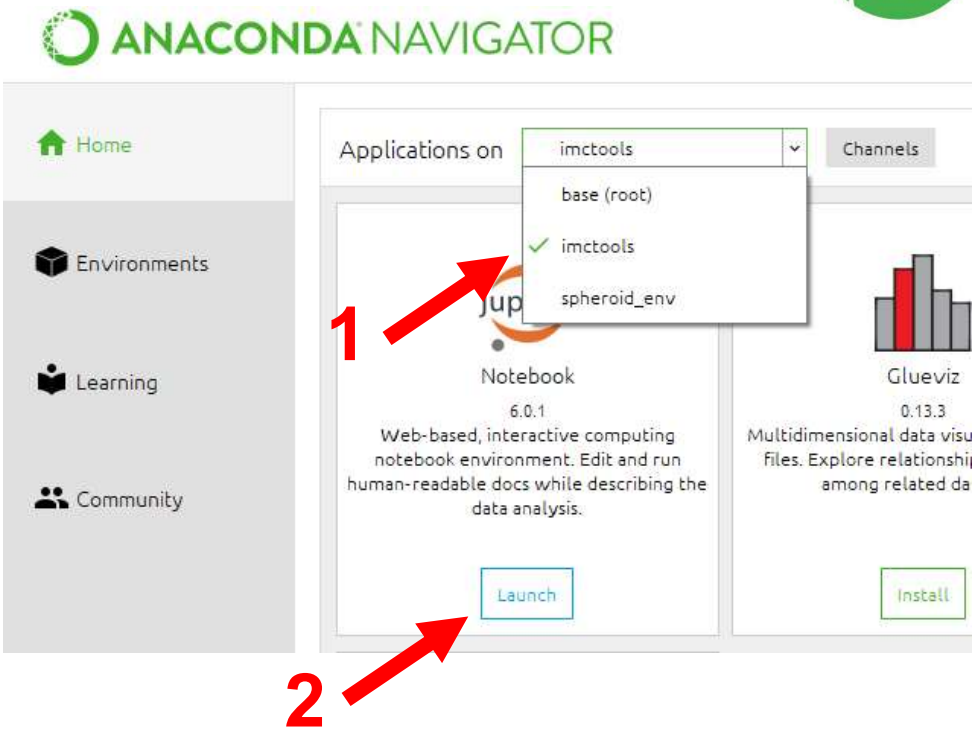
- Launch a Jupyter notebook (2)

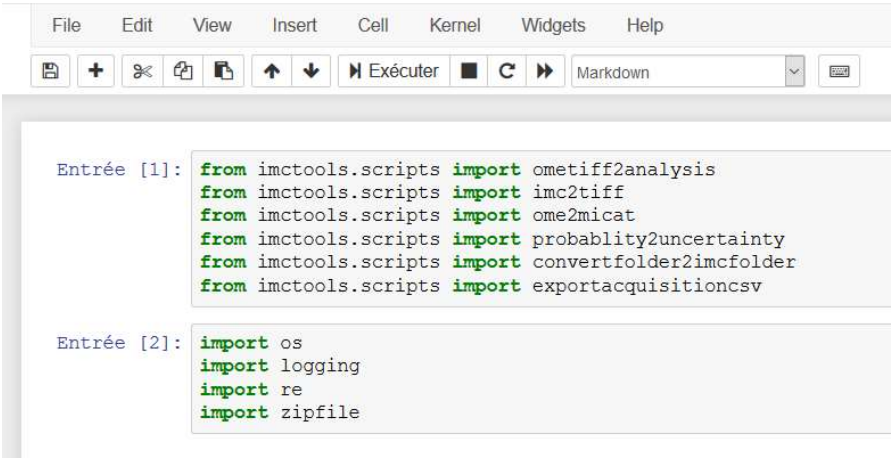
- Chose your favorite browser if prompted

- Navigate to this file and open it:

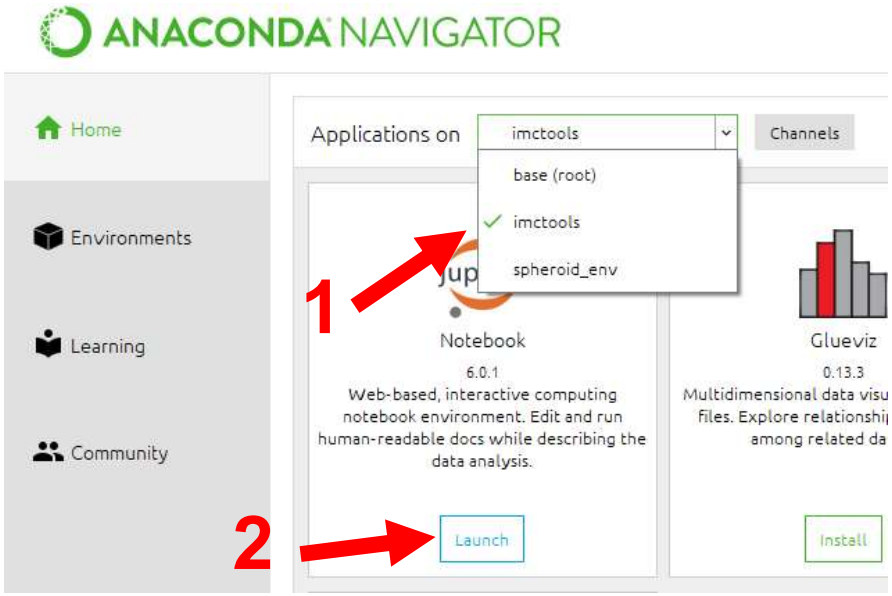
…/2019_IMCWorkshop/ImcSegmentationPipeline/scripts/ imc_preprocessing.ipynb

- Run the two first cells (click on “Run” or hit **`shift + enter`**)



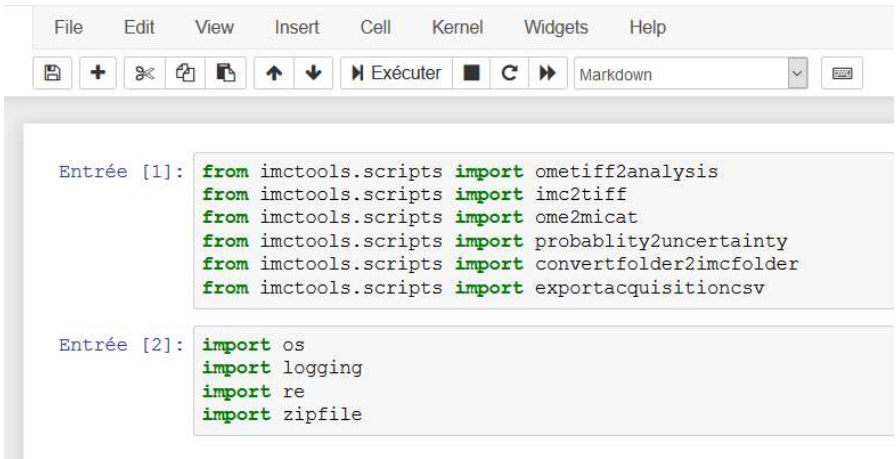


Hands-on session | Extract images from .mcd files

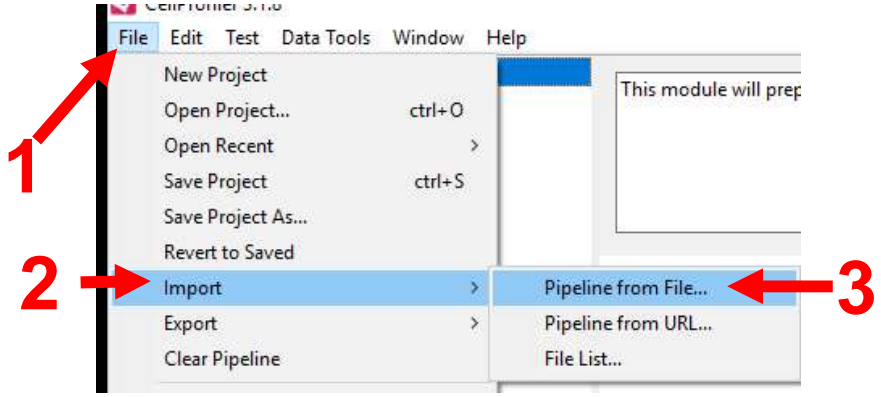


3 Navigate to this file and open it:

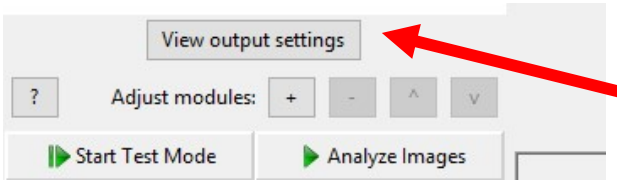
.../2019_IMCWorkshop/ImcSegmentationPipeline/scripts/ imc_preprocessing.ipynb

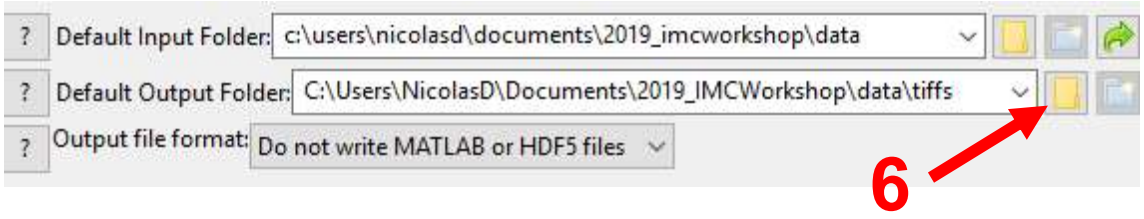


Hands-on session | Prepare images for ilastik

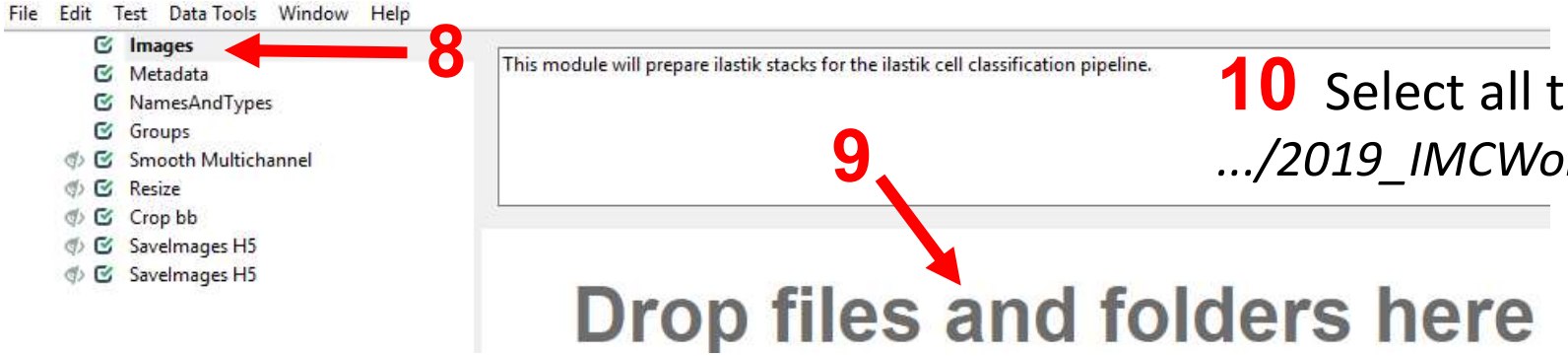


4 Select: *2019_IMCWorkshop/data/ImcSegmentationPipeline/cp3_pipelines/1_prepare_ilastik.cppipe*

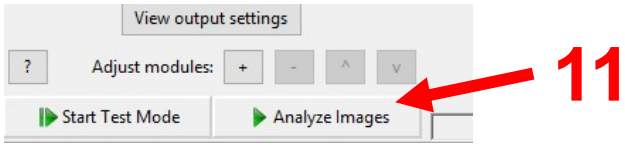


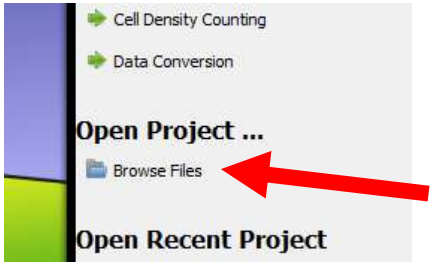


7 Select: *.../2019_IMCWorkshop/data/tiffs*



10 Select all the files in *.../2019_IMCWorkshop/data/tiffs*

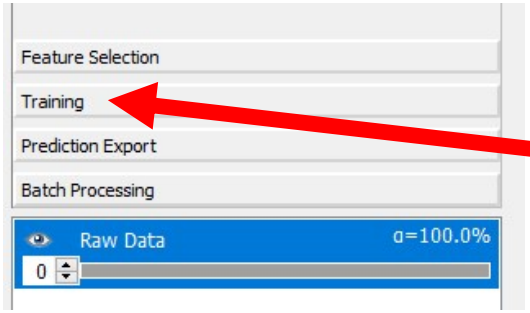




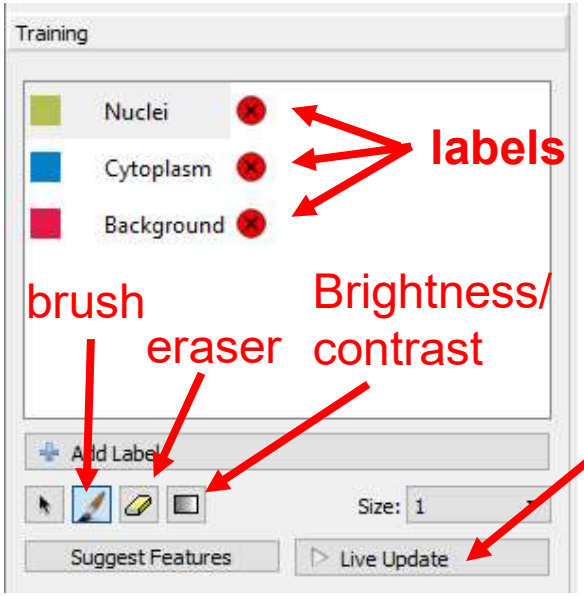
1

2 Select:

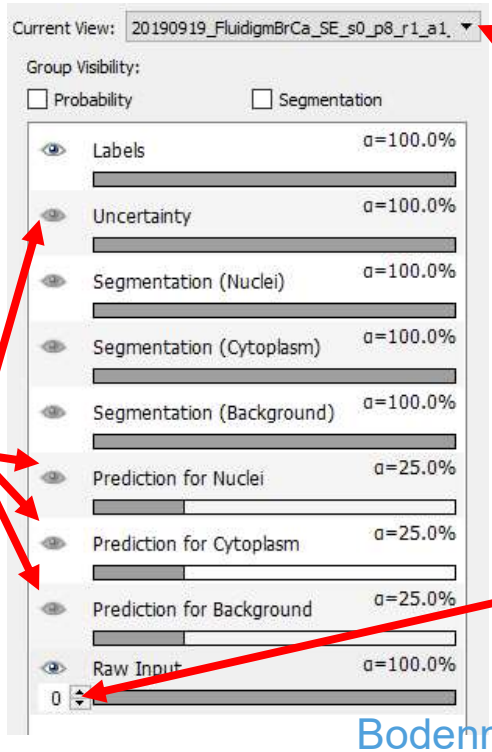
.../2019_IMCWorkshop/data/ilastik/pretrained_classifier.ilp



3



Enable to check your progress, disable to save resources

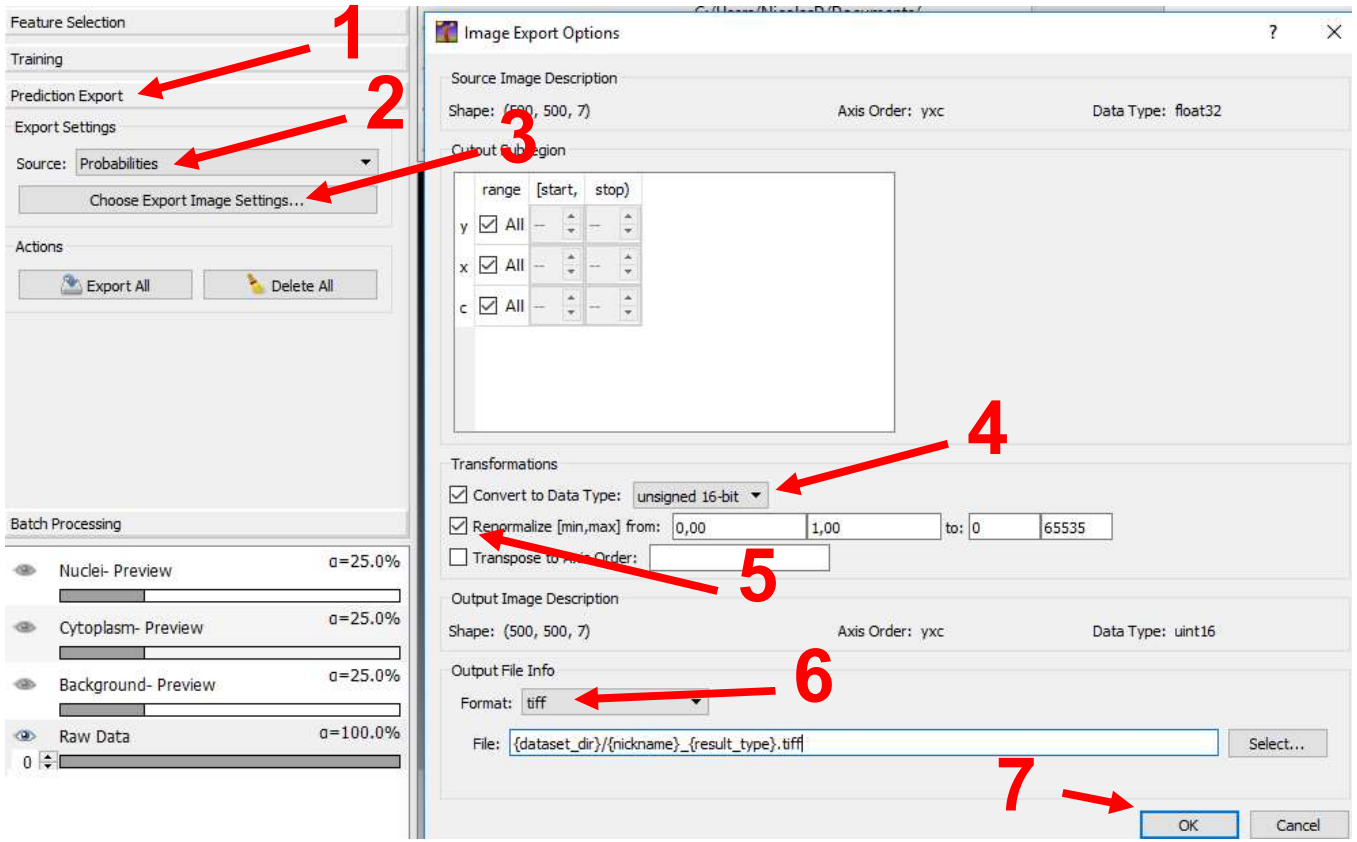


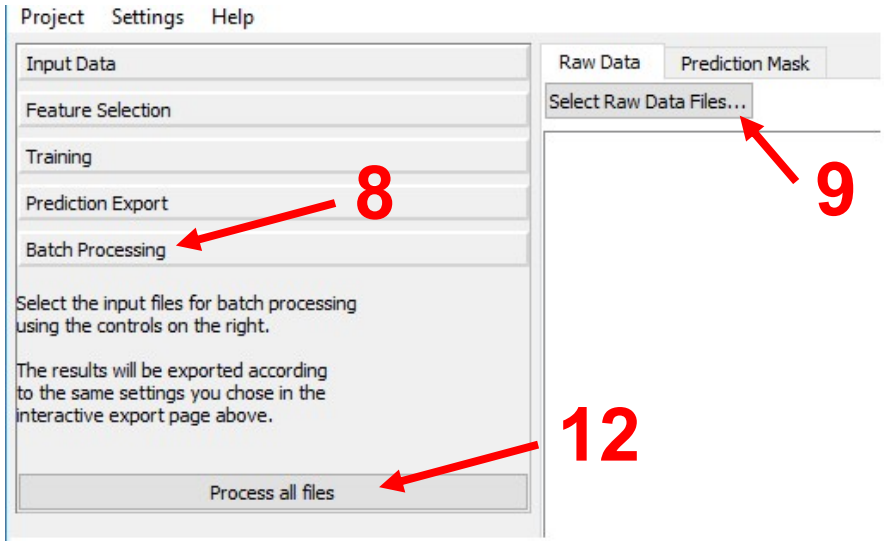
Currently displayed image

Display/hide layers

Channels

Hands-on session | Export probability maps

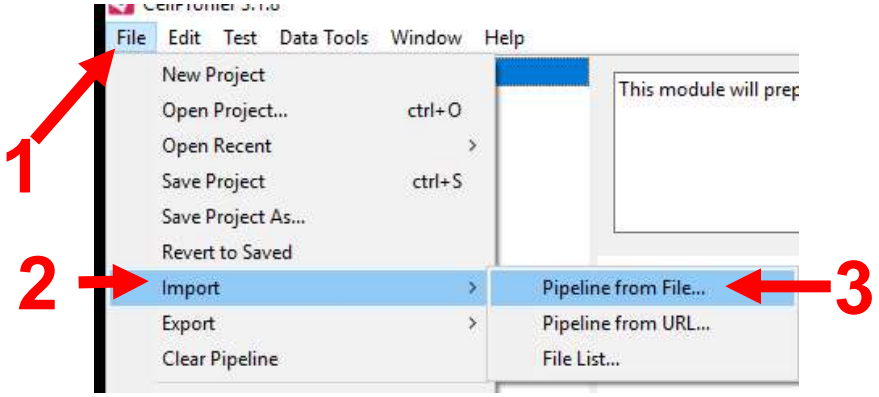




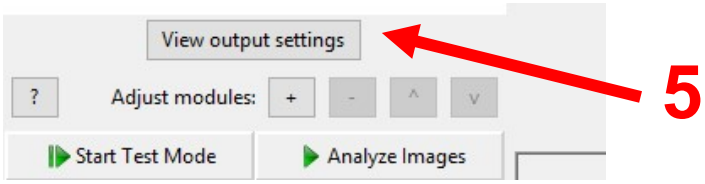
10 Go to: *.../2019_IMCWorkshop/data/tiffs*

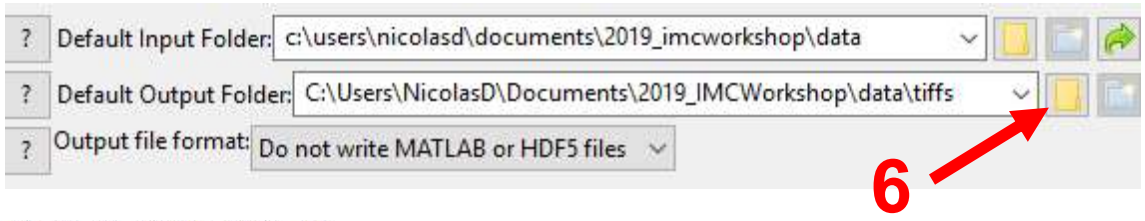
11 Select all **"_s2.h5"** files

Hands-on session | Generate cell masks



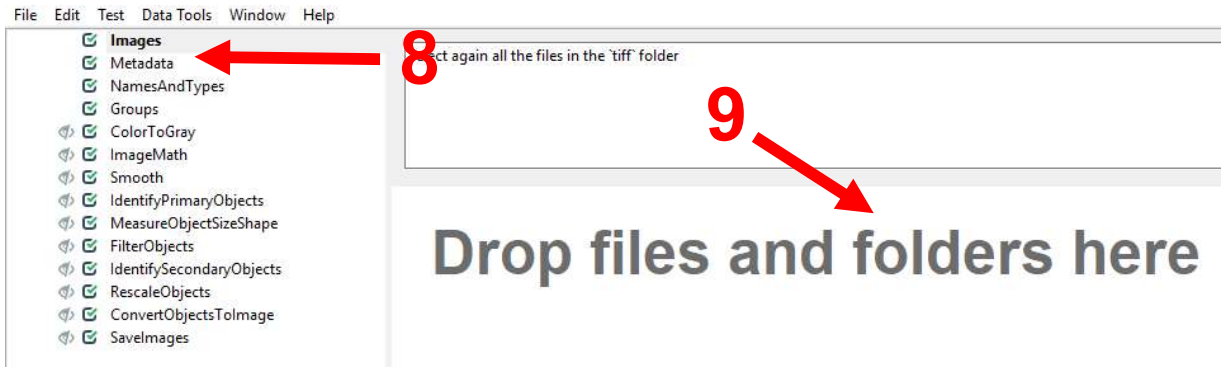
4 Select: *2019_IMCWorkshop/data/ImcSegmentationPipeline/cp3_pipelines/2_segment_ilastik.cppipe*



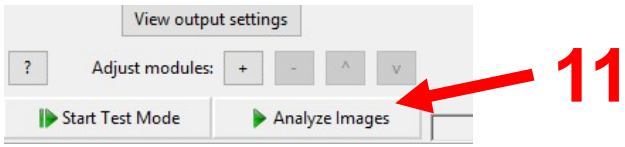


7 Select:

.../2019_IMCWorkshop/data/tiffs



10 Select all the files in *.../2019_IMCWorkshop/data/tiffs*



Hands-on session | Generate histoCAT folders

In Jupyter, this file should still be open:

.../2019_IMCWorkshop/ImcSegmentationPipeline/scripts/ imc_preprocessing.ipynb

Run this cell (at the very bottom of the script):

Generate histoCAT folders with masks

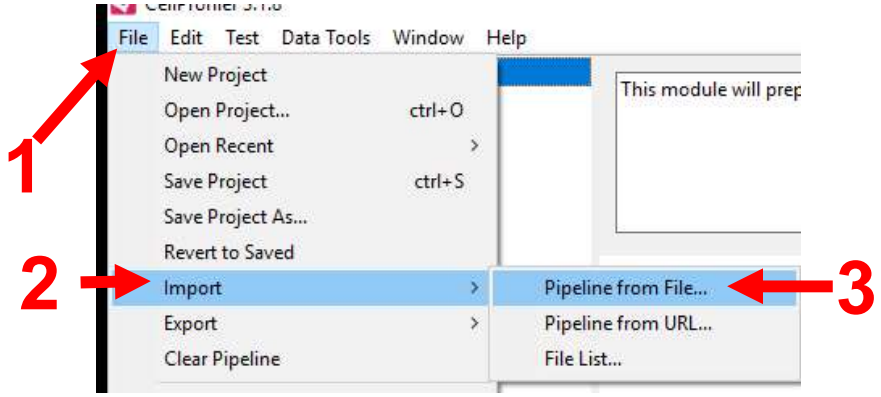
Entrée [ ]:
```
%%time
for fol in os.listdir(folder_ome):
    ome2micat.omefolder2micatfolder(os.path.join(folder_ome,fol), folder_histocat,
                                    fol_masks=folder_analysis, mask_suffix=suffix_mask, dtype='uint16')
```

The folders in *.../2019_IMCWorkshop/data/histocat*

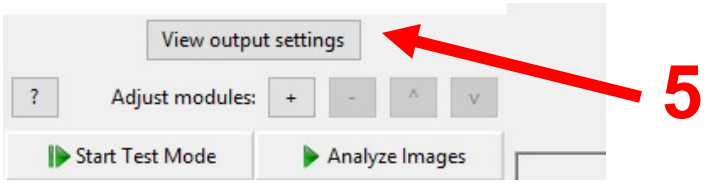
can directly be loaded into histoCAT

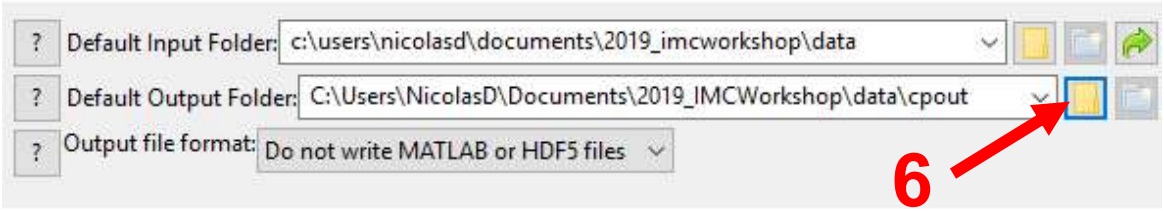
histoCAT
Histology Topography Cytometry Analysis Toolbox

Hands-on session | Extract single cell data



4 Select: *2019_IMCWorkshop/data/ImcSegmentationPipeline/cp3_pipelines/3_measure_mask_basic.cppipe*





7 Select: *.../2019_IMCWorkshop/data/cpout*

8 In Images, select all the files in *.../2019_IMCWorkshop/data/tiffs*