

Gisele Miranda

**KTH Royal Institute of
Technology**

**Division of Computational
Science & Technology, EECS**



gmirand@kth.se



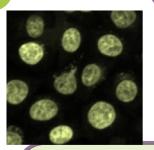
Workshop: Data Analysis Strategies for Single-Cell Image-Based Profiling:

Day 2: Cell Segmentation and Feature Extraction

Workshop overview



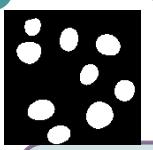
1



Introduction to Bioimage Analysis

- ❖ Basics of Digital Images
- ❖ File formats & Metadata
- ❖ Fund. Image Processing: filtering, denoising, etc
- ❖ Image Quality and reproducibility
- ❖ Hands-on session using Fiji

2



Cell Segmentation and Feature extraction

- ❖ Overview of segmentation techniques
- ❖ Introduction to machine and deep learning-based segmentation
- ❖ Feature extraction: measurements
- ❖ Exporting single-cell datasets for analysis

3



Single Cell Data Analysis

- ❖ Overview of high-content data structure
- ❖ Dimensionality reduction: PCA, LDA, t-SNE, UMAP
- ❖ Clustering methods: K-means, GMM
- ❖ Hands-on session with single-cell datasets

4



Daily Schedule

- ⌚ 09:00 – 10:20: First Session
- ☕ 10:20 – 10:40: Break
- ⌚ 10:40 – 12:00: Second Session
- 🍽 12:00 – 13:30: Lunch Break
- ⌚ 13:30 – 14:30: Third Session
- ☕ 14:30 – 14:45: Break
- ⌚ 14:45 – 16:00: Fourth Session

For today's practical exercises

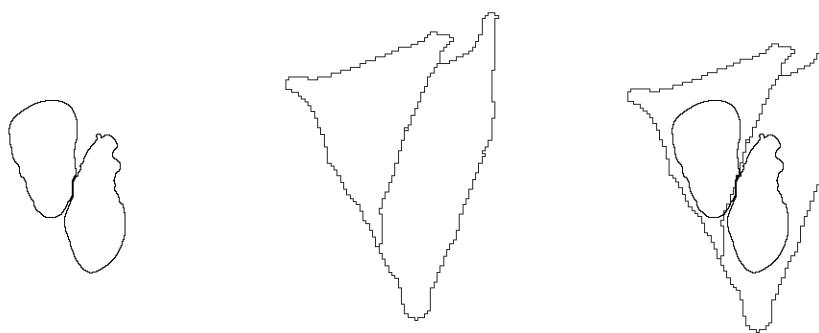
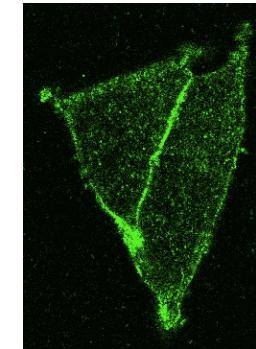
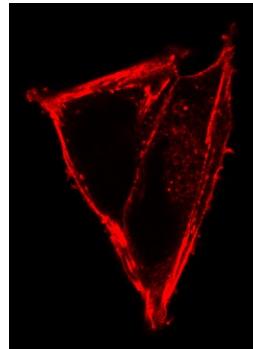
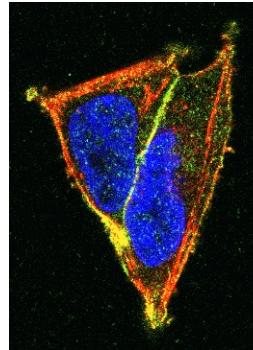
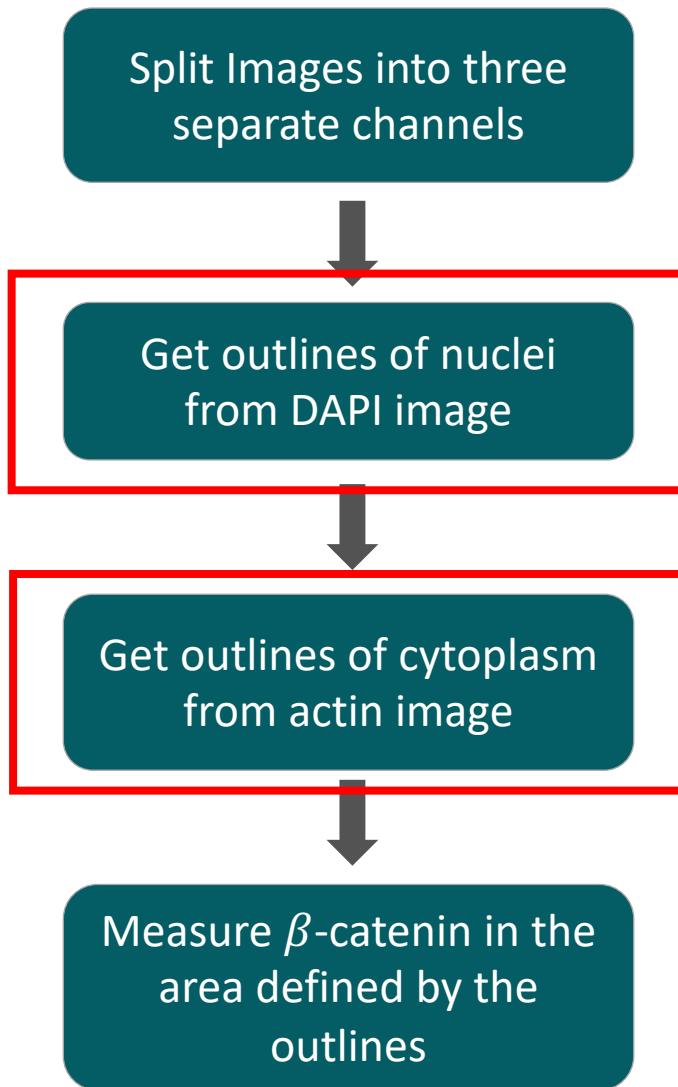


- Folder structure: data_day2-3

Folders	Folders	Folders	Folders	Images
JUMP-CP	Brefeldin-A-like	Orig	AGP	CP1-SC1-08_P06_T0001F001L01A04Z01C05.tif
JUMP-CP_complete	Etoposide		DNA	CP1-SC1-08_P06_T0001F002L01A04Z01C05.tif
JUMP-CP_mini	Nocodazole		ER	CP1-SC1-08_P06_T0001F003L01A04Z01C05.tif
	Rapamycin-like		Mito	CP1-SC1-08_P06_T0001F004L01A04Z01C05.tif
	Staurosporine-like		RNA	CP1-SC1-08_P06_T0001F005L01A04Z01C05.tif
				CP1-SC1-08_P06_T0001F006L01A04Z01C05.tif
				CP1-SC1-08_P06_T0001F007L01A04Z01C05.tif
				CP1-SC1-08_P06_T0001F008L01A04Z01C05.tif

Overview of segmentation techniques

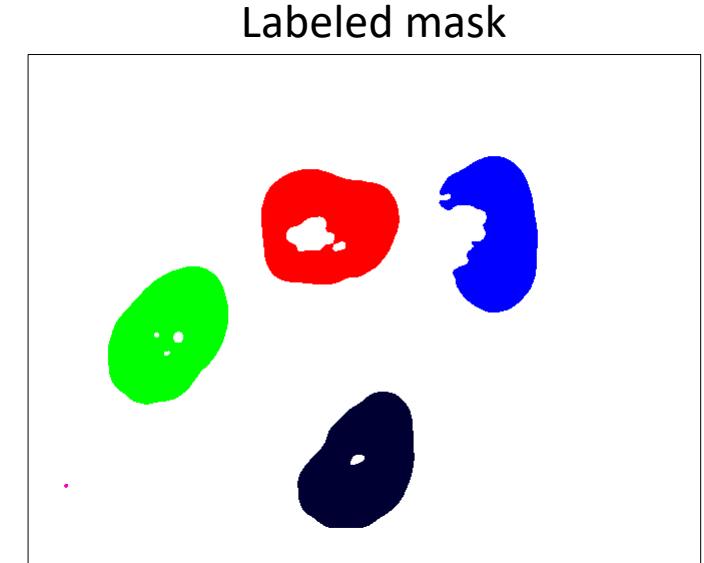
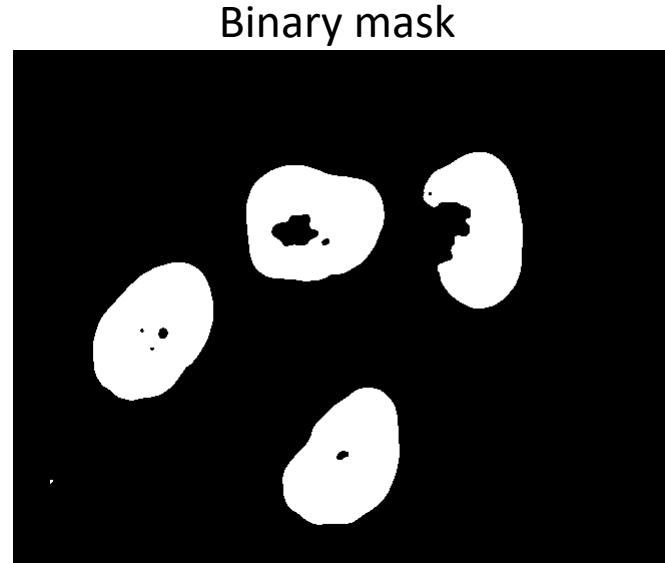
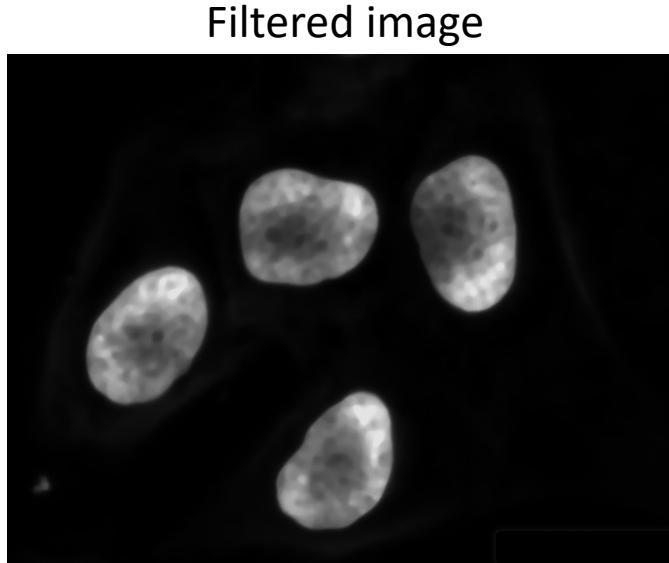
Summarizing the workflow...



Segmentation



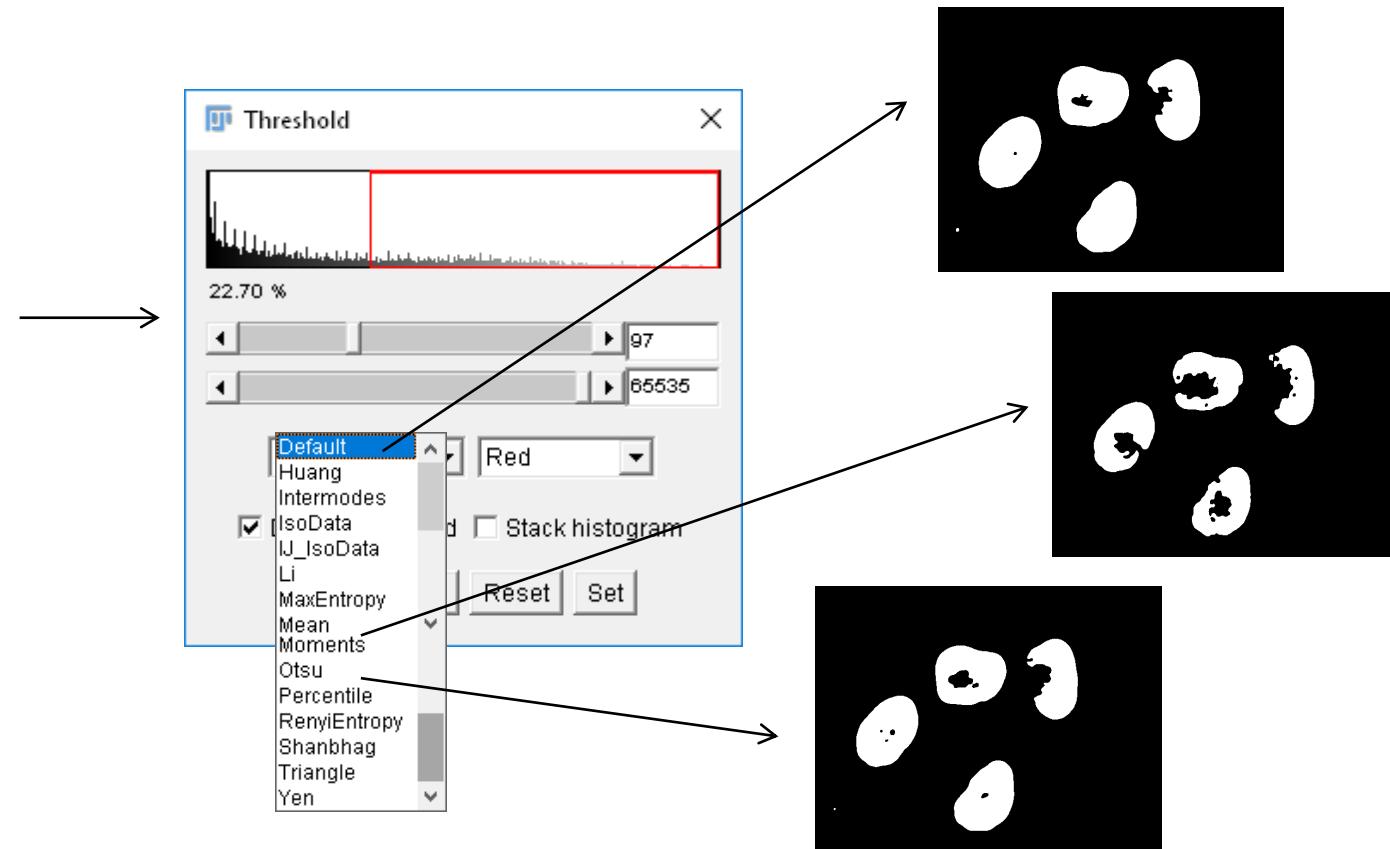
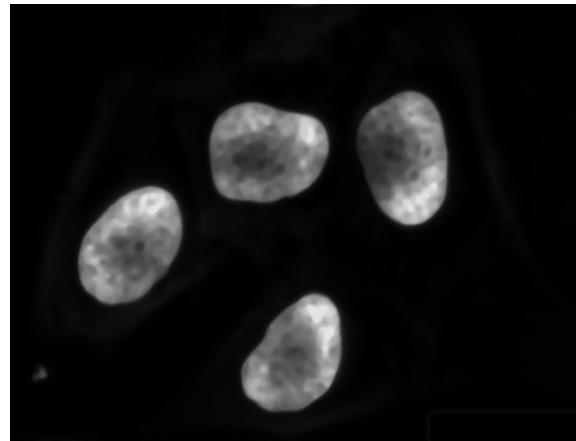
- Separates the objects of interest from the background
 - Threshold methods are the simplest segmentation approaches



Segmentation - thresholding



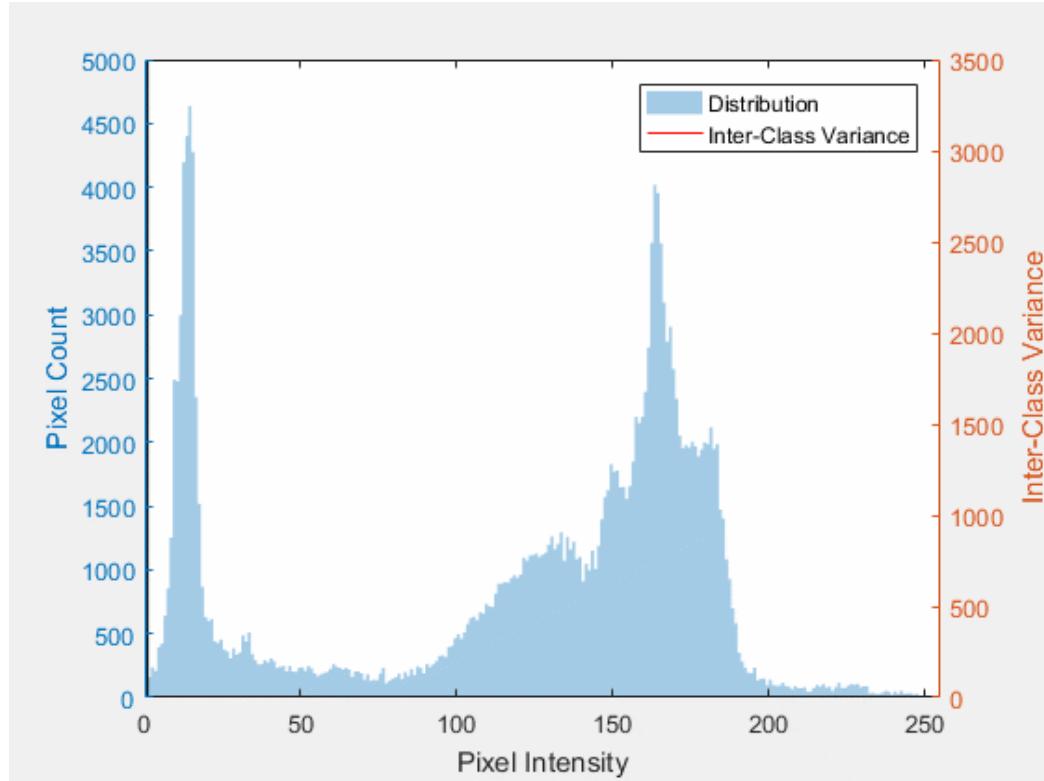
- Thresholding is a basic segmentation method
 - Manual thresholding vs automatic thresholding



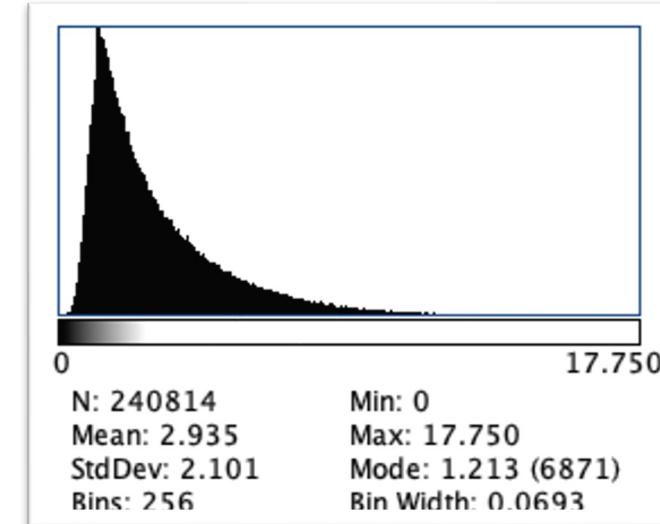
Segmentation - thresholding



- Otsu method



✓ Algorithm returns the intensity that maximizes inter-class variance between foreground and background



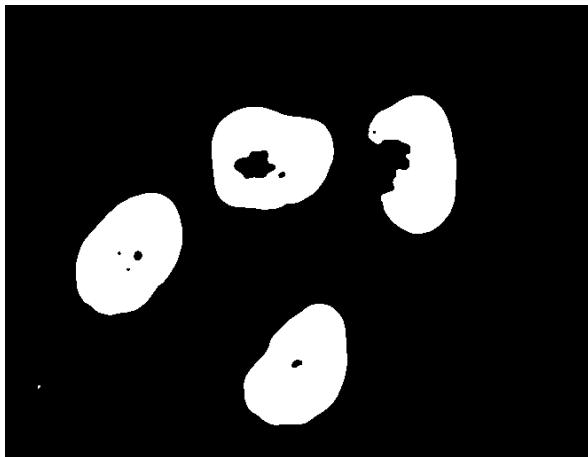
What about this histogram?

Post processing

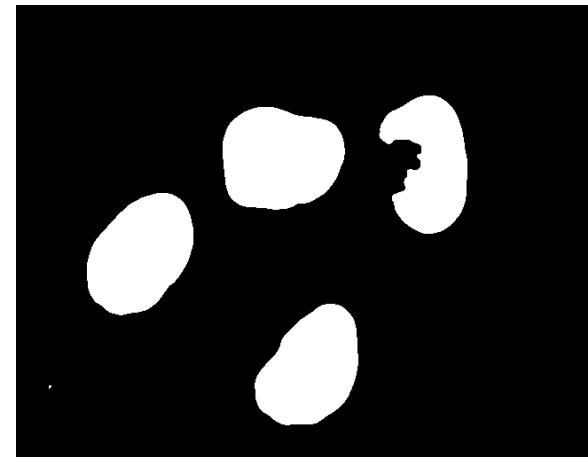


- Refinement of the binary mask
 - Ex.: fill holes present in the segmented objects; filter objects by area, size or circularity; filter objects by intensity, etc.

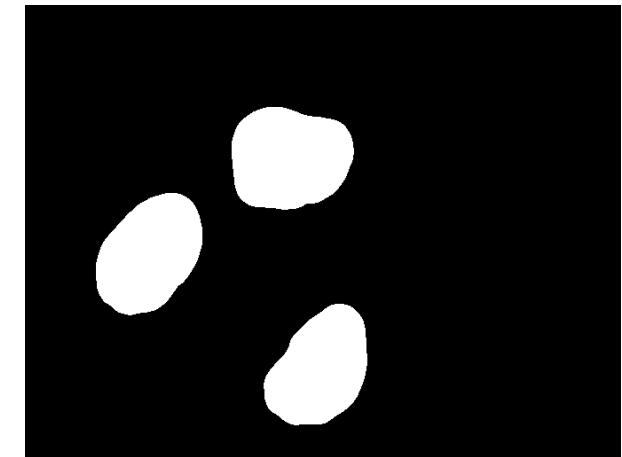
Original binary mask



Fill holes operation



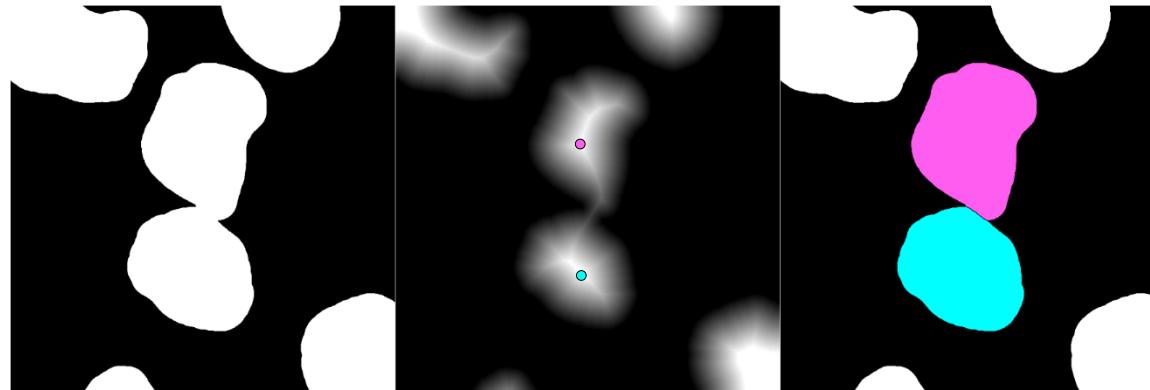
Binary mask filtered by circularity





Post processing

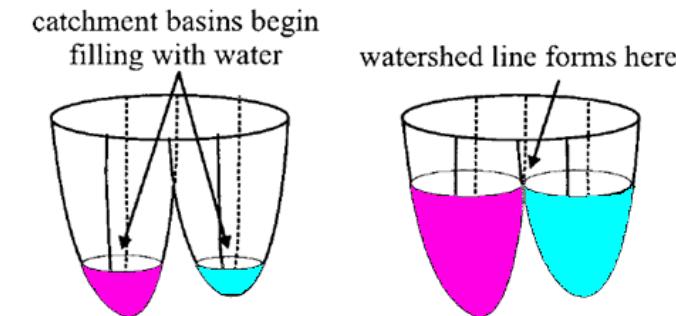
- Connected component analysis



Binary mask

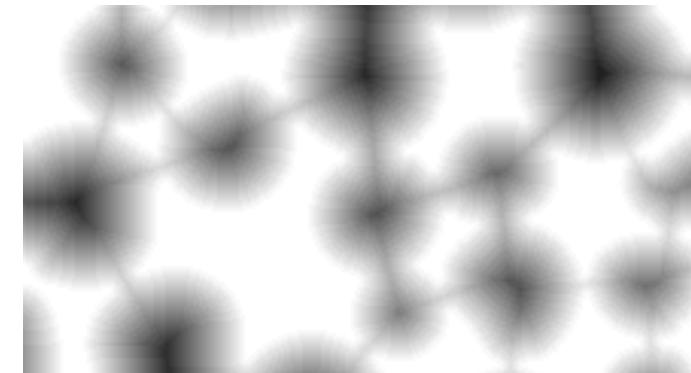
Euclidean Distance map

Separated Objects



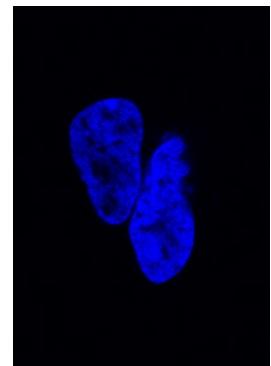
catchment basins begin
filling with water

watershed line forms here

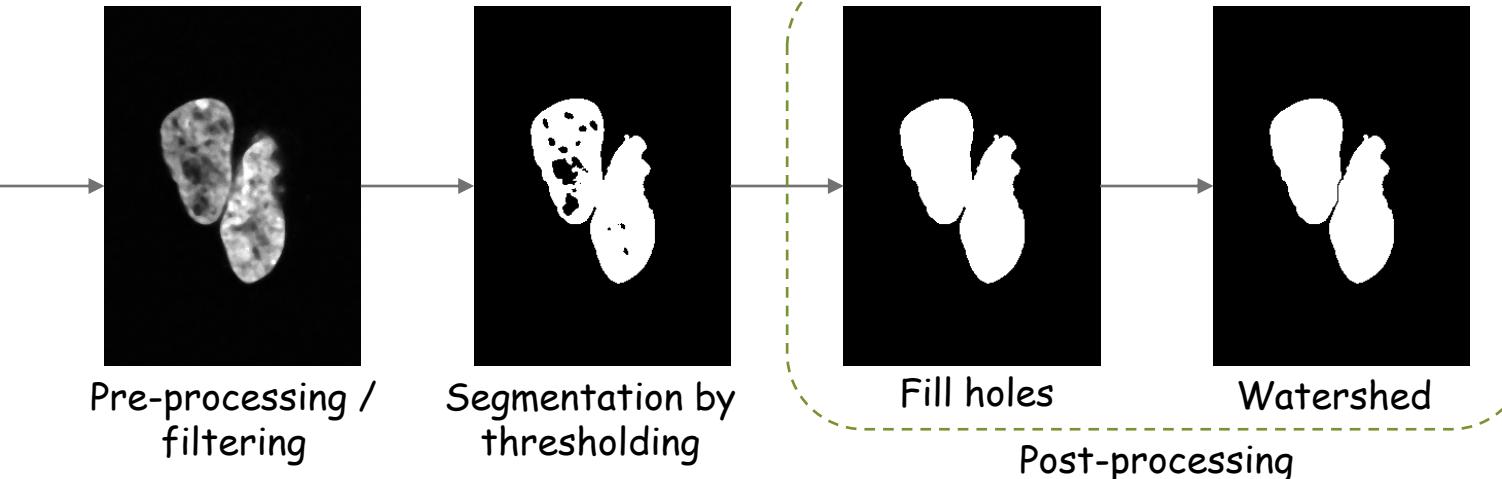


[https://imagej.nih.gov/ij/images/
watershed-animation.gif](https://imagej.nih.gov/ij/images/watershed-animation.gif)

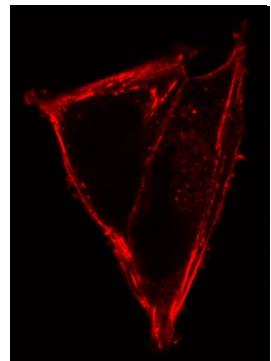
Get outlines of nuclei
from DAPI image



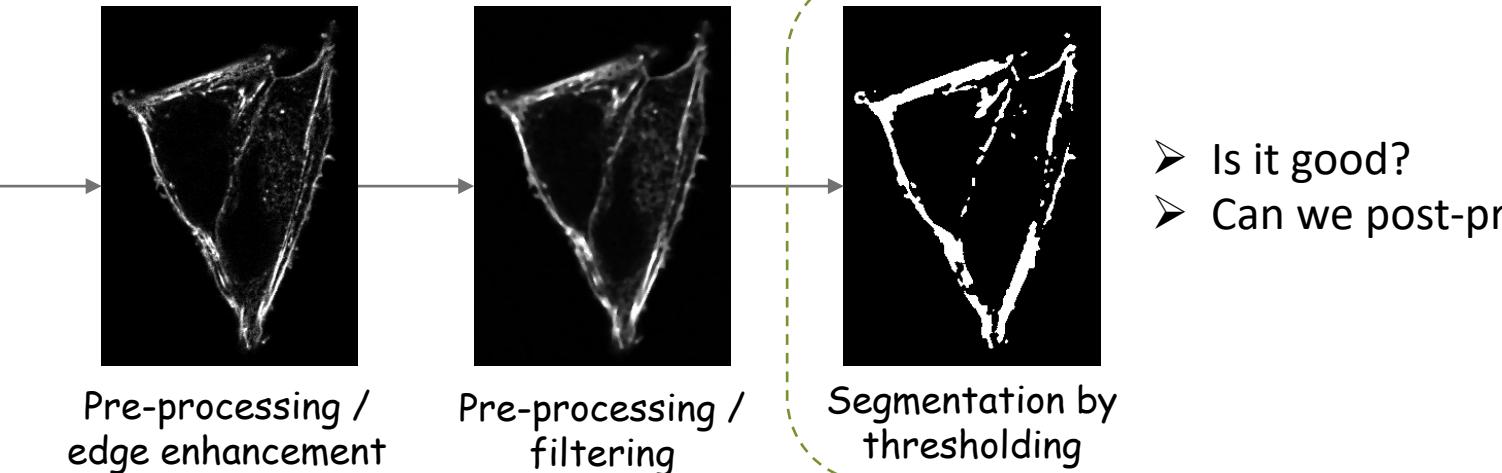
Nuclei



Get outlines of cytoplasm
from actin image



Actin



Practical Exercises



- Image files to be used in this session are located in
..../data_day2/part1/
 1. Open the file “Nuclei_no_avg.nd2”
 2. Subtract background (Process / Subtract Background...)
 3. Apply mean filter (Process / Filters)
 4. Go to Image / Adjust / Threshold and try different thresholding methods
 5. Once you select a method, run the ROI Manager to generate the Results table with area, Feret’s diameter, perimeter and mean gray value

Remember to always calculate measures over the original image, not the binary mask!

Batch Processing in Fiji with Macros



- **Why Use Macros?**

- ✓ Automate repetitive tasks (e.g., apply the same workflow to many images)
- ✓ Reduce user error and save time
- ✓ Reproduce analysis pipelines

- **What Is an ImageJ Macro?**

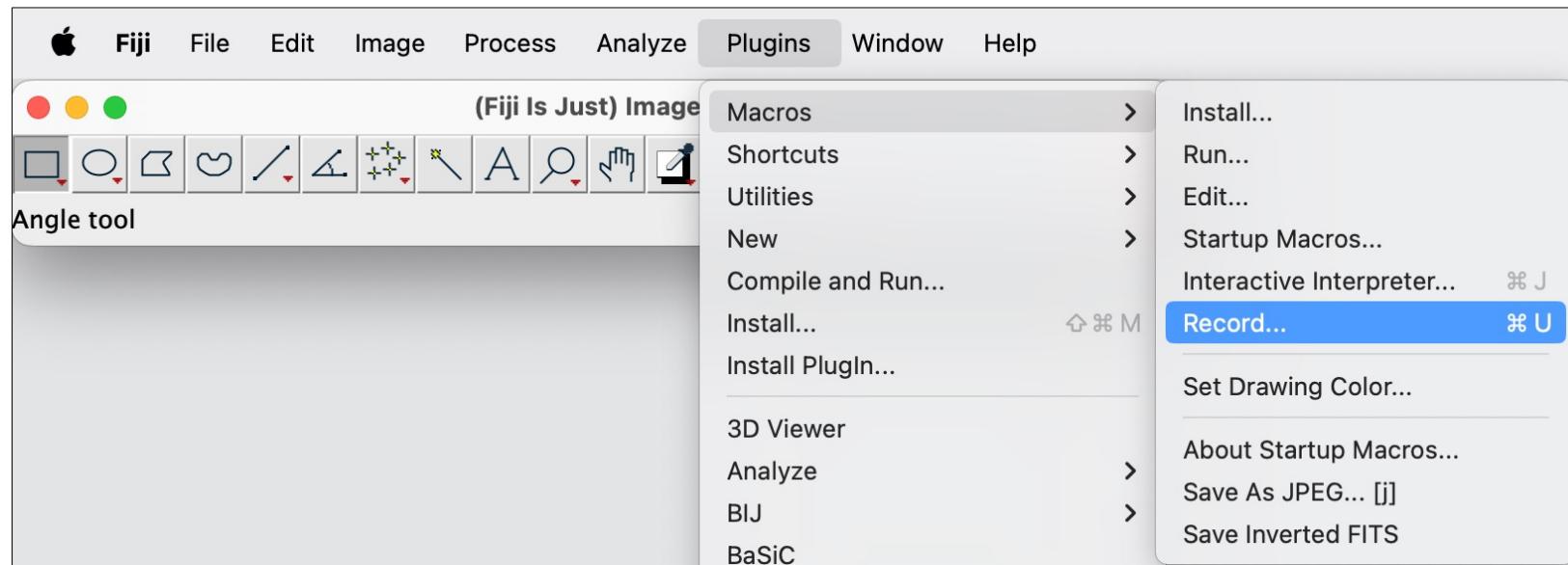
- ✓ A script that automates ImageJ commands
- ✓ Macros can:
 - Run filters, thresholding, segmentation
 - Process folders of images
 - Save output (e.g., measurements, masks)

Batch Processing in Fiji with Macros



- **Where to Start?**

- ✓ Use the **Macro Recorder** in Fiji: Plugins → Macros → Record...
- ✓ The recorder logs your GUI actions as macro code
- ✓ Tip: You can copy/edit the recorded macro to process full datasets!



Batch Processing in Fiji with Macros



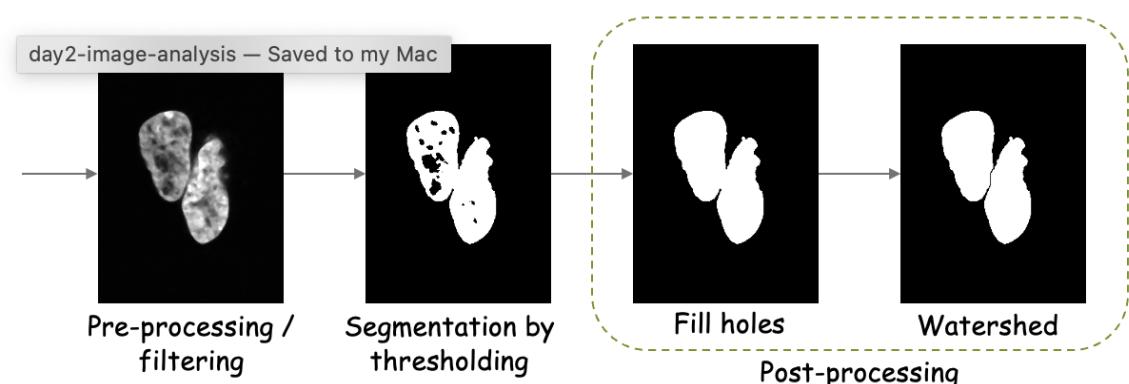
- **How to Batch Process a Folder of Images**

- ✓ Record your macro steps for one image
- ✓ Generalize it using loops and folder paths
- ✓ Official guide: <https://imagej.net/ij/developer/macro/macros.html>

- **Exercise:**

- ✓ Implement a macro that batch process the nuclei segmentation for the pipeline below.
- ✓ Tip: You can run the [visualize_jcp_image.ijm](#) macro to visualize a sample of the JUMP-CP dataset and check the code as an example

Use folder: `../data_day2/JUMP-CP_mini/Brefeldin-A-like/Orig/DNA`



Citing Fiji



ImageJ and variants



Schneider, C. A., Rasband, W. S., & Eliceiri, K. W. (2012). NIH Image to ImageJ: 25 years of image analysis. *Nature Methods*, 9(7), 671–675.
[doi:10.1038/nmeth.2089](https://doi.org/10.1038/nmeth.2089)

ImageJ



Rueden, C. T., Schindelin, J., Hiner, M. C., DeZonia, B. E., Walter, A. E., Arena, E. T., & Eliceiri, K. W. (2017). ImageJ2: ImageJ for the next generation of scientific image data. *BMC Bioinformatics*, 18(1). [doi:10.1186/s12859-017-1934-z](https://doi.org/10.1186/s12859-017-1934-z)

ImageJ2



Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., ... Cardona, A. (2012). Fiji: an open-source platform for biological-image analysis. *Nature Methods*, 9(7), 676–682. [doi:10.1038/nmeth.2019](https://doi.org/10.1038/nmeth.2019)

Fiji

Citable software

The following table lists all citable software packages, plugins, etc., documented on the site. To add a tool to this list, add the for its publication to the tool's wiki page.

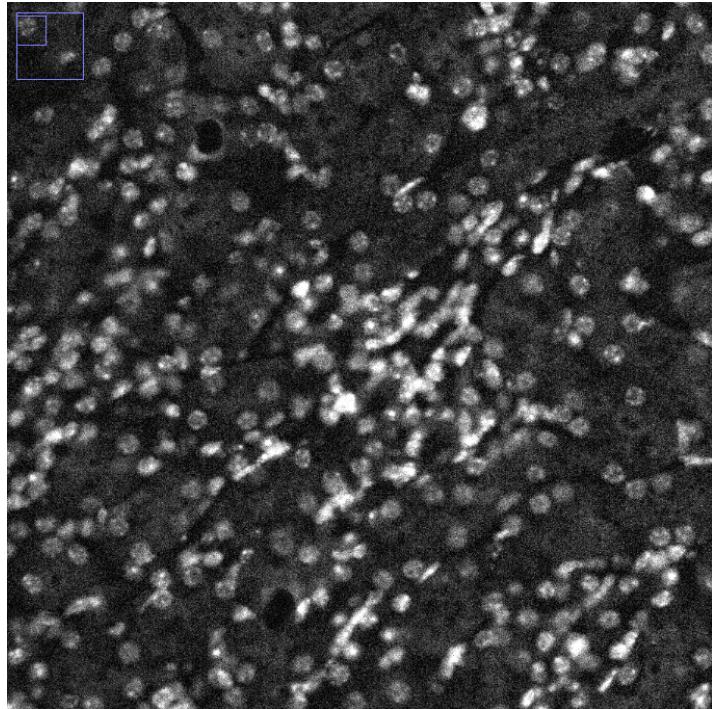
Project	Publication(s) to cite
Bio-Formats	Linkert, M., Rueden, C. T., Allan, C., Burel, J.-M., Moore, W., Patterson, A., ... Swedlow, J. R. (2010). Metadata matters: access to image data in the real world. <i>Journal of Cell Biology</i> , 189(5), 777–782. doi:10.1083/jcb.201004104
Flavors of ImageJ	Schindelin, J., Rueden, C. T., Hiner, M. C., & Eliceiri, K. W. (2015). The ImageJ ecosystem: An open platform for biomedical image analysis. <i>Molecular Reproduction and Development</i> , 82(7–8), 518–529. doi:10.1002/mrd.22489
IJ-OpenCV	Domínguez, C., Heras, J., & Pascual, V. (2017). IJ-OpenCV: Combining ImageJ and OpenCV for processing images in biomedicine. <i>Computers in Biology and Medicine</i> , 84, 189–194. doi:10.1016/j.combiomed.2017.03.027
ImageJ Ops	Rueden, C., Dietz, C., Horn, M., Schindelin, J., Northan, B., Berthold, M. & Eliceiri, K. (2021). ImageJ Ops [Software]. https://imagej.net/Ops .

Deep Learning-based workflows for segmentation

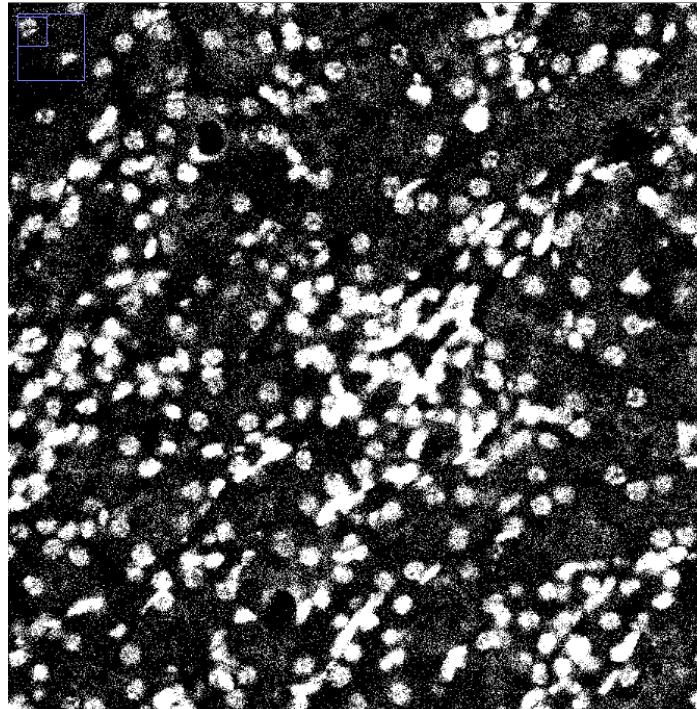
When thresholding fails



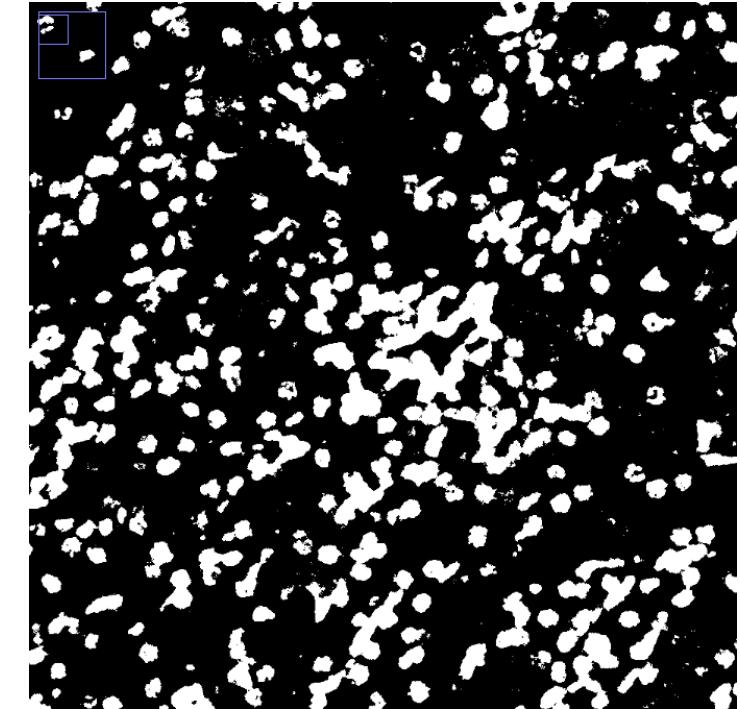
Original



Filtered



Thresholded



- Uneven illumination → poor thresholding
- Touching/overlapping cells → incorrect separation

When thresholding fails

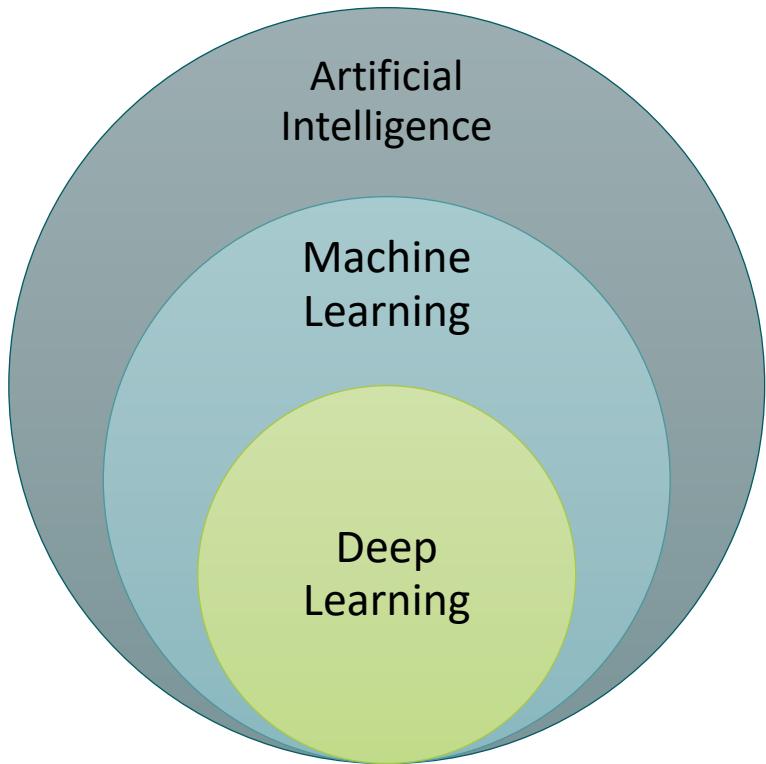


- We Need More Robust Methods
 - Better generalization across datasets

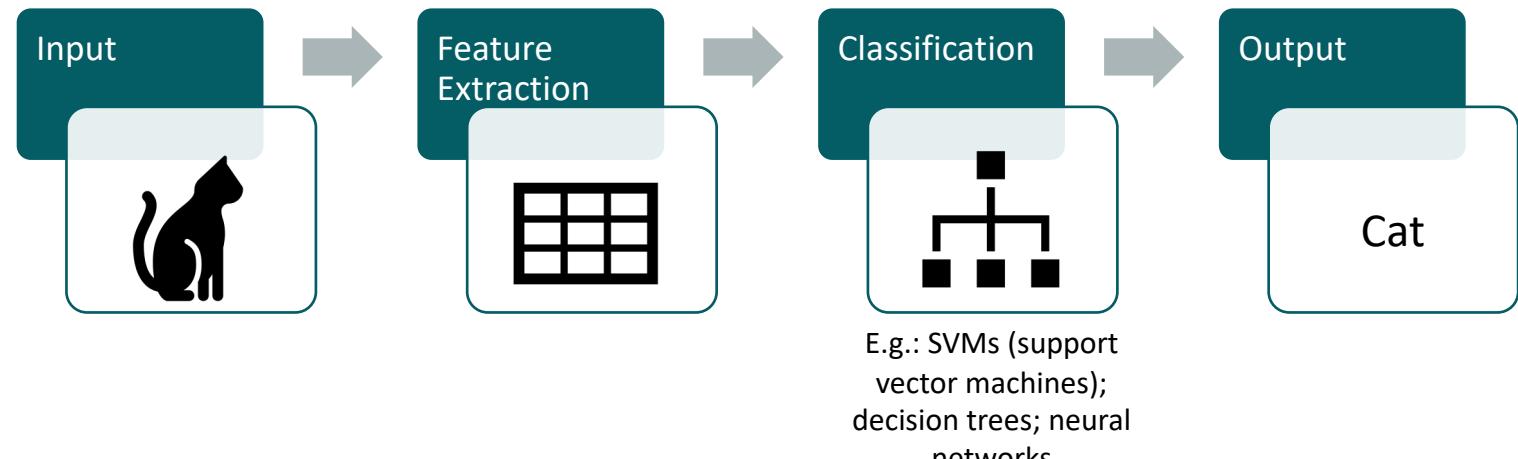


- We need models that learn what makes a nucleus a nucleus, regardless of brightness, shape, or surrounding conditions.

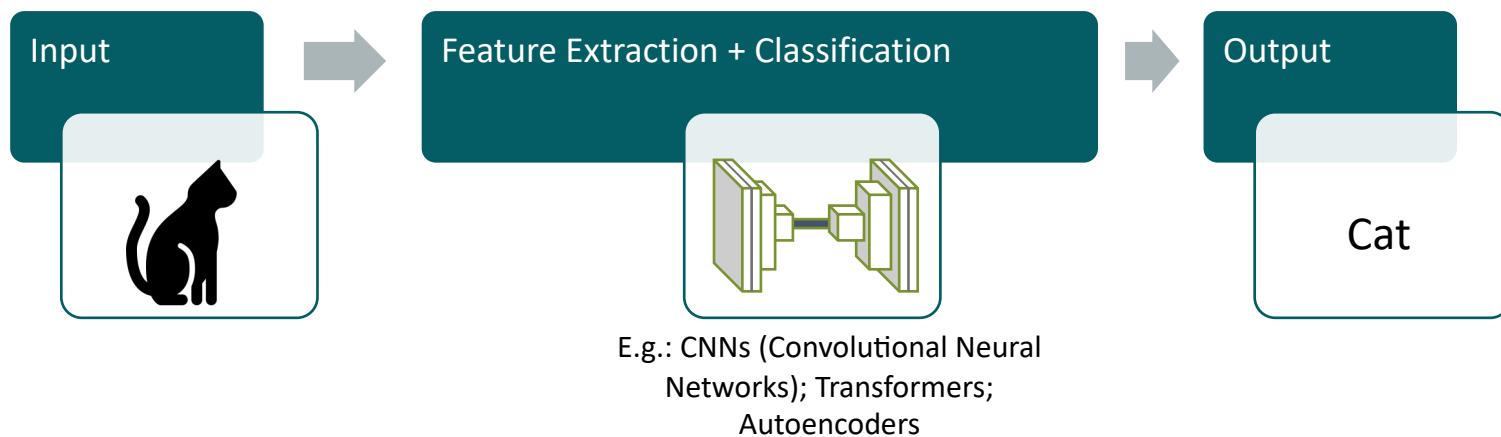
Learning from data itself



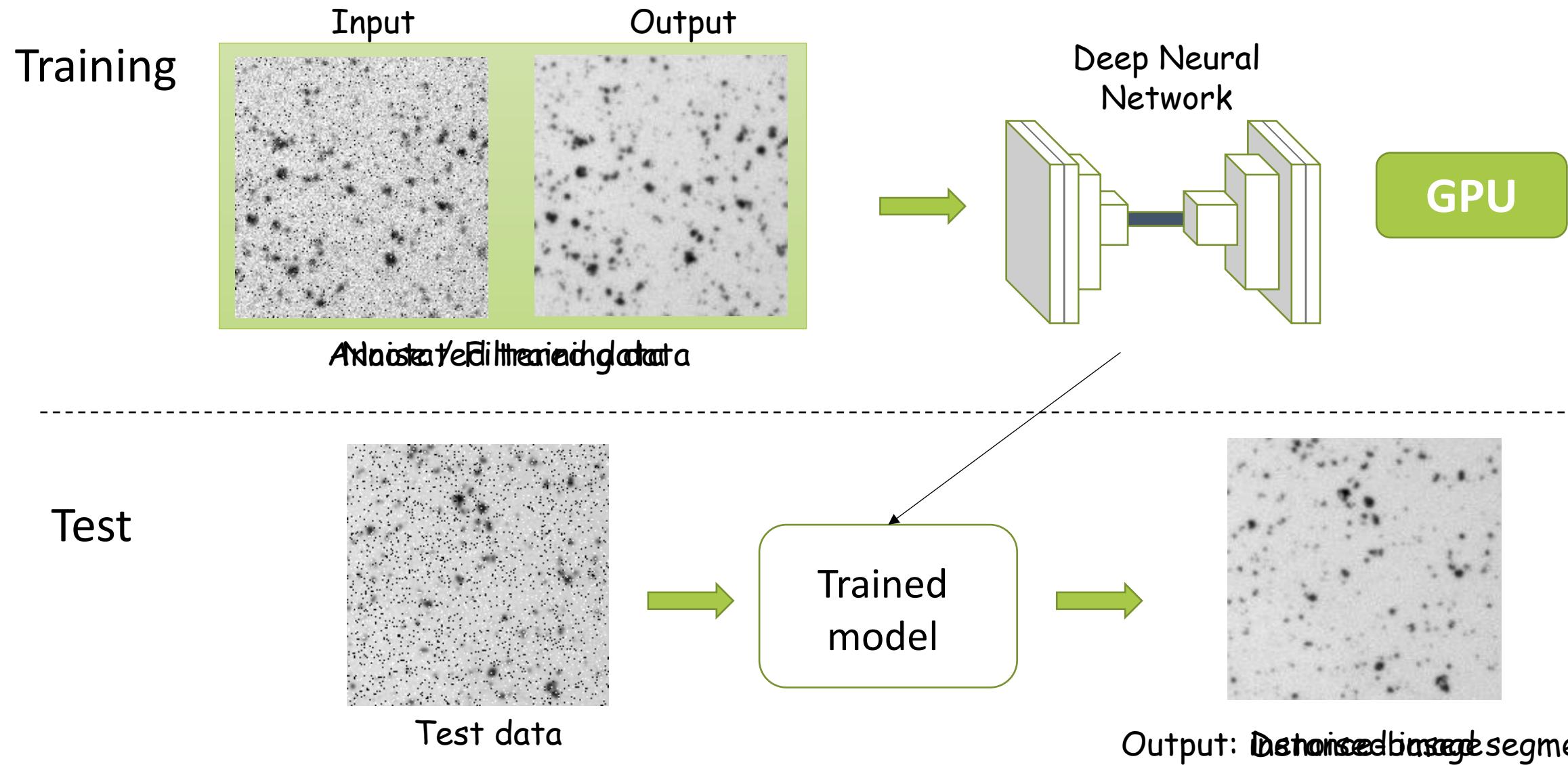
Machine Learning



Deep Learning



Deep learning-based segmentation



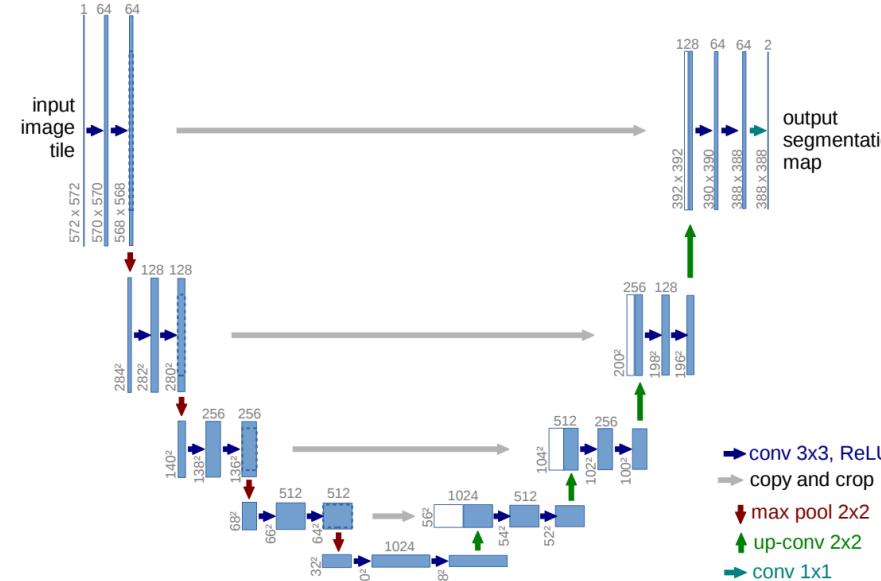
U-Net



- A convolutional neural network designed for biomedical image segmentation.
 - ✓ Built with an encoder-decoder structure and skip connections.
 - ✓ Learns to map an input image to a pixel-wise segmentation mask.

Encoder:

- Convolutions and downsampling
- Learns edges, shapes, texture



Decoder

- Reconstruction by upsampling and convolutions
- Pixel-wise class prediction (cell vs background)

Skip connections

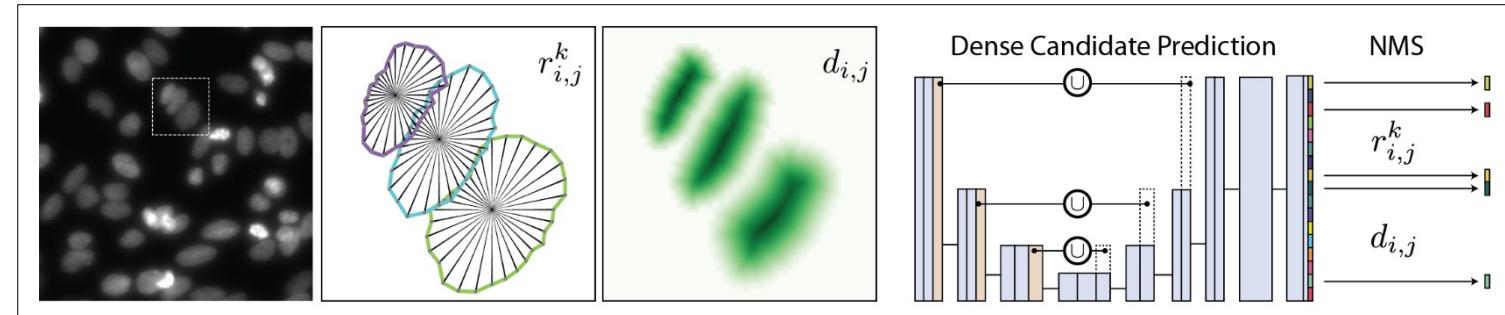
Bottleneck: high level features

StarDist: Star-convex shaped nuclei detection



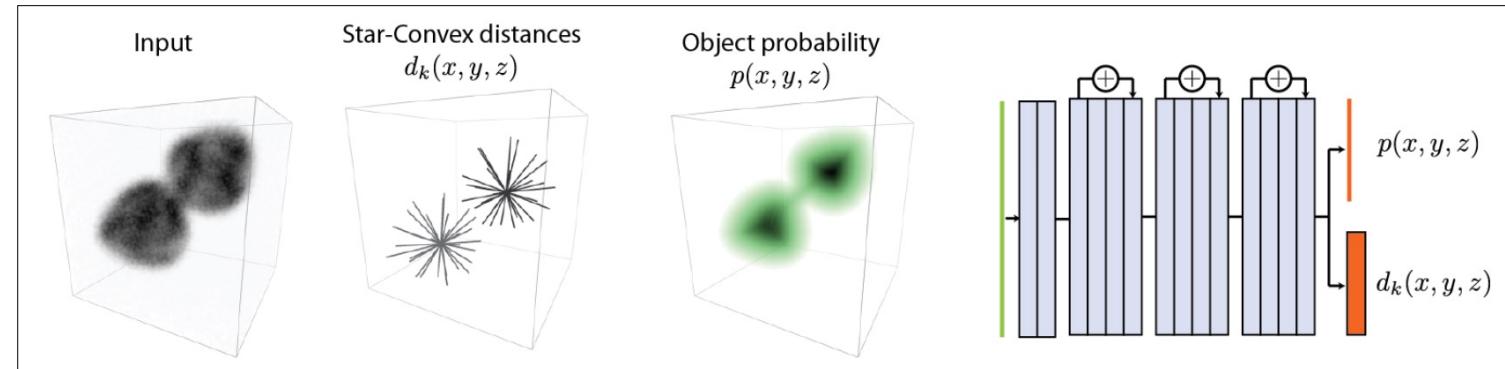
- Models a shape by predicting distances r_{ij}^k from the center of each pixel to the boundary along a fixed set of rays (usually 32 or more).
- Approximates each object with a polygon (2D) or polyhedron (3D) centered around each pixel.

How far the boundary is in multiple directions (rays)



2D

And how likely that pixel belongs to an object.



3D

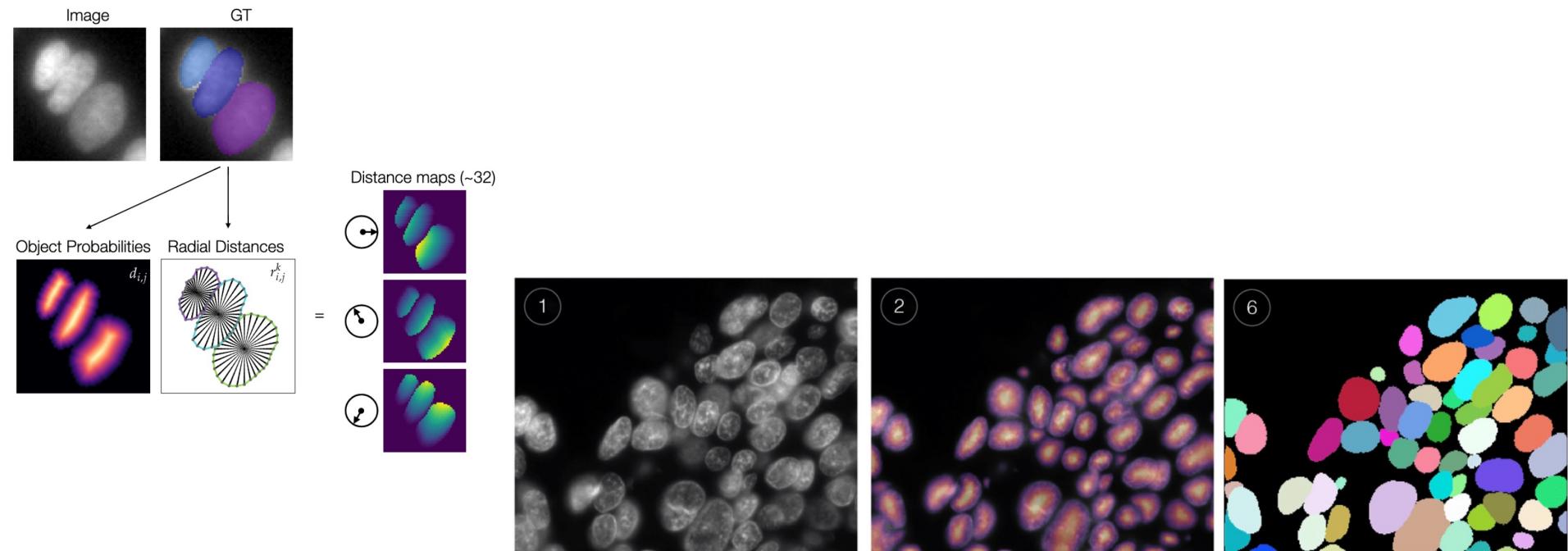
StarDist: Star-convex shaped nuclei detection



- Generates many overlapping star-convex shape candidates from the predicted distances and probabilities for all pixels.
- To reduce redundancy, it applies NMS(Non-Maximum Suppression) to keep only the best, non-overlapping object predictions.

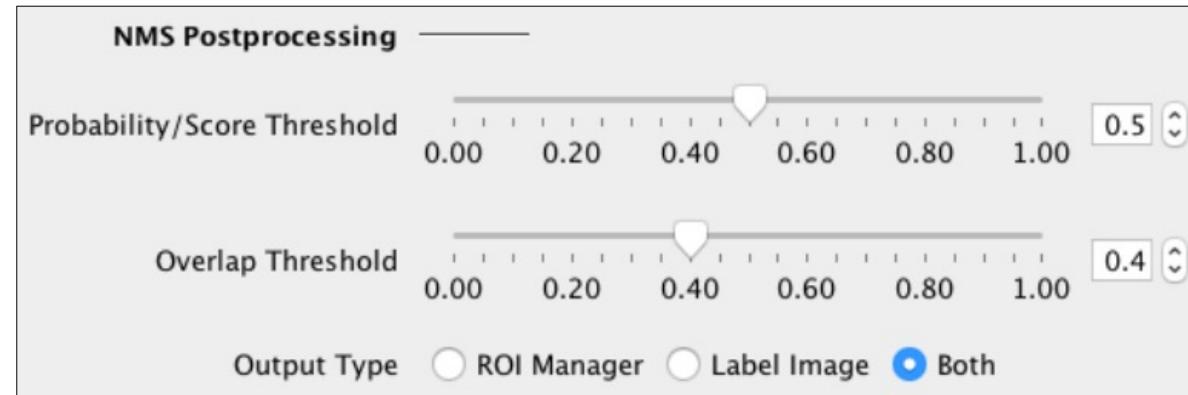
It keeps the best shape prediction and removes overlapping, lower-confidence ones.

Clean segmentation map, no overlaps, only the most confident detections.



StarDist: Star-convex shaped nuclei detection

- Fiji plugin:
 - **Probability/Score Threshold** - higher values lead to fewer segmented objects, but will likely avoid false positives.
 - **Overlap Threshold** - higher values allow segmented objects to overlap substantially.



Adjusting the NMS

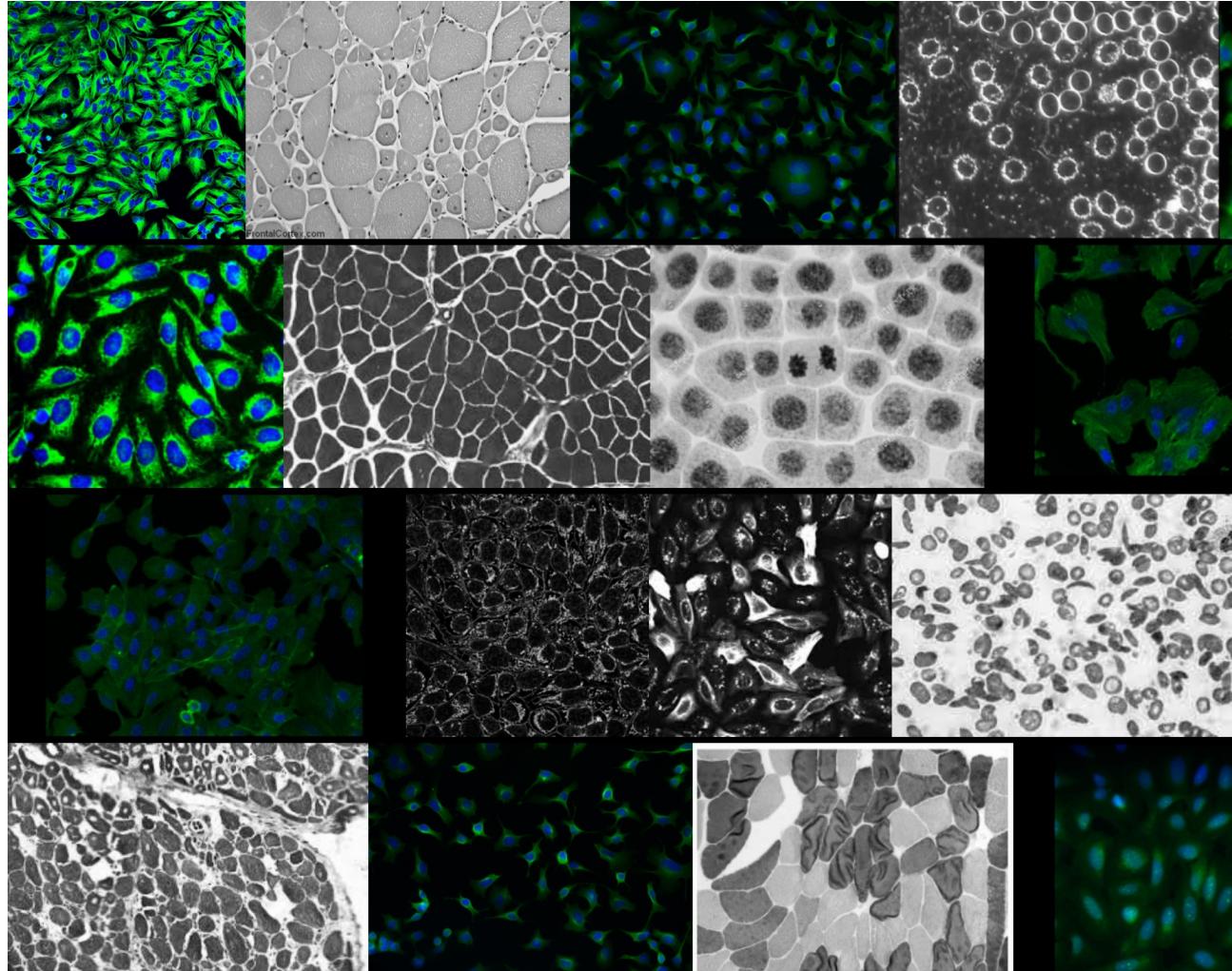
CellPose: general cell segmentation



✓ Generalist algorithm:

trained on a large dataset of cell images from a variety of microscopy modalities and fluorescent markers

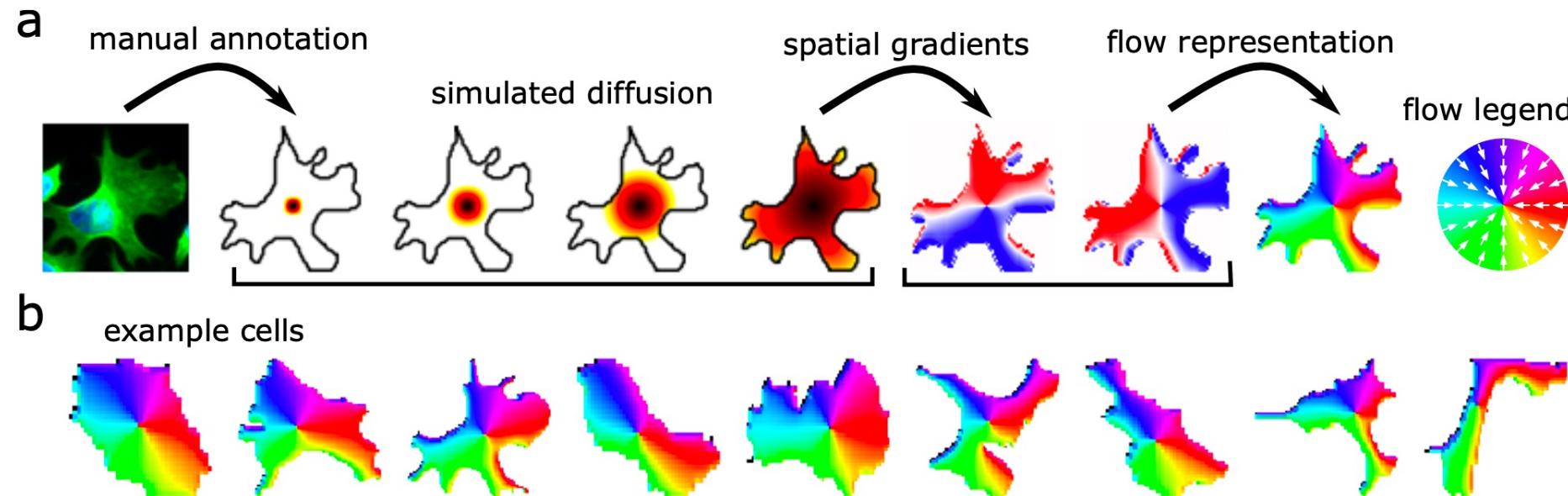
- Inspired by the 2018 *DataBowl challenge* (nuclei segmentation challenge)
- Stringer, C., Wang, T., Michaelos, M. *et al.* Cellpose: a generalist algorithm for cellular segmentation. *Nat Methods* **18**, 100–106 (2021)



CellPose: general cell segmentation



- ✓ For each mask: simulations of a heat-diffusion process with a heat source at the center of the mask

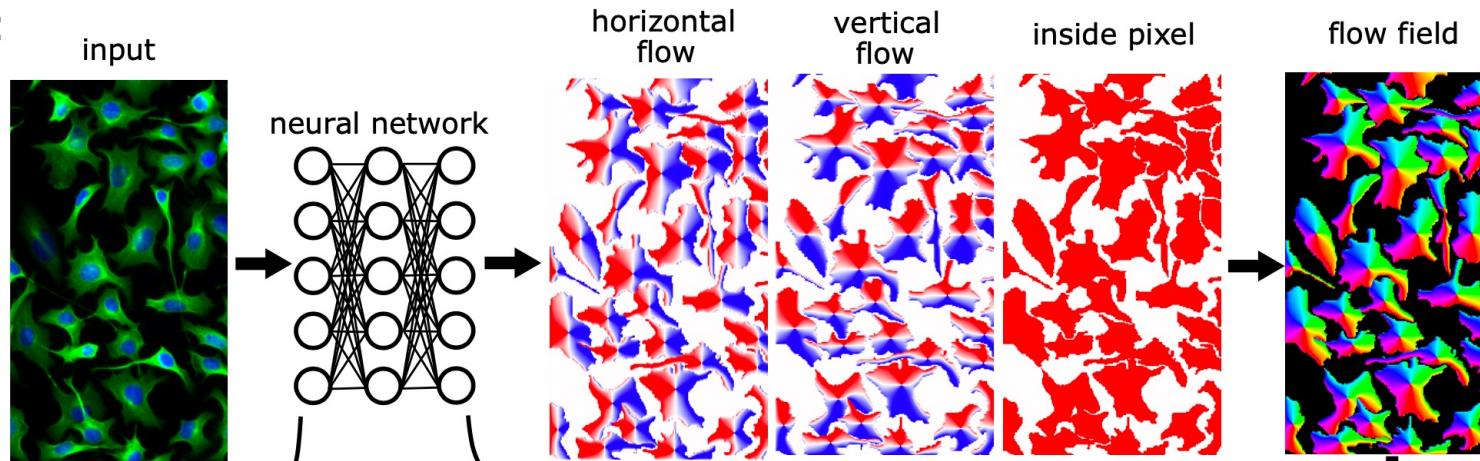


- ✓ The X and Y gradients are combined into a single normalized direction from 0 to 360 degrees

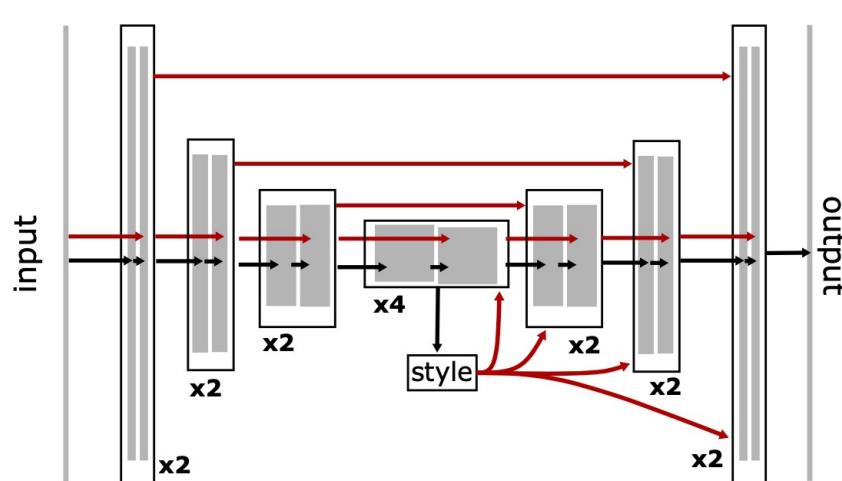
CellPose: general cell segmentation



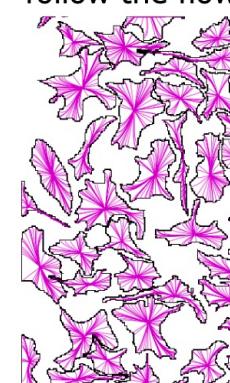
C



d



e "follow the flows"



f



A neural network is trained to predict the horizontal and vertical flows, as well as whether a pixel belongs to any cell. The three predicted maps are combined into a flow field.

CellPose is trained to learn how to grow a cell from the center, making it especially good at segmenting irregular or overlapping shapes.



cellpose

a generalist algorithm for cellular segmentation



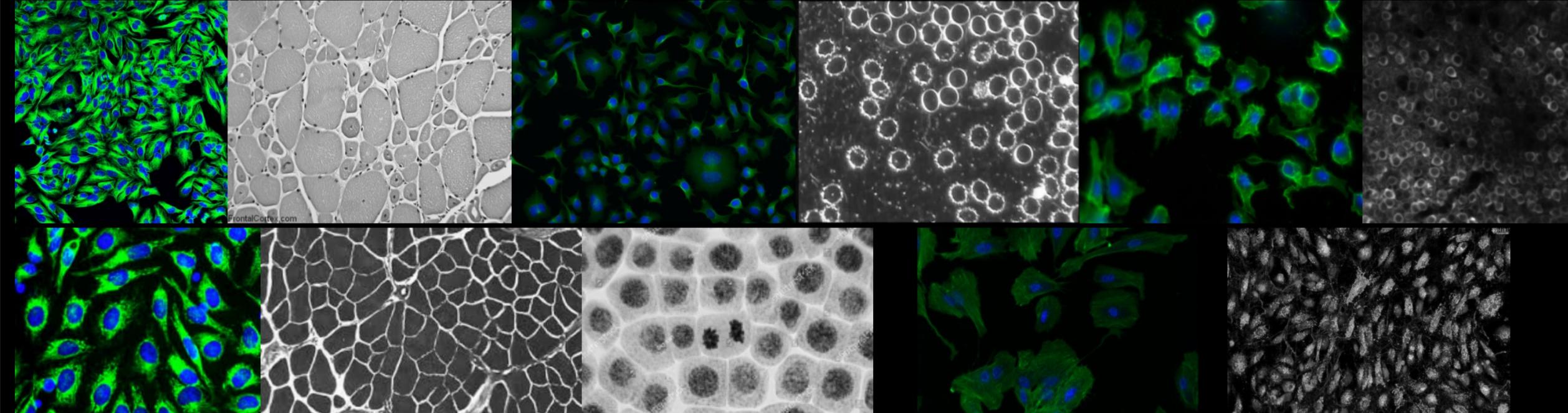
<http://www.cellpose.org>

Check out full documentation [here](#). For software advice, check out our topic on [image.sc](#)
Try cellpose by uploading one PNG or JPG <10 MB. Images are resized to a max size of **512x512** pixels.

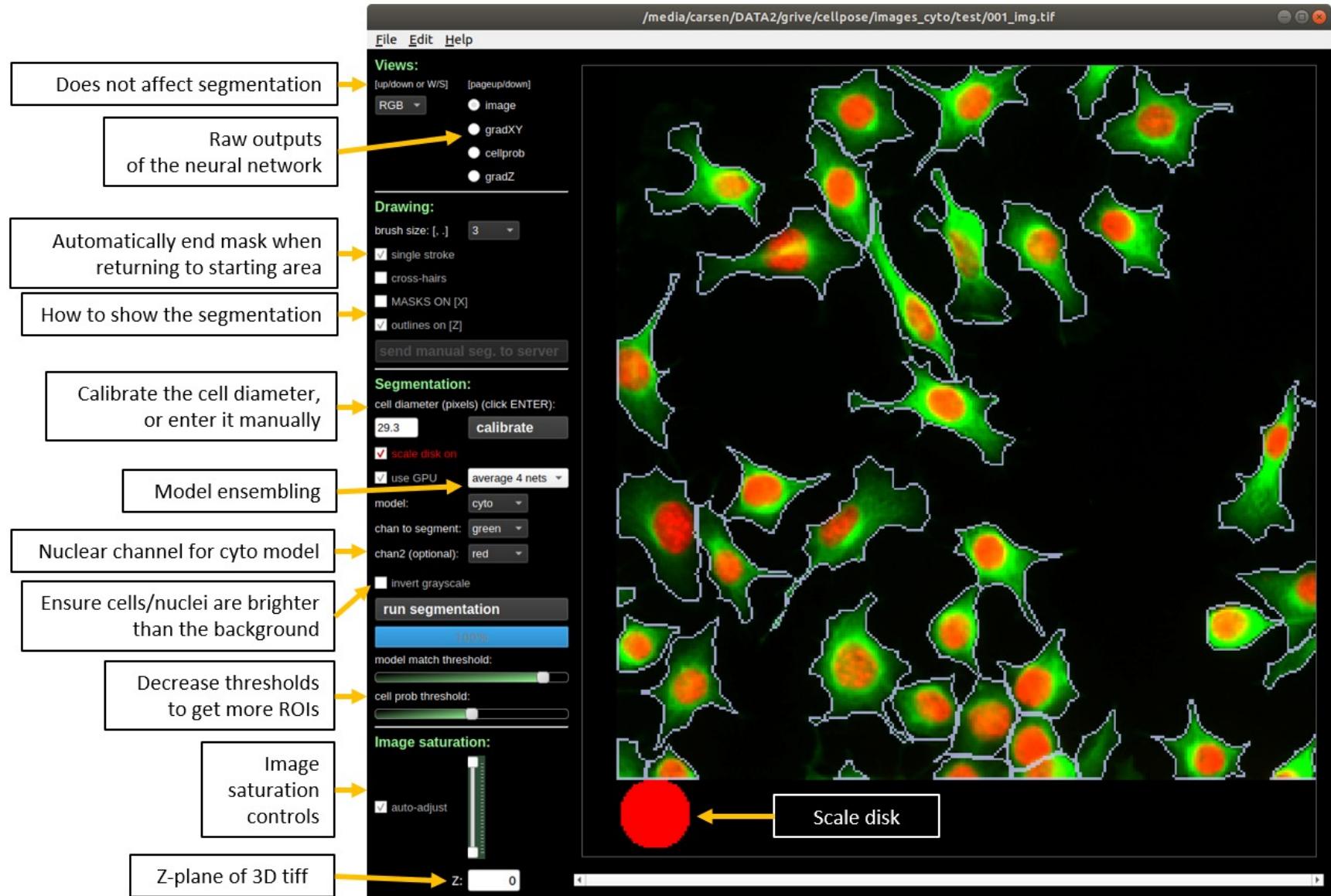


Drop files here or click to upload.

or click on an example image from our test set:



CellPose GUI



How to install

README BSD-3-Clause license

Cellpose

<https://github.com/MouseLand/cellpose>

docs passing tests passing codecov 49% pypi package 4.0.6 downloads 1M
downloads/month 43k python 3 license BSD-3-Clause contributors 62 website up
forum 559 topics repo size 116.6 MiB Stars 1.8k Forks 502

Cellpose-SAM: cell and nucleus segmentation with superhuman generalization. It can be optimized for your own data, applied in 3D, works on images with shot noise, (an)isotropic blur, undersampling, contrast inversions, regardless of channel order and object sizes.

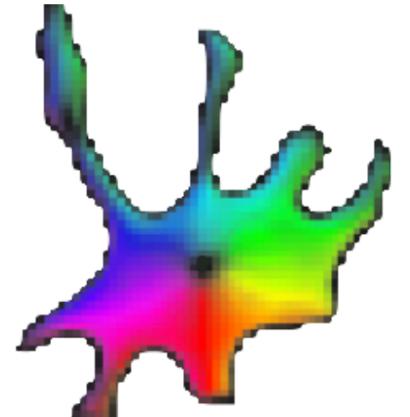
To learn about Cellpose-SAM read the [paper](#) or watch the [talk](#). For info on fine-tuning a model, watch this [tutorial talk](#), and see this example [video](#) of human-in-the-loop training. For support, please open an [issue](#).

Please see install instructions [below](#), and also check out the detailed documentation at [cellpose.readthedocs.io](#). The Cellpose-SAM [website](#) allows batch processing of images with a free account on Hugging Face.

Example notebooks:

- [run_Cellpose-SAM.ipynb](#) Open in Colab : run Cellpose-SAM on your own data, mounted in google drive
- [test_Cellpose-SAM.ipynb](#) Open in Colab : shows running Cellpose-SAM using example data in 2D and 3D
- [train_Cellpose-SAM.ipynb](#) Open in Colab : train Cellpose-SAM on your own labeled data (also optional example data provided)

► The Cellpose-SAM model is trained on data that is licensed under CC-BY-NC. The Cellpose annotated dataset is also CC-BY-NC.





Parameters

✓ Channels

1. *Cytoplasm model*: trained on two-channel images, 1) channel to segment, and 2) optional nuclear channel
2. *Nuclei model*: trained on two-channel images, 1) channel to segment, and 2) the second channel is always set to an array of zeros

✓ Diameter: needs a user-defined cell diameter or a size estimation

- Cellpose models have been trained on images rescaled to all have the same diameter (30 pixels for the *cyto* model and 17 pixels for the *nuclei* model).
- Estimation: size prediction based using the linear regression model from the style vector



Parameters

- ✓ Flow threshold: maximum allowed error of the flows for each mask.
 - Default value is 0.4. Should be increased if Cellpose is not returning as many masks as expected, or, decreased if Cellpose is returning too many ill-shaped masks.

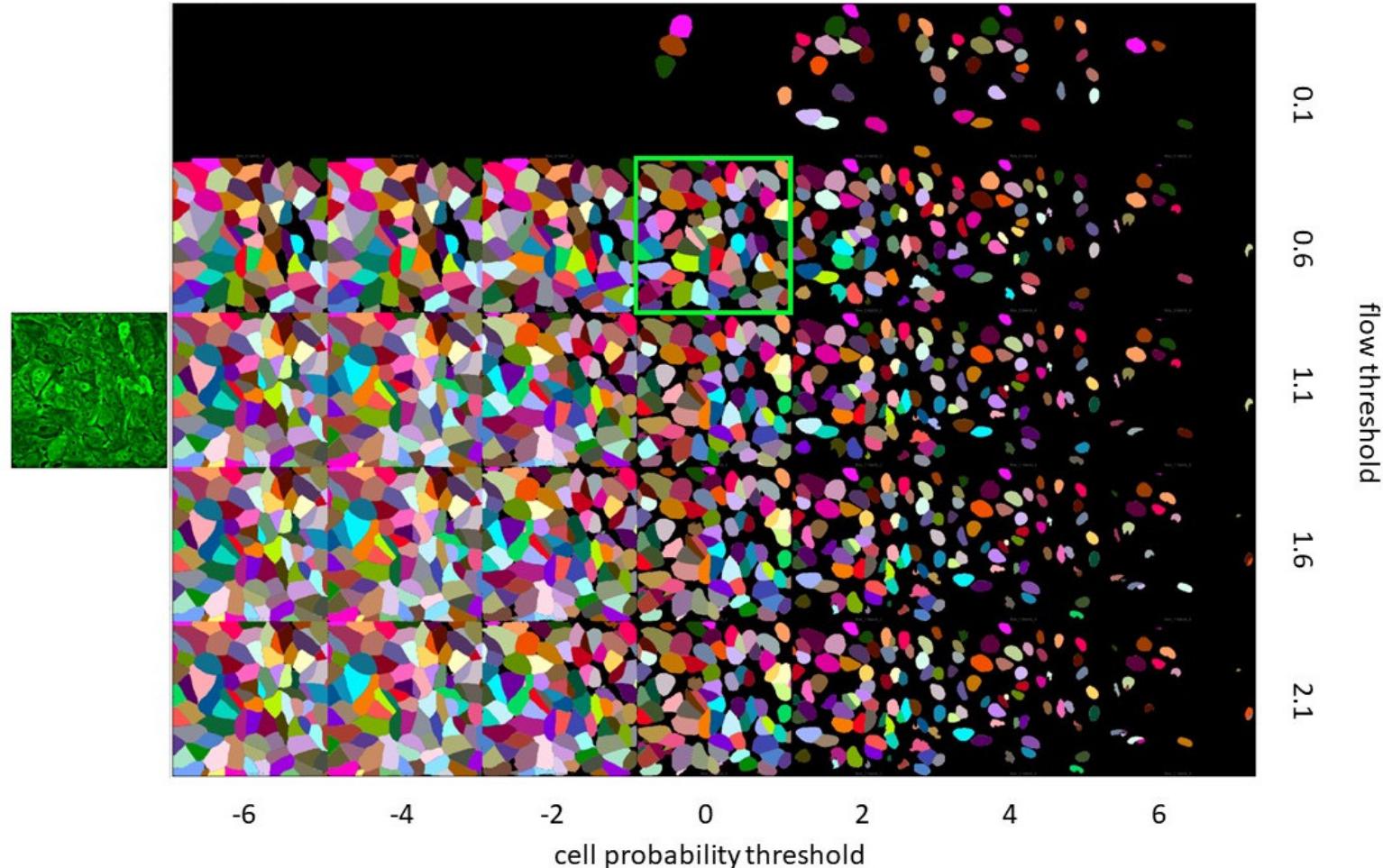
- ✓ Cell probability threshold: used to run dynamics and determine masks.
 - The default is `cellprob_threshold=0.0`. Should be increased if Cellpose is returning too many masks particularly from dim areas.

flow_threshold: controls how strong the vector flows must be to count as an object. i.e. higher = fewer but higher-confidence masks

cellprob_threshold: minimum probability for a pixel to be considered part of a cell



Parameters

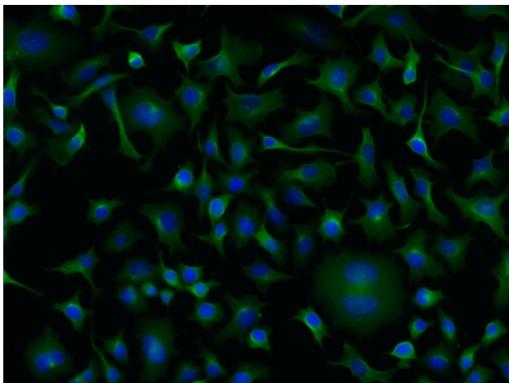


Output

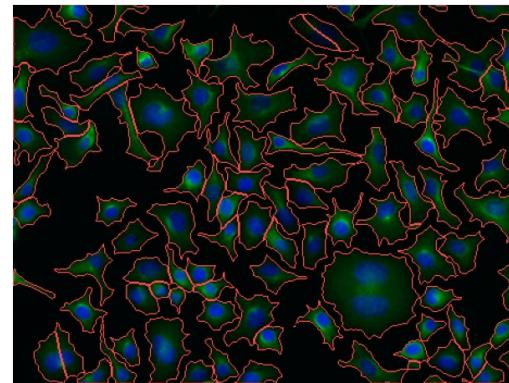


- ✓ Predicted outlines and masks
- ✓ Outlines as text for Fiji

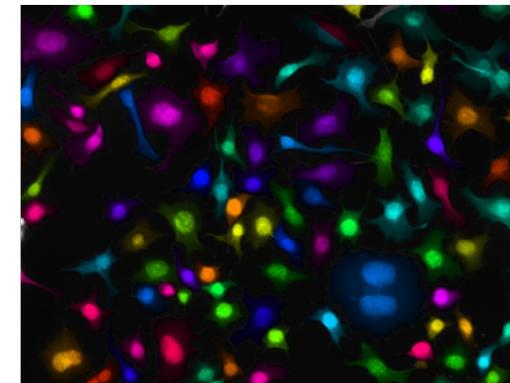
original image



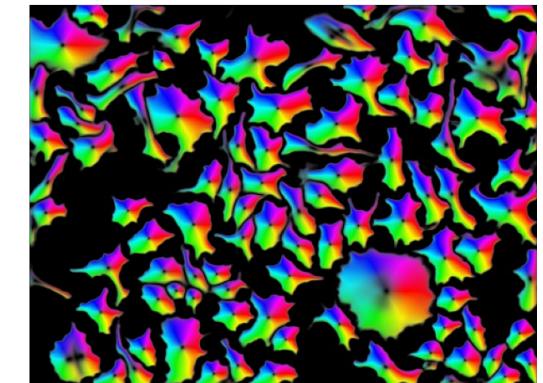
predicted outlines



predicted masks



predicted cell pose



LabelsToROIs

A Fiji/ImageJ plugin to generate ROIs from label images, allowing ROI erosion and quantification

<https://labelstorois.github.io/>

[View on GitHub](#)

[Download .zip](#)

[Download .tar.gz](#)

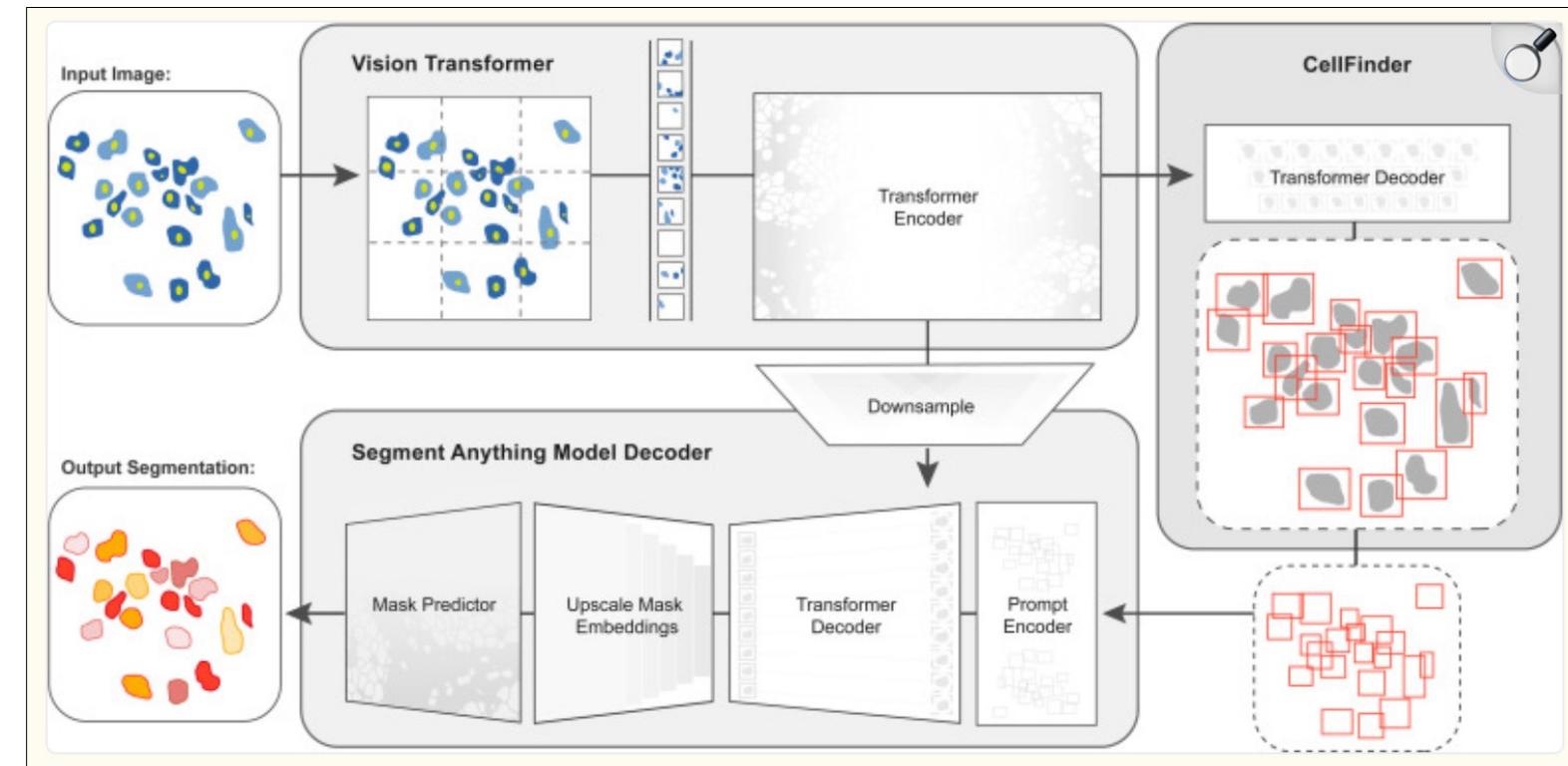
CellSAM



- Zero-shot, promptable instance segmentation model for cell and tissue images.
- It builds upon Meta's Segment Anything Model (SAM), tailored to bioimages

*Fine-tunes the image encoder and decoder on **cell images**.*

Uses biomedical annotations for training.

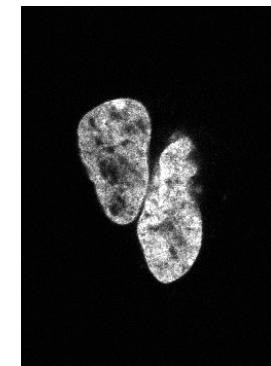


Coming back to our workflow...

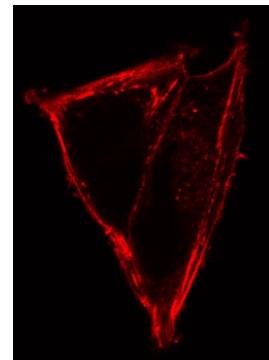
Get outlines of nuclei
from DAPI image



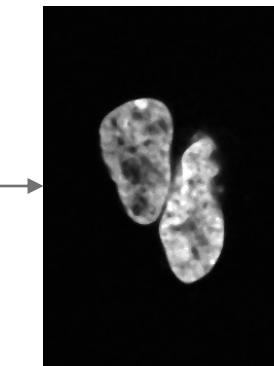
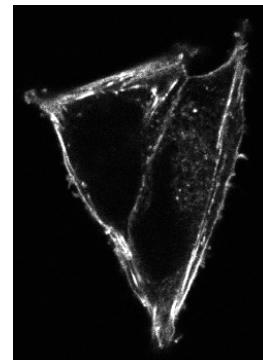
Nuclei



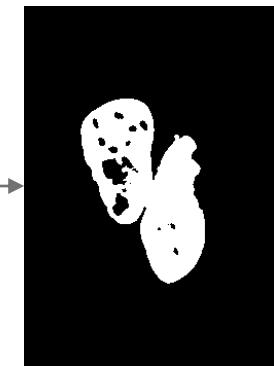
Get outlines of cytoplasm
from actin image



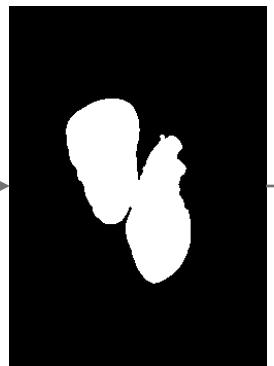
Actin



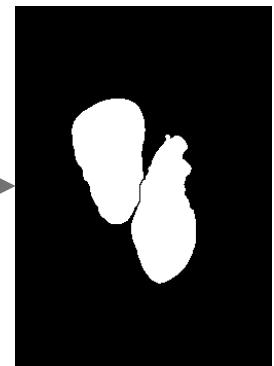
Pre-processing /
filtering



Segmentation by
thresholding



Fill holes



Watershed

Post-processing

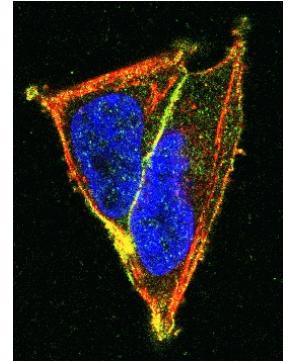
Pre-processing /
edge enhancement

Pre-processing /
filtering

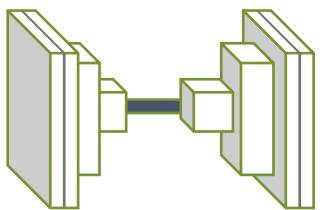
Segmentation by
thresholding

- Is it good?
- Can we post-process it?

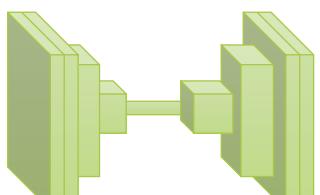
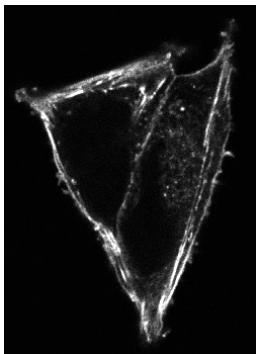
Cell and nuclei segmentation: a deep-learning approach



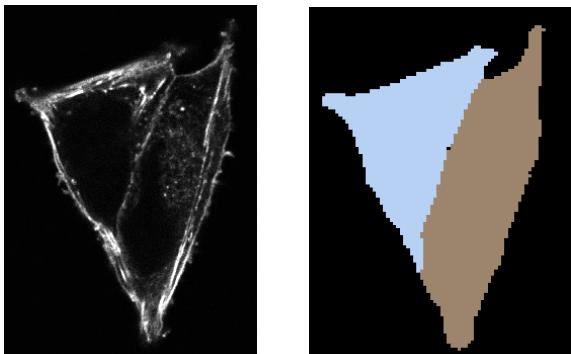
Test image



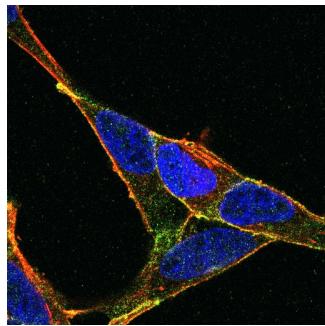
Trained model
Nuclei segmentation



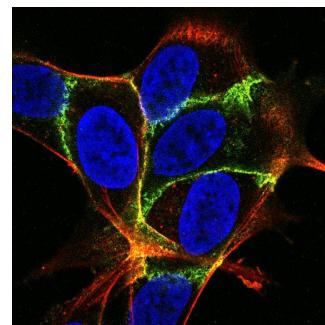
Trained model
Cell segmentaiton



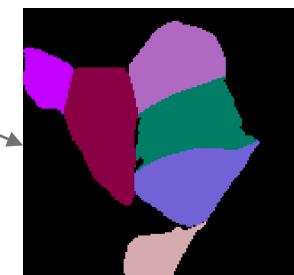
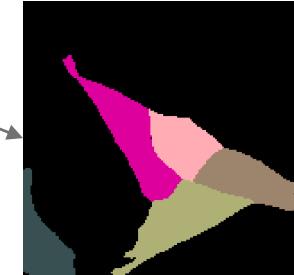
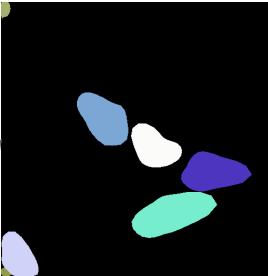
Other examples



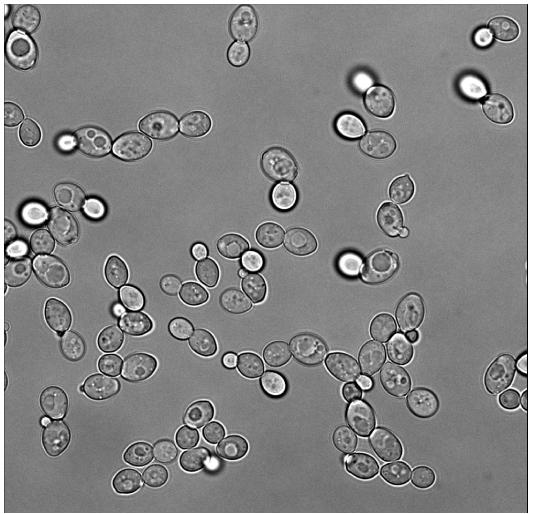
Test image



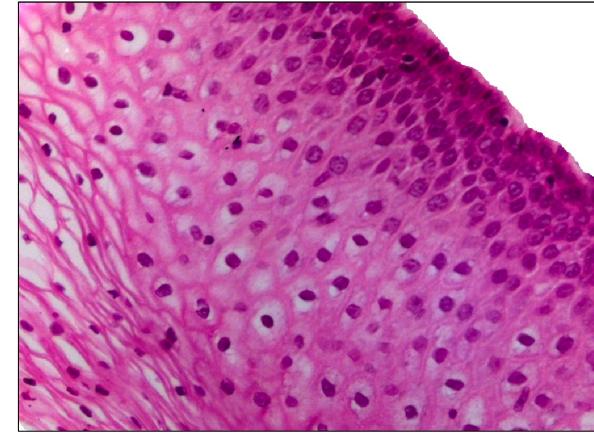
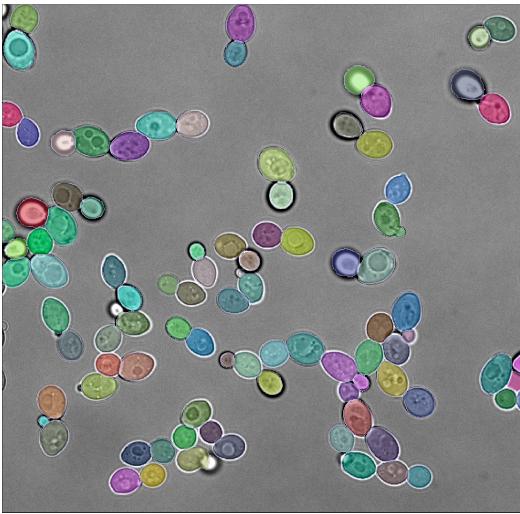
Test image



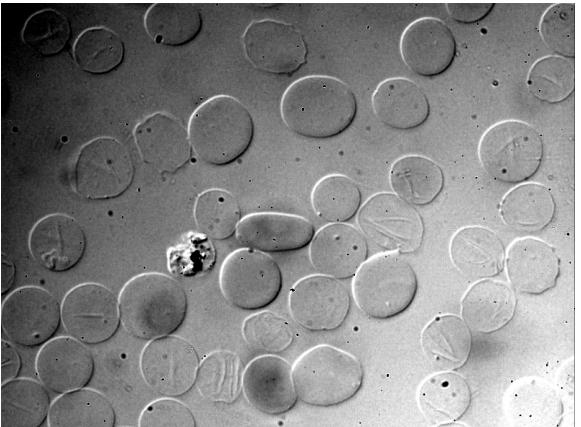
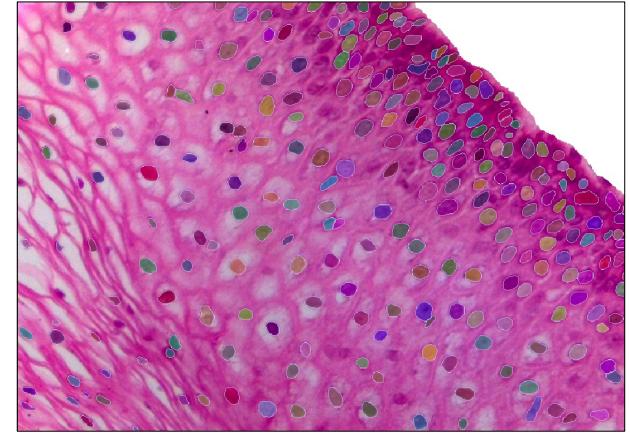
CellPose: instance-based segmentation



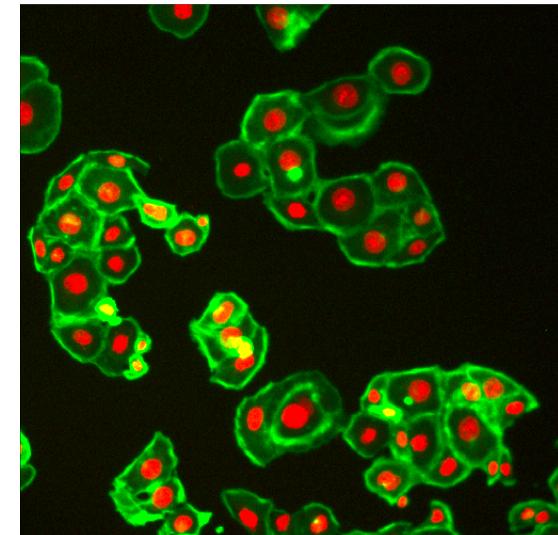
Yeast



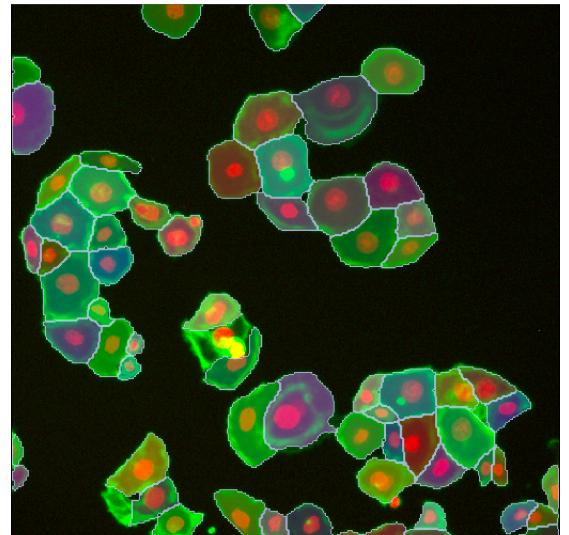
H&E stained tissue



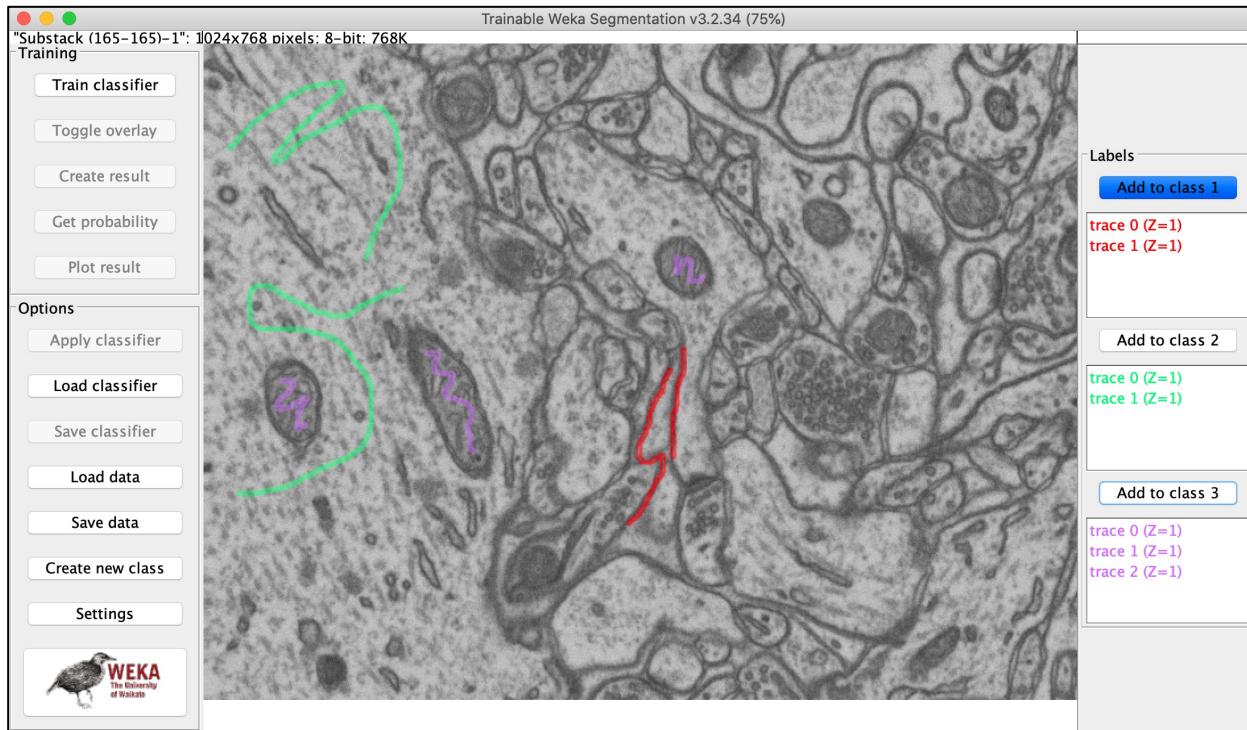
Red blood cells



Fluorescent cells



Machine learning-based segmentation



- ✓ Pixel-based classification
 - Trainable Weka segmentation
 - Ilastik

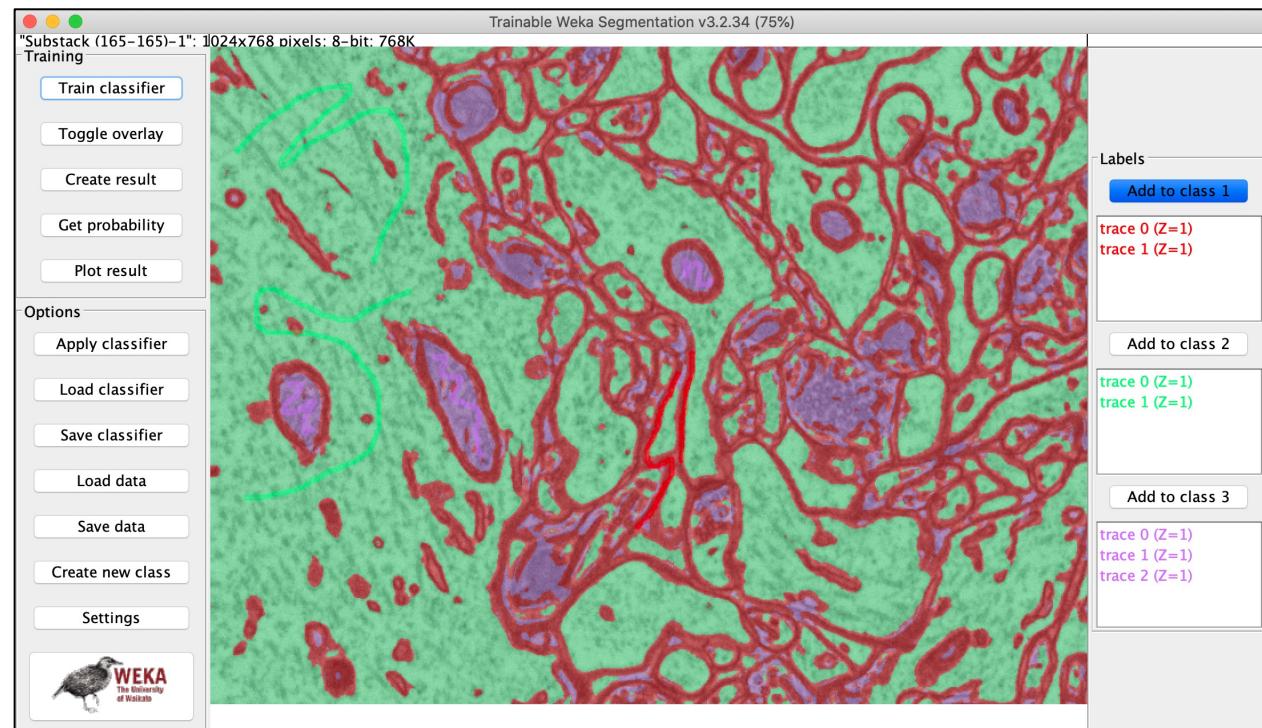


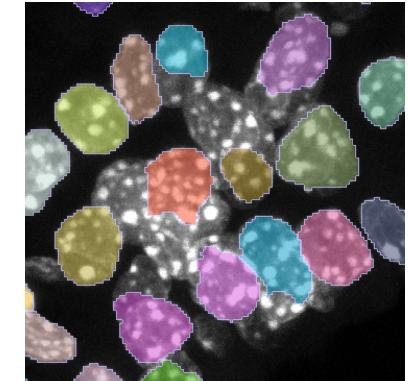
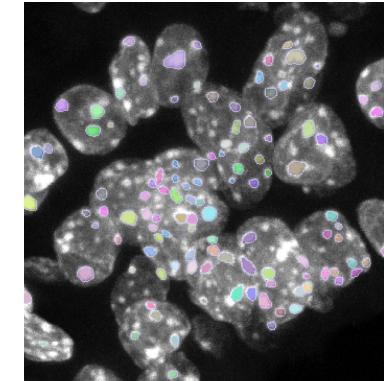
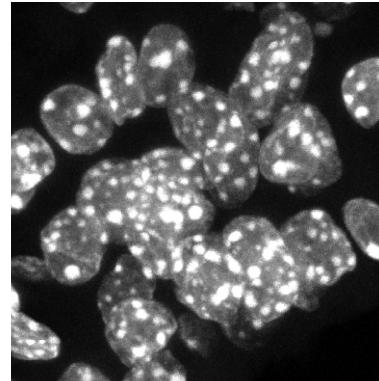
Image source: <https://www.epfl.ch/labs/cvlab/software/biomedical/biomedplugins/>

https://imagej.net/Trainable_Weka_Segmentation

Why do we still use classical approaches



- DL generalization problem



- Lack of annotated training data or small datasets
- Image analysis is not limited to cell/nuclei segmentation

Practical Use + Resources



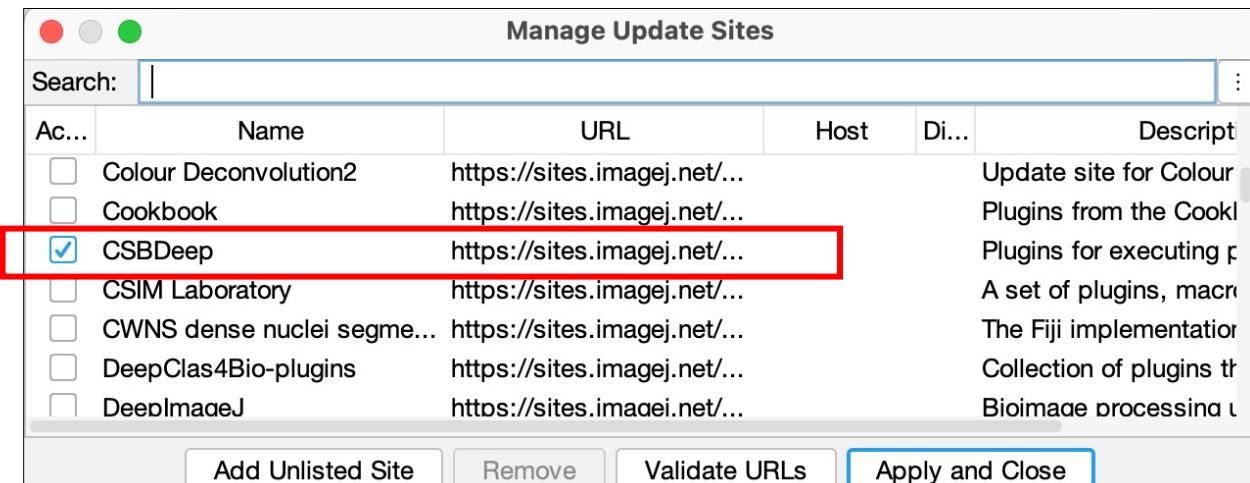
- Installation and environments
 - Google Colab
 - Fiji
 - Napari plugins: <https://napari.org/stable/>
 - Ilastik: <https://www.ilastik.org/>
- Links to:
 - BioImage.IO: <https://bioimage.io/#/models>
 - DL4Microscopy: <https://github.com/HenriquesLab/ZeroCostDL4Mic>
 - Hugging Face: <https://huggingface.co/>

Practical Exercises



□ Install the **StarDist** plugin for Fiji:

- Go to Help / Update... / Manage update sites.
- Select **CSBDeep**, **StarDist** and **Tensorflow** as shown in the figures below
- Click on “Close” and then “Apply changes”.
- You will be requested to restart Fiji.
- More info on: <https://imagej.net/plugins/stardist>



Practical Exercises



- Choose any image from any DNA folder:
 1. Open the chosen image in Fiji
 2. Open the StarDist plugin in the “Plugins” menu
 3. Set StarDist parameters
 4. Apply (click “OK”)
 5. Check if parameters need to be adjusted. If so, restart from 1

 6. After you’ve selected the parameter values, use the data in the ROI Manager to generate the Results Table, like yesterday
 7. Try re-implementing the previous macro by replacing the thresholding segmentation with StarDist

Using Google Colab to run a Jupyter Notebook

Using Google Colab



1. Go to <https://colab.research.google.com/>
2. Click the “GitHub” tab
3. In the search box, paste the GitHub repository URL
4. Press Enter
5. Click on the notebook you want to open (.ipynb file)

Note: you can also use public notebooks without logging in, but for saving changes or using a GPU, log into your Google account