

Deep Learning for Bio-image Analysis

Robert Haase

Funded by



Bundesministerium
für Bildung
und Forschung

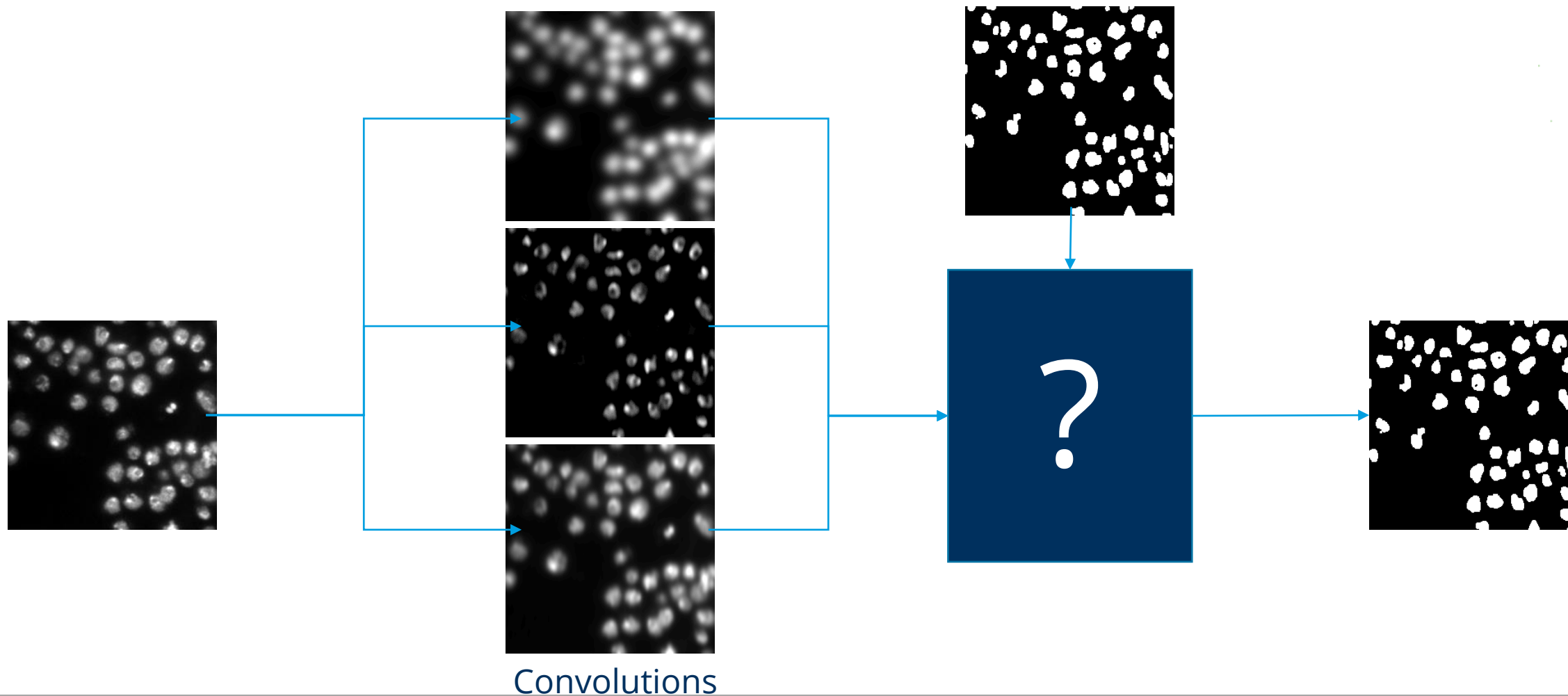
SACHSEN



Diese Maßnahme wird gefördert durch die Bundesregierung
aufgrund eines Beschlusses des Deutschen Bundestages.
Diese Maßnahme wird mitfinanziert durch Steuermittel auf
der Grundlage des von den Abgeordneten des Sächsischen
Landtags beschlossenen Haushaltes.

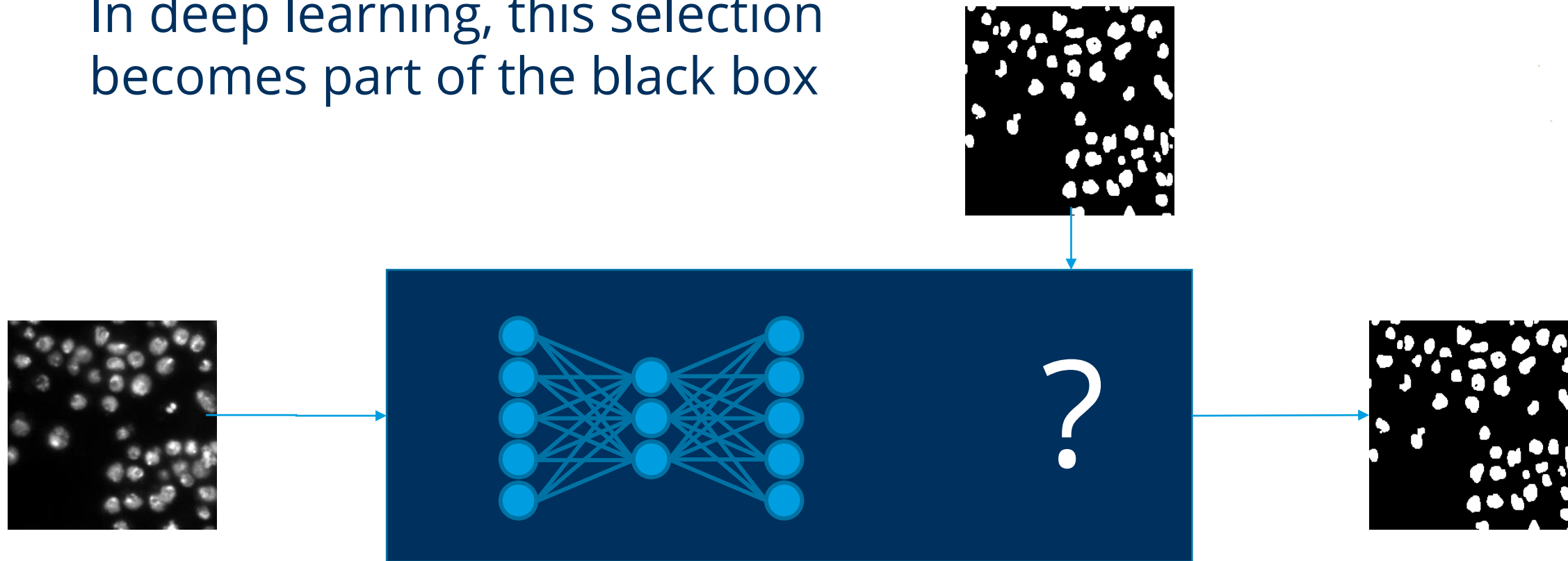
Machine learning for image analysis

In classical machine learning, we typically select features for training our classifier



Outlook: Deep learning for image analysis

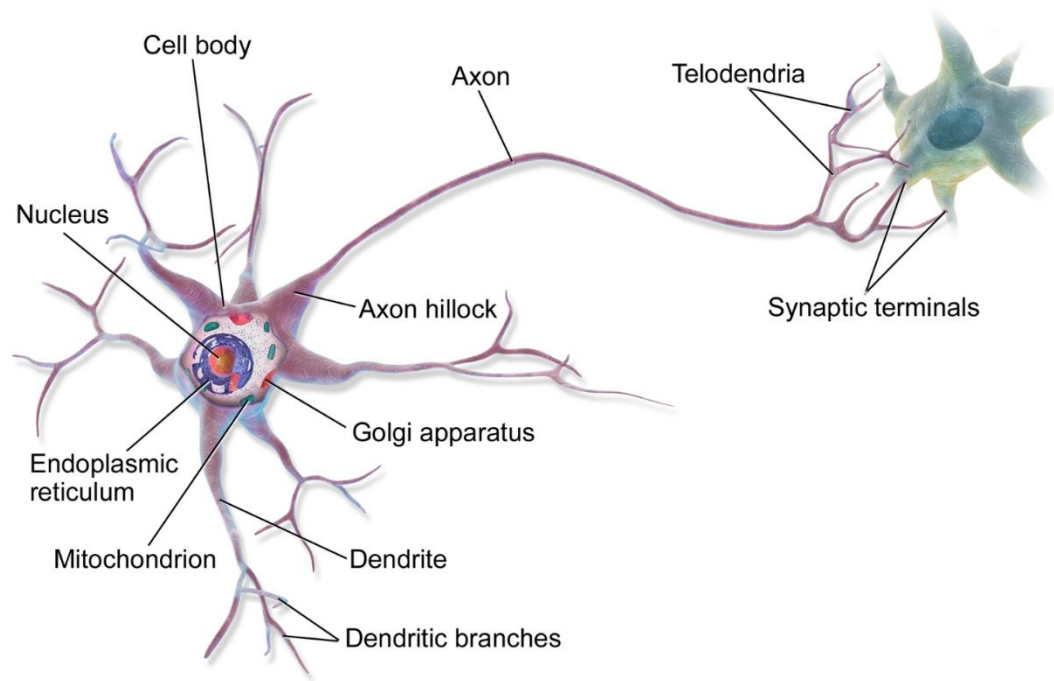
In deep learning, this selection becomes part of the black box



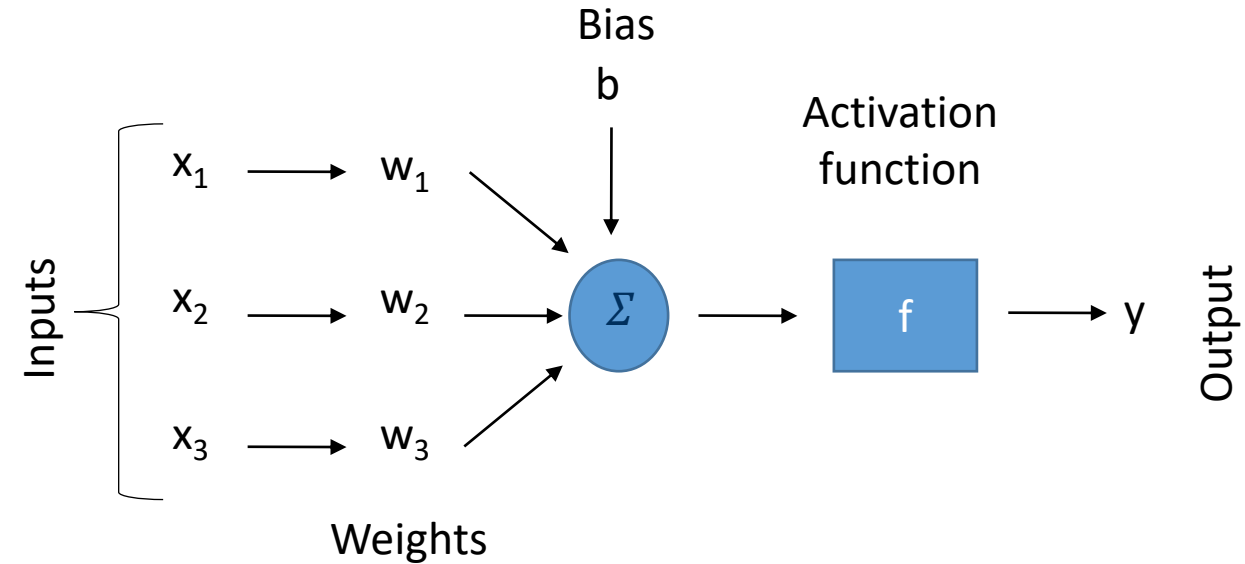
Convolutional neural networks

Neural networks

- How biologists see neurons



- How computer scientists see neurons
“perceptron”



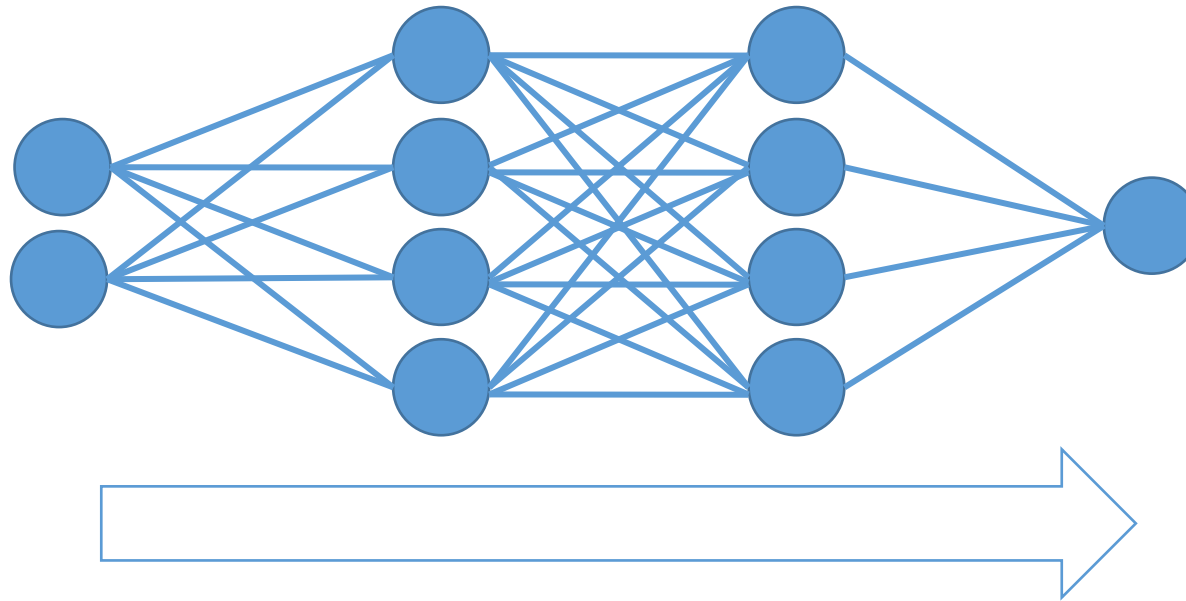
$$y = f(w_1x_1 + w_2x_2 + w_3x_3 + b)$$

Neural Networks

- Early form: “Multilayer Perceptron”
- fully connected class of feedforward artificial neural network

If there are *many* hidden layers, we speak of a *deep* neural network

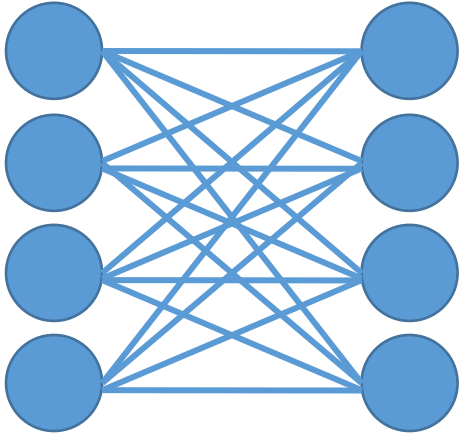
Input layer Hidden layer(s) Output layer



Convolutional neural networks

- Layer types

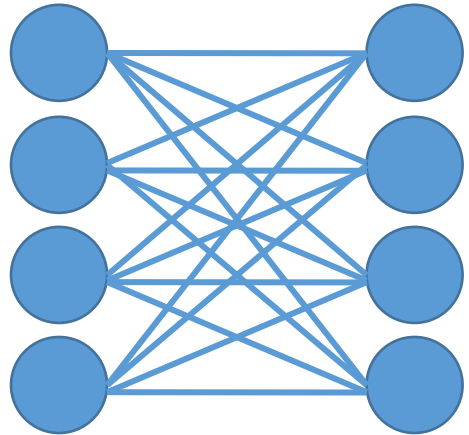
Fully connected layer



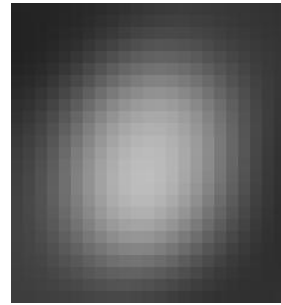
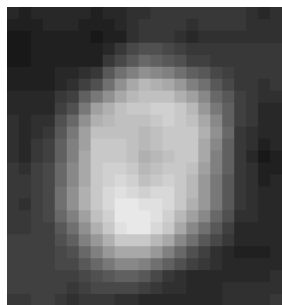
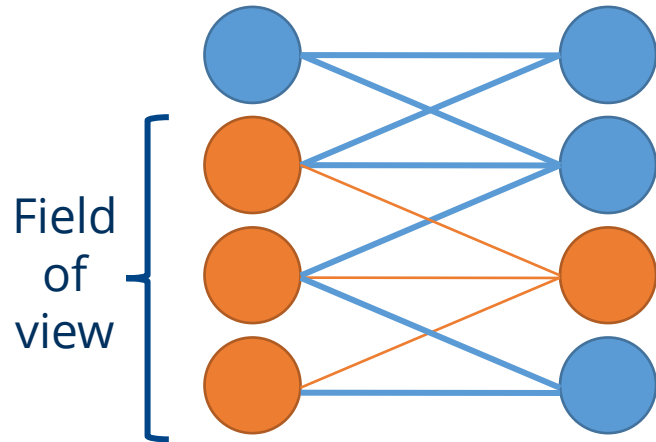
Convolutional neural networks

- Layer types

Fully connected layer



Convolutional layer



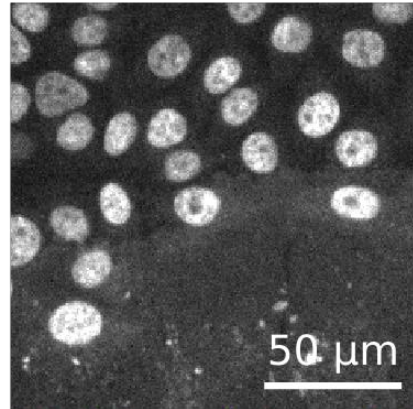
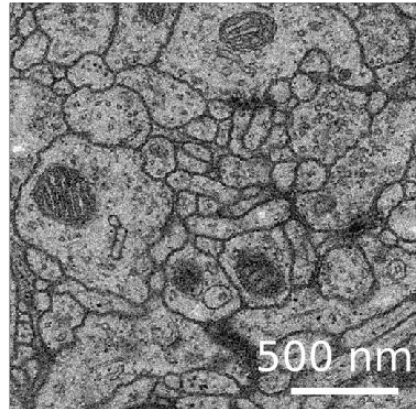
Deep Learning for Microscopy

Segmentation

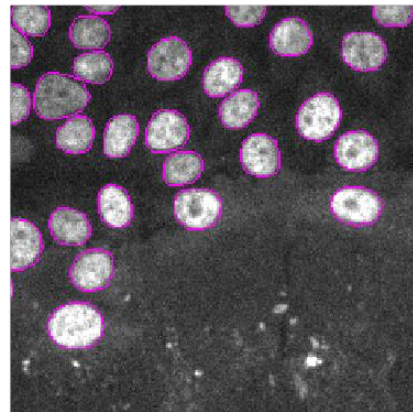
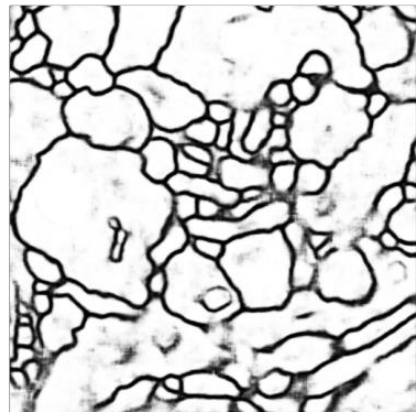
U-net

Stardist

Input



Output



Denoising

Noise2Void

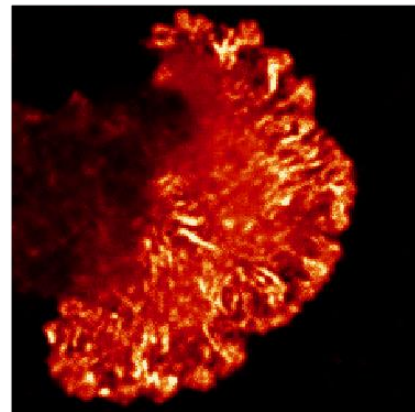
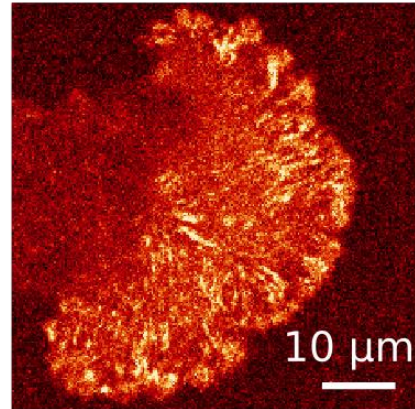
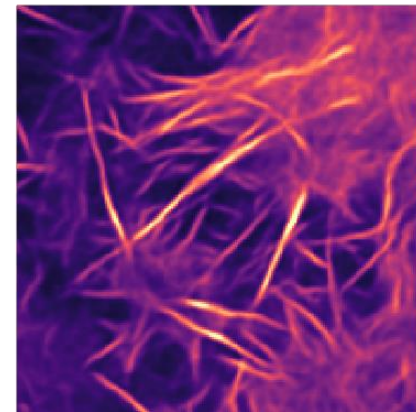
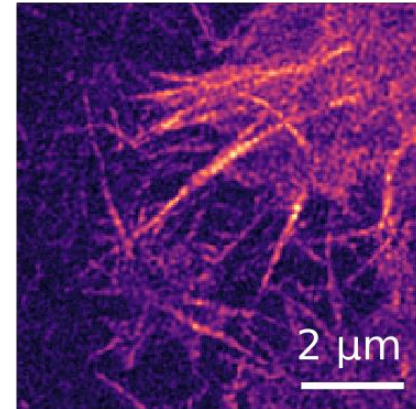


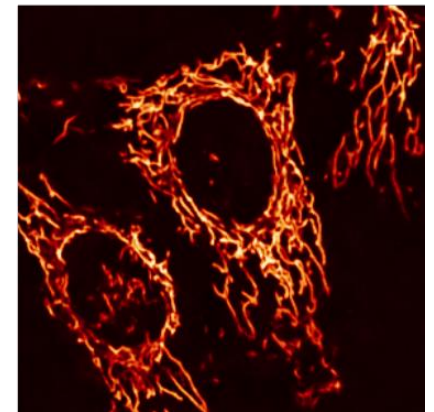
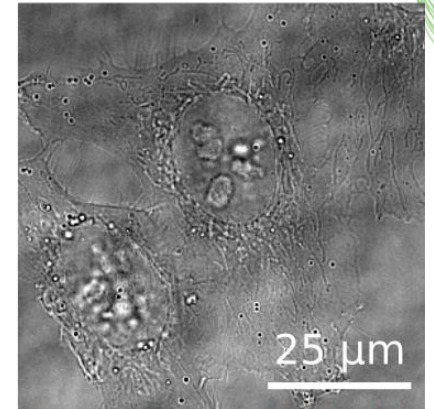
Image restoration

CARE

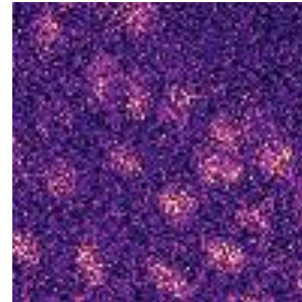
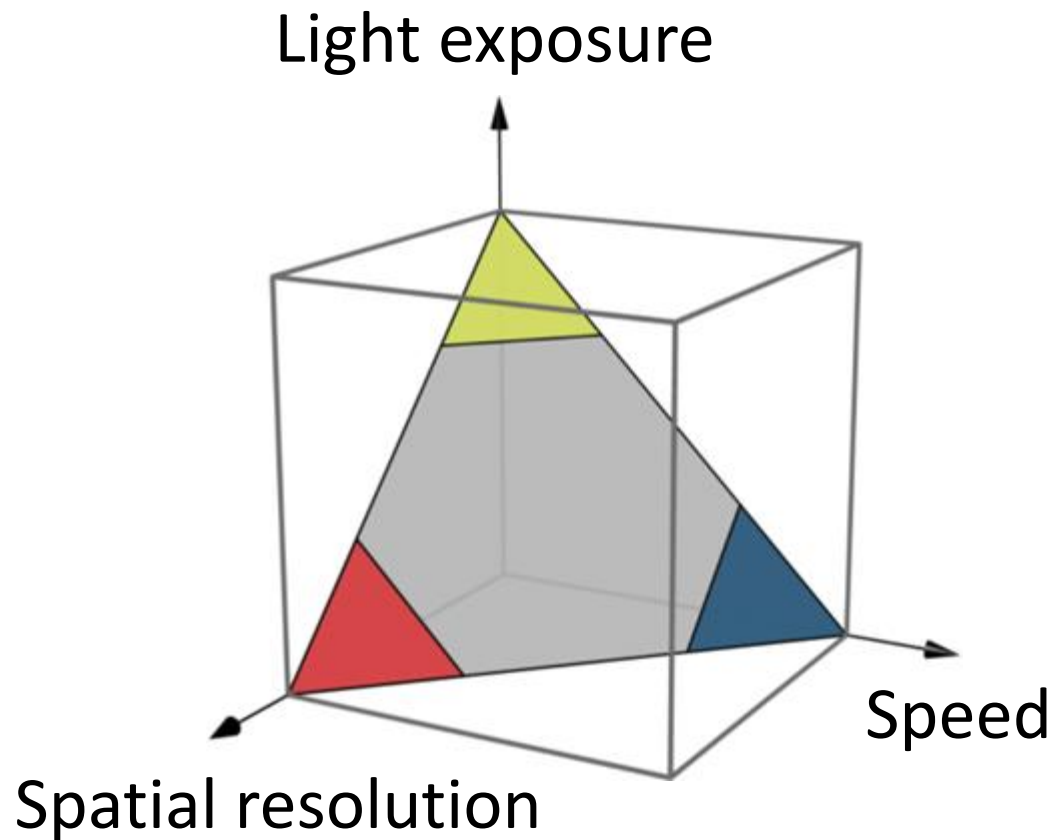


Artificial labelling

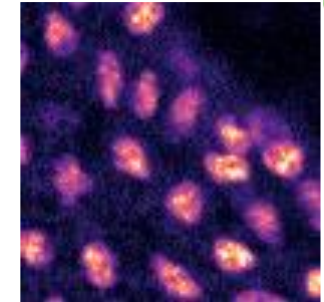
fnet



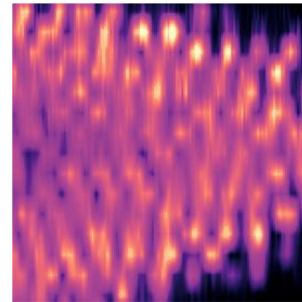
Trade-offs in live-cell imaging



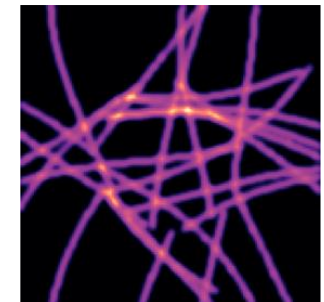
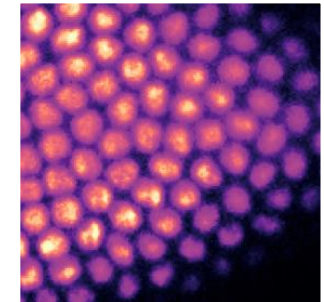
Light Exposure



Speed



Spatial Resolution



The U-net

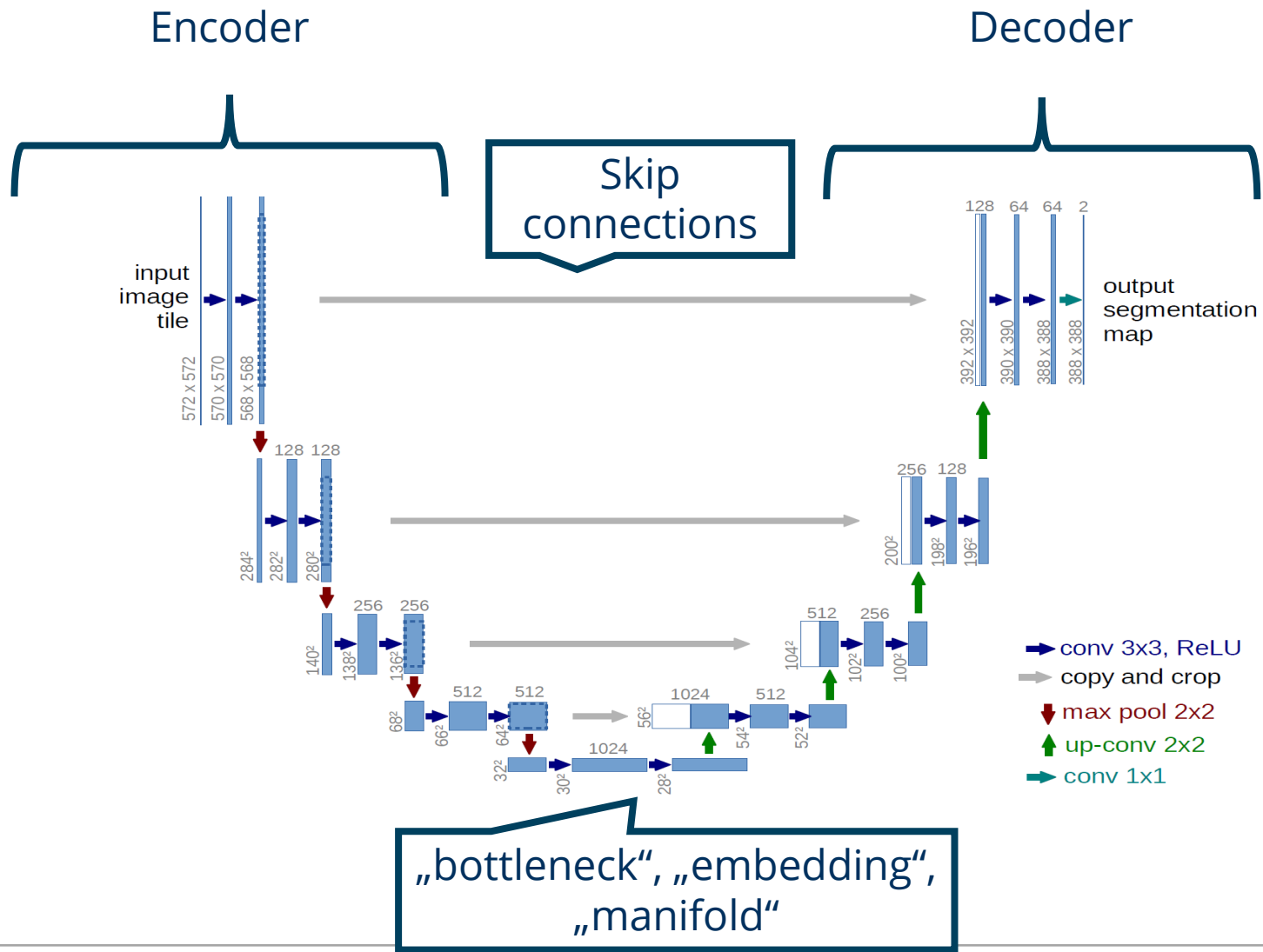


Image denoising: CARE

- Content aware image restoration (CARE)
- Image acquisition of pairs of images: A high-quality and a low-quality image.
- Problem: shot noise, biology moves!

Pair of
images
required!

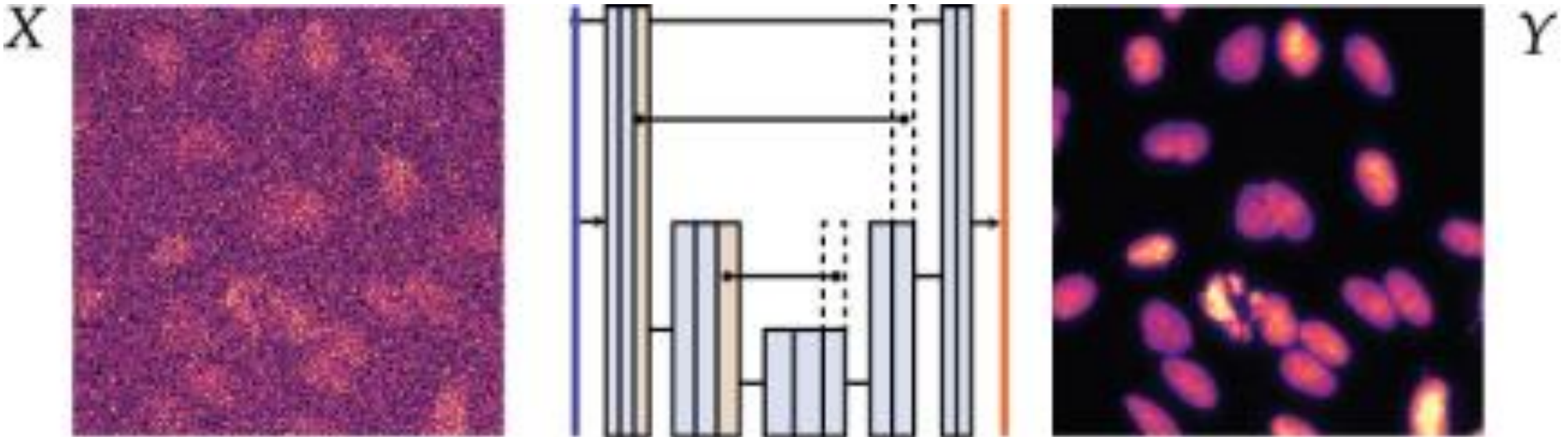
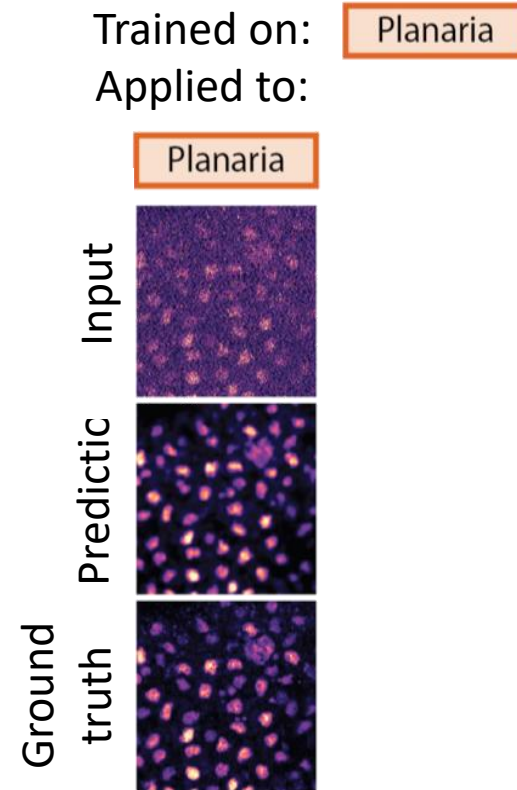


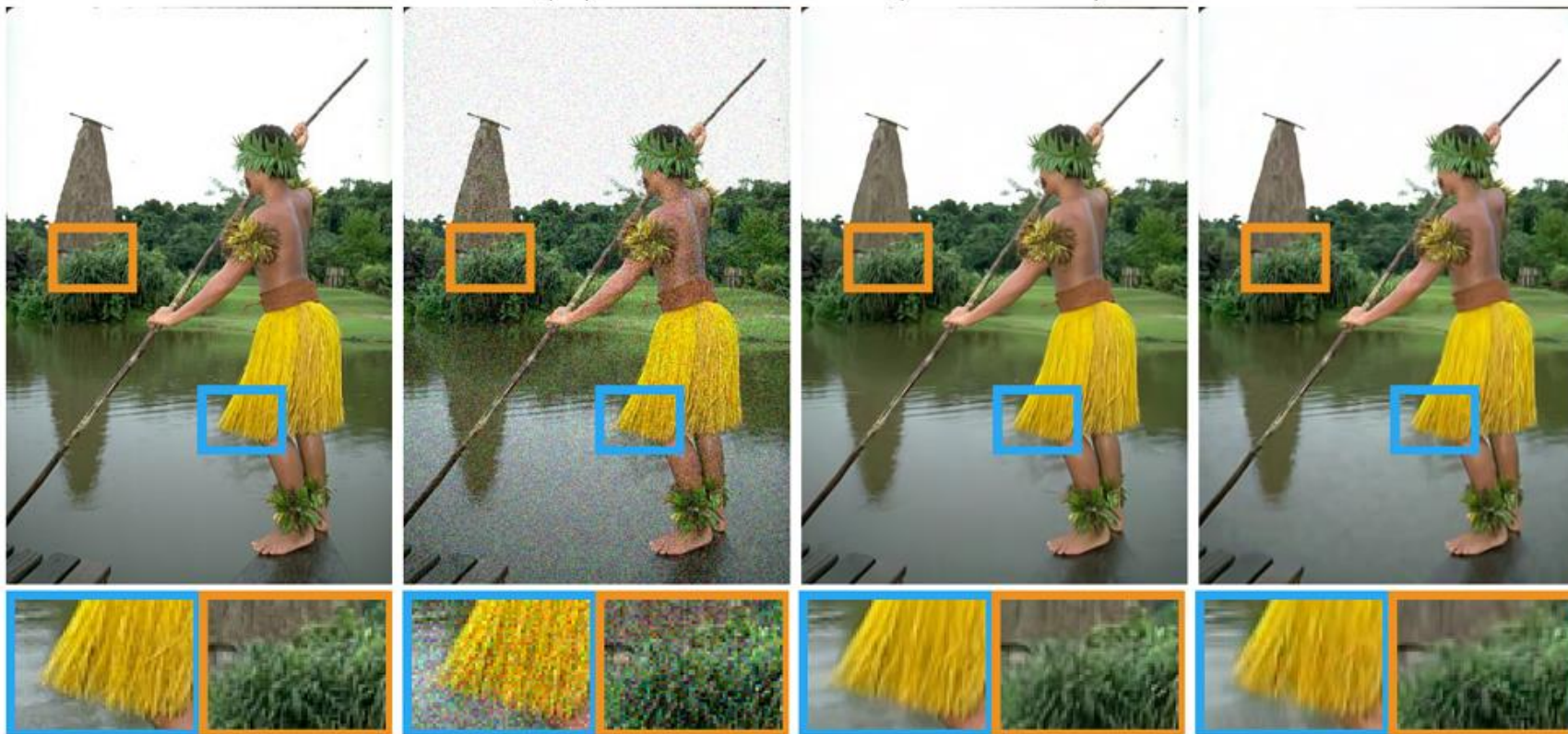
Image denoising: CARE

- Important to use on the same conditions/structures/staining that the networks were trained on!



Noise2noise

(a) Gaussian ($\sigma = 25$)



Ground truth

Input

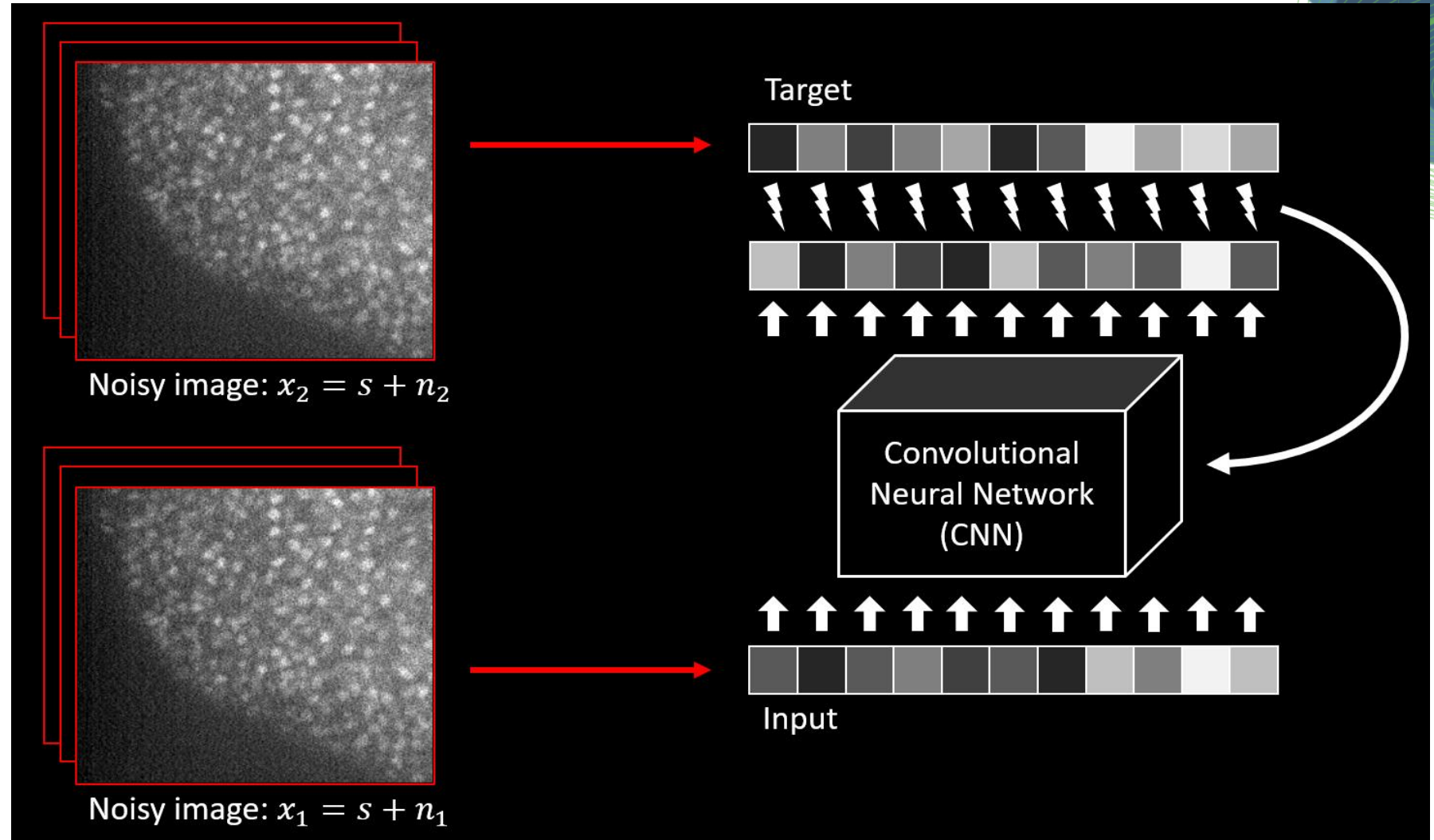
Our

Comparison

BM3D

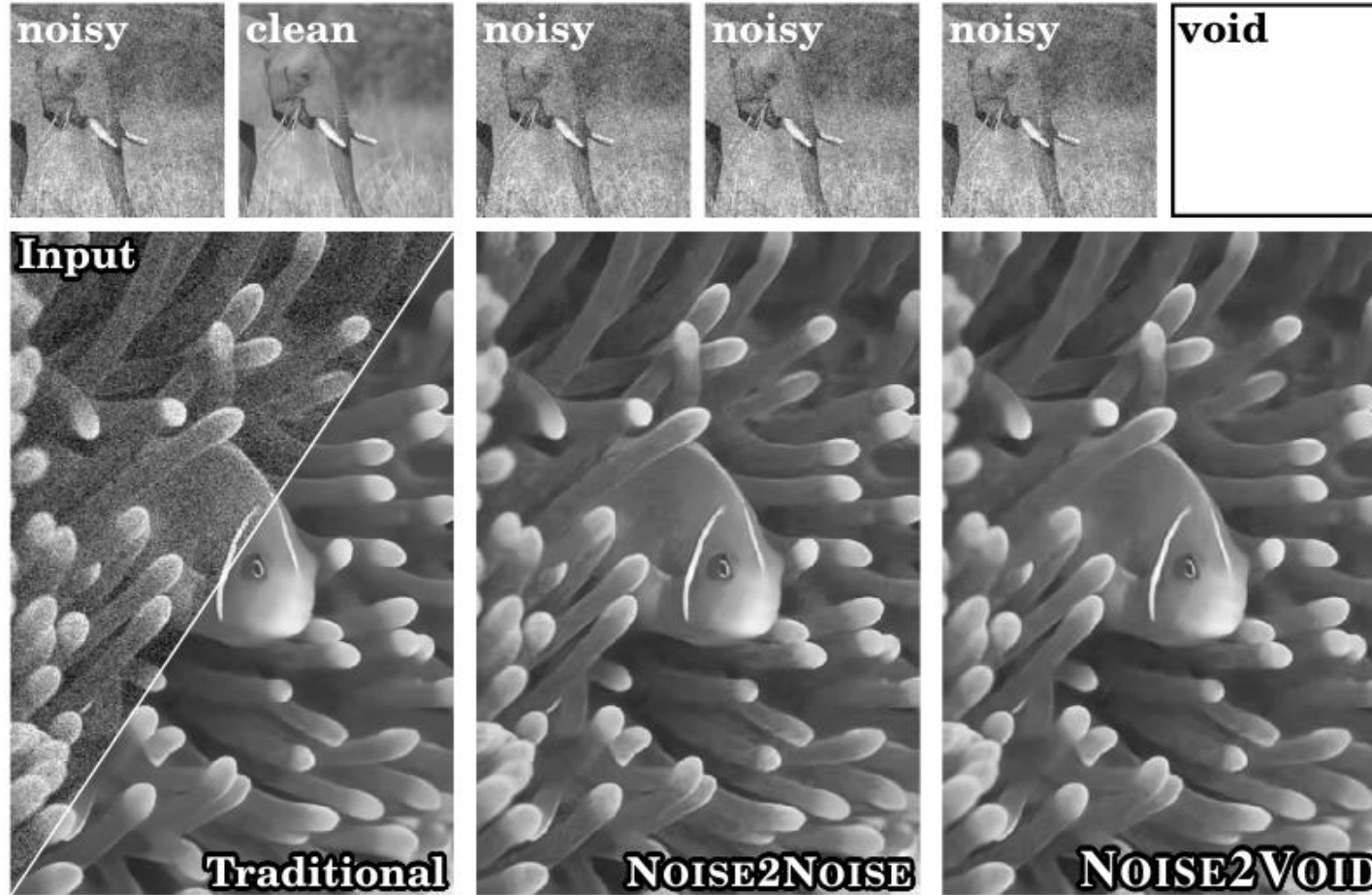
Noise2noise

- by Nvidia (Lehtinen 2018)
<https://arxiv.org/pdf/1803.04189.pdf>



Noise2void

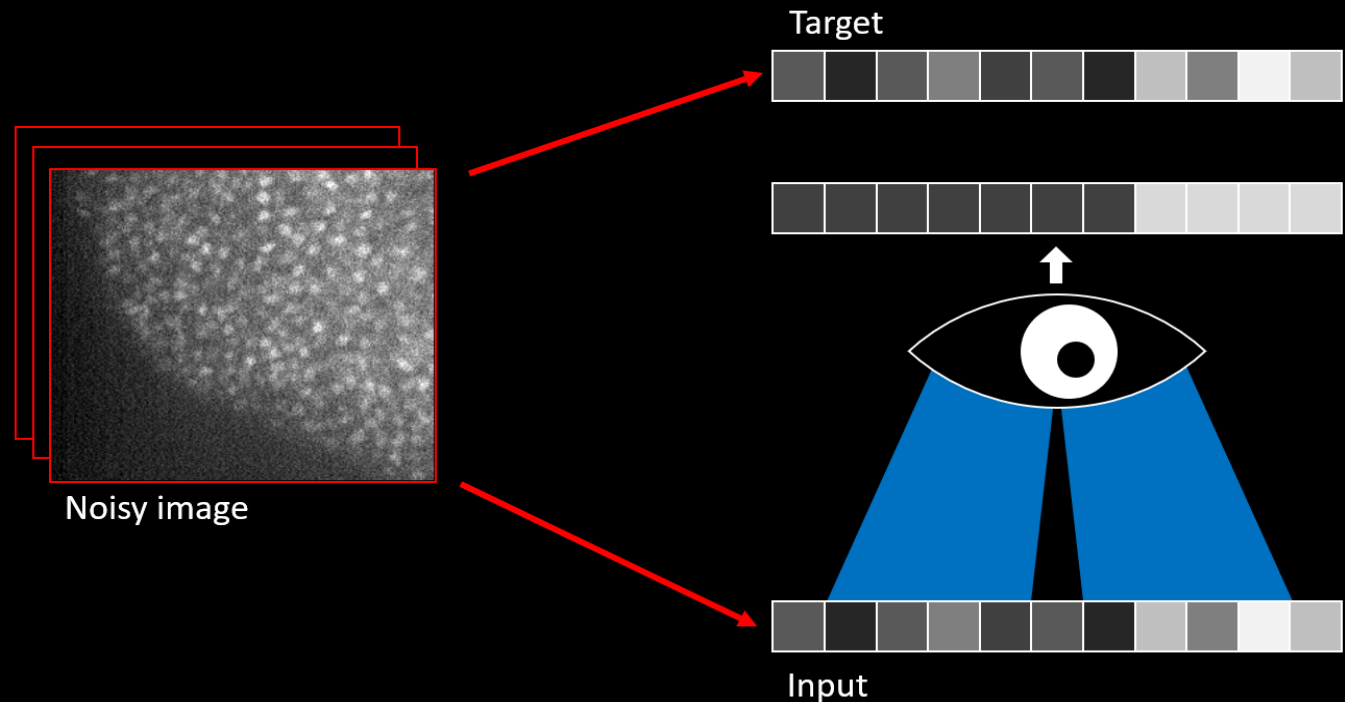
Image denoising without image pairs



Noise2void

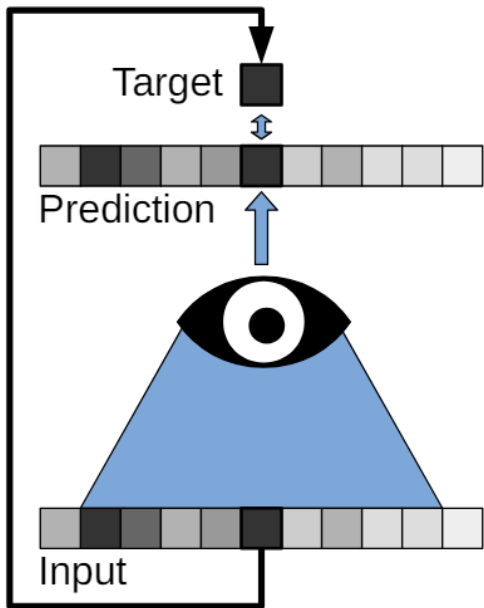
- Noise2void: Krull et al (2019)
<https://arxiv.org/abs/1811.10980>
- Noise2self: Batson and Royer (2019)
<https://arxiv.org/abs/1901.11365>

Noise2Void - Blind Spot Network

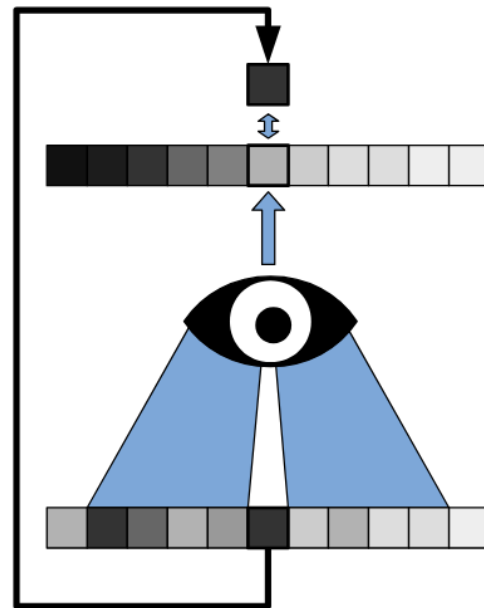


Noise2void

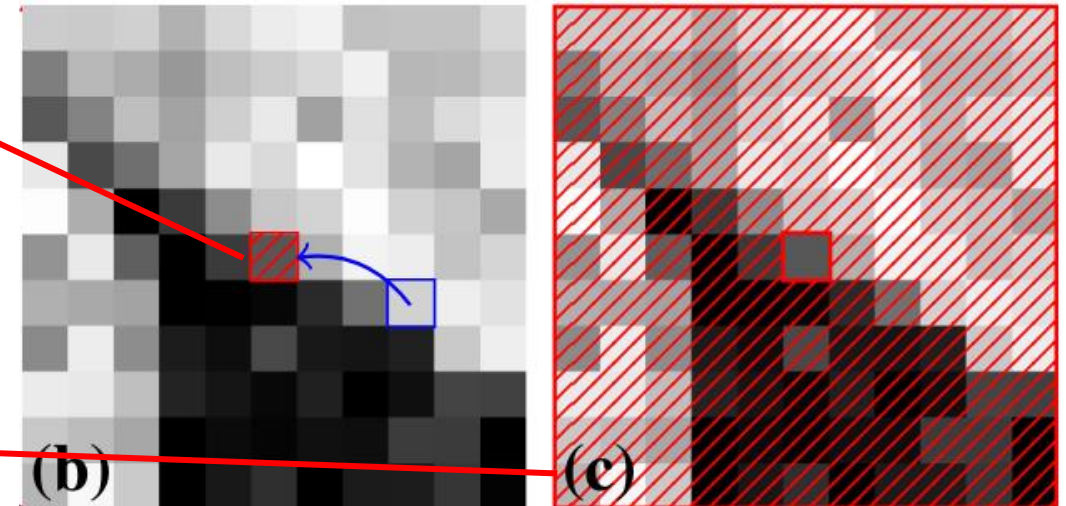
„Blind-spot-network“



(a)



(b)



(b)

(e)

Noise2void

Napari-plugin

napari-n2v: N2V Train

Train

Reset model Continue training

Training progress

Epoch 200/200

Step 400/400

loss

epoch

Train

Val

Open in TensorBoard

Prediction

Tile prediction

Number of tiles 4

Prediction 3172/3172

Predict again

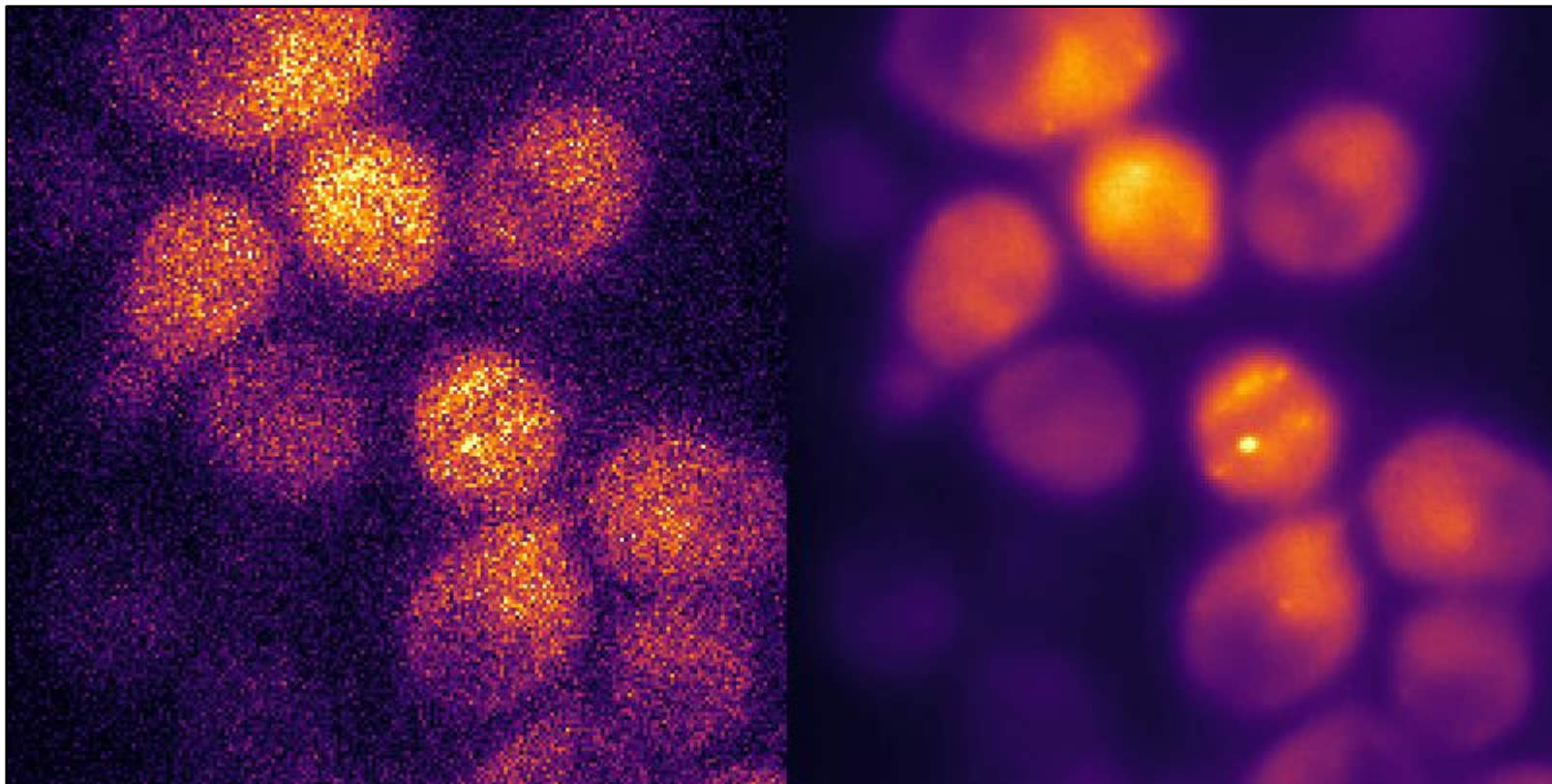
Noisy

Denoised

0 1584 | 3167

Image denoising

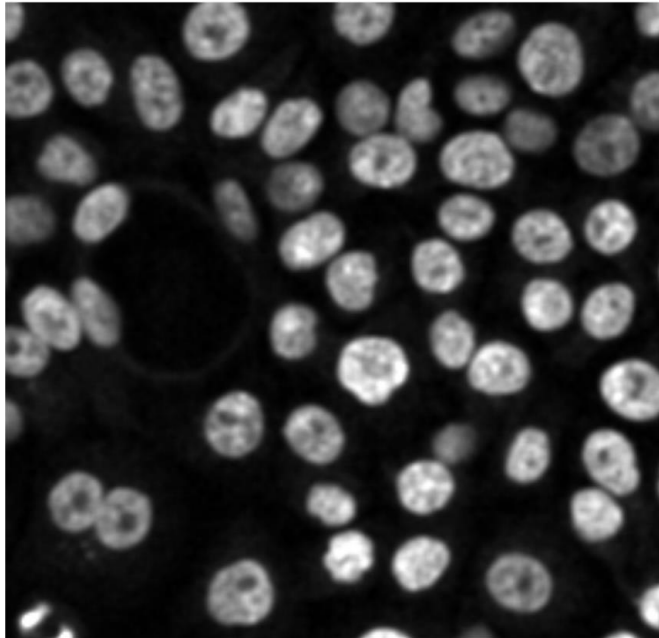
- Noise2Void



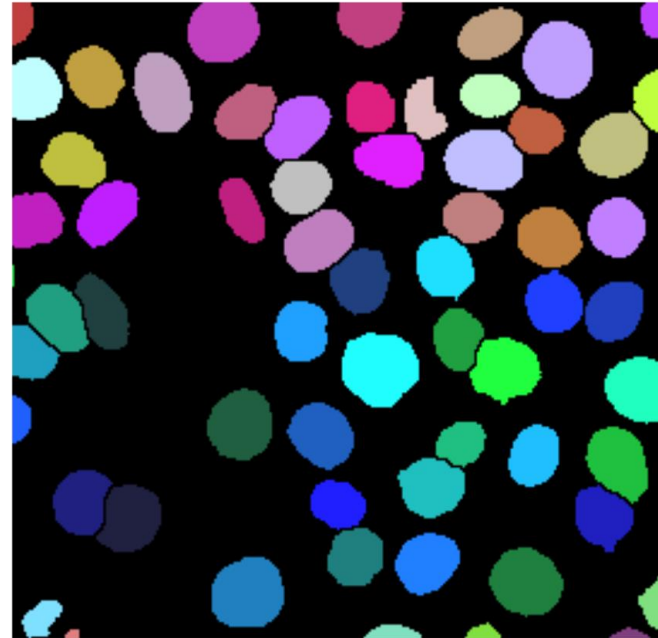
StarDist: Nuclei segmentation

Advanced algorithms are necessary when nuclei become too dense.

Image of nuclei



Corresponding masks



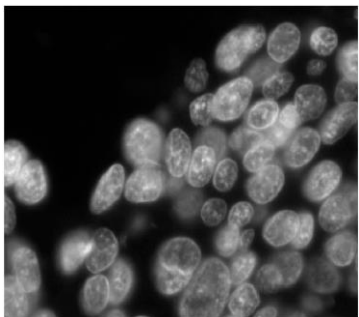
StarDist

- Prediction of probable object centers + polygon outlines
- Non-maximum-suppression of less likely polygons

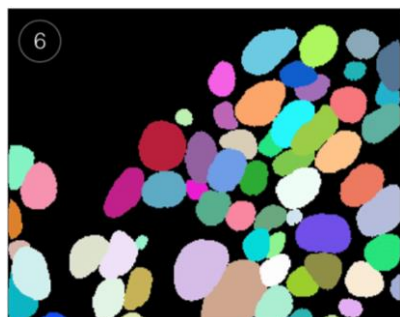
Class. Image proc.

Deep learning

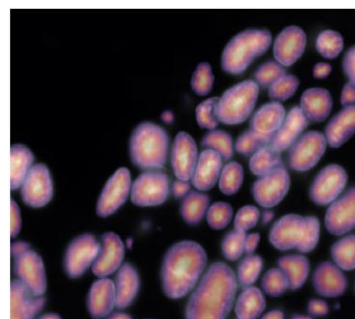
Input



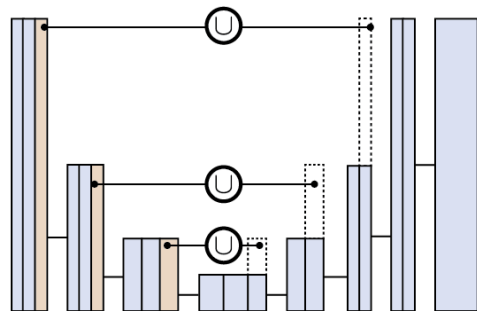
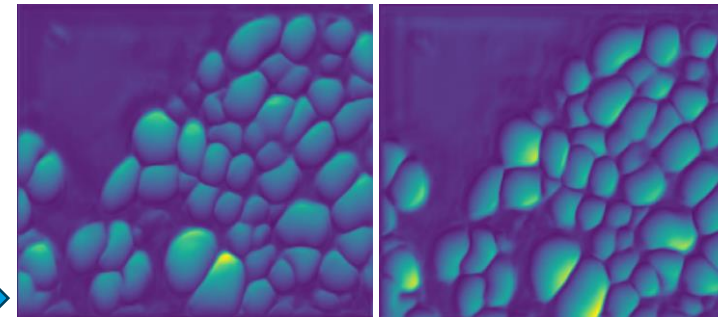
Ground truth



Object probabilities



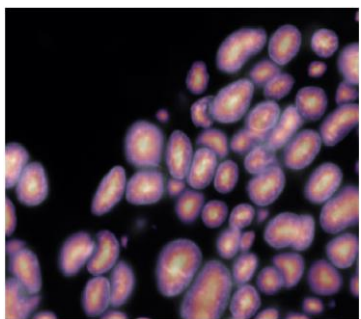
Directional distance maps (32x)



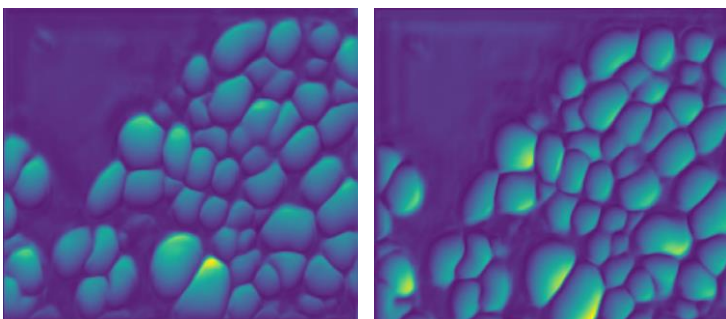
StarDist

- Prediction of probable object centers + polygon outlines
- Non-maximum-suppression of less likely polygons

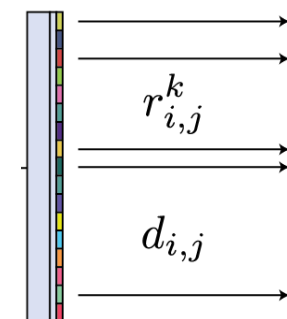
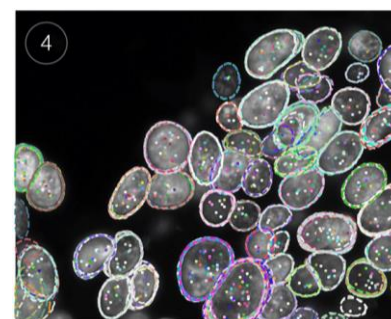
Object probabilities



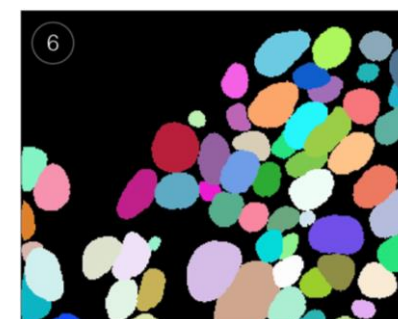
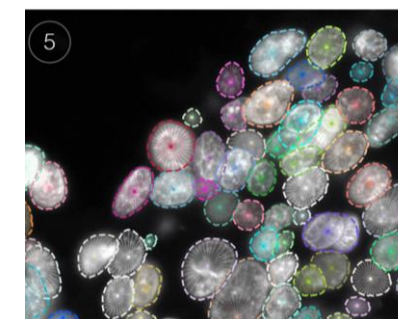
Directional distance maps



Polygon candidates

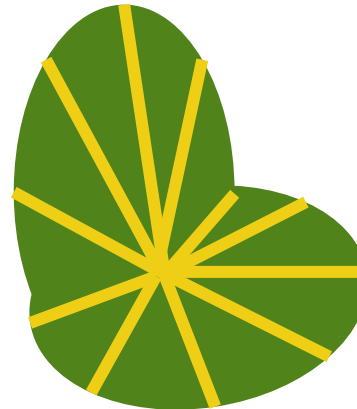
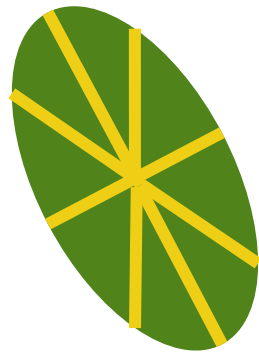
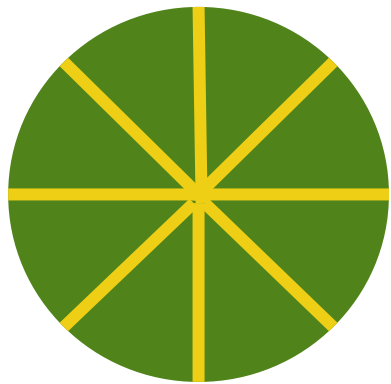


Final segmentation



StarDist: Limitations

Star-convex shapes!



Good for nuclei, bad for cells.

StarDist: Python

Initialize model / download pretrained model

```
[3]: # creates a pretrained model
model = StarDist2D.from_pretrained('2D_versatile_fluo')

Found model '2D_versatile_fluo' for 'StarDist2D'.
Loading network weights from 'weights_best.h5'.
Loading thresholds from 'thresholds.json'.
Using default values: prob_thresh=0.479071, nms_thresh=0.3.
```

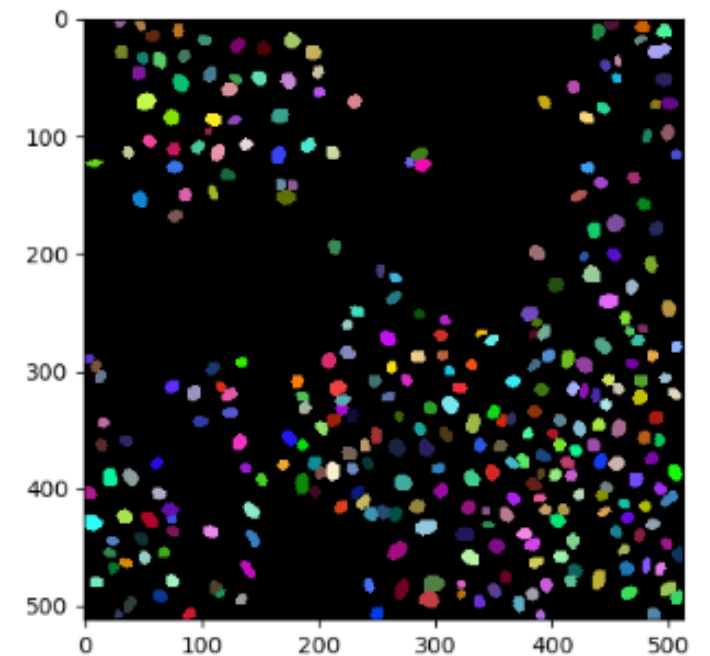
Normalize intensity to range [0, 1]

```
[4]: axis_norm = (0,1)
image = normalize(image, 1,99.8, axis=axis_norm)
```

Apply model

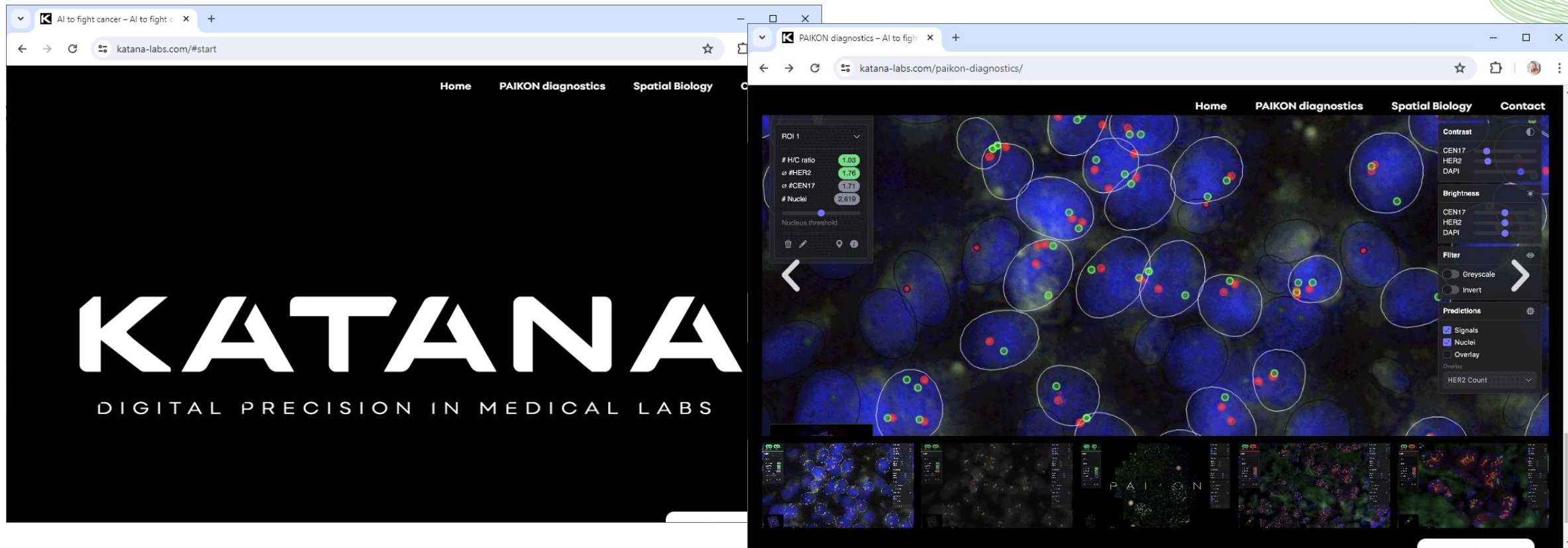
```
[5]: labels, details = model.predict_instances(image)
stackview.insight(labels)
```

[5]:



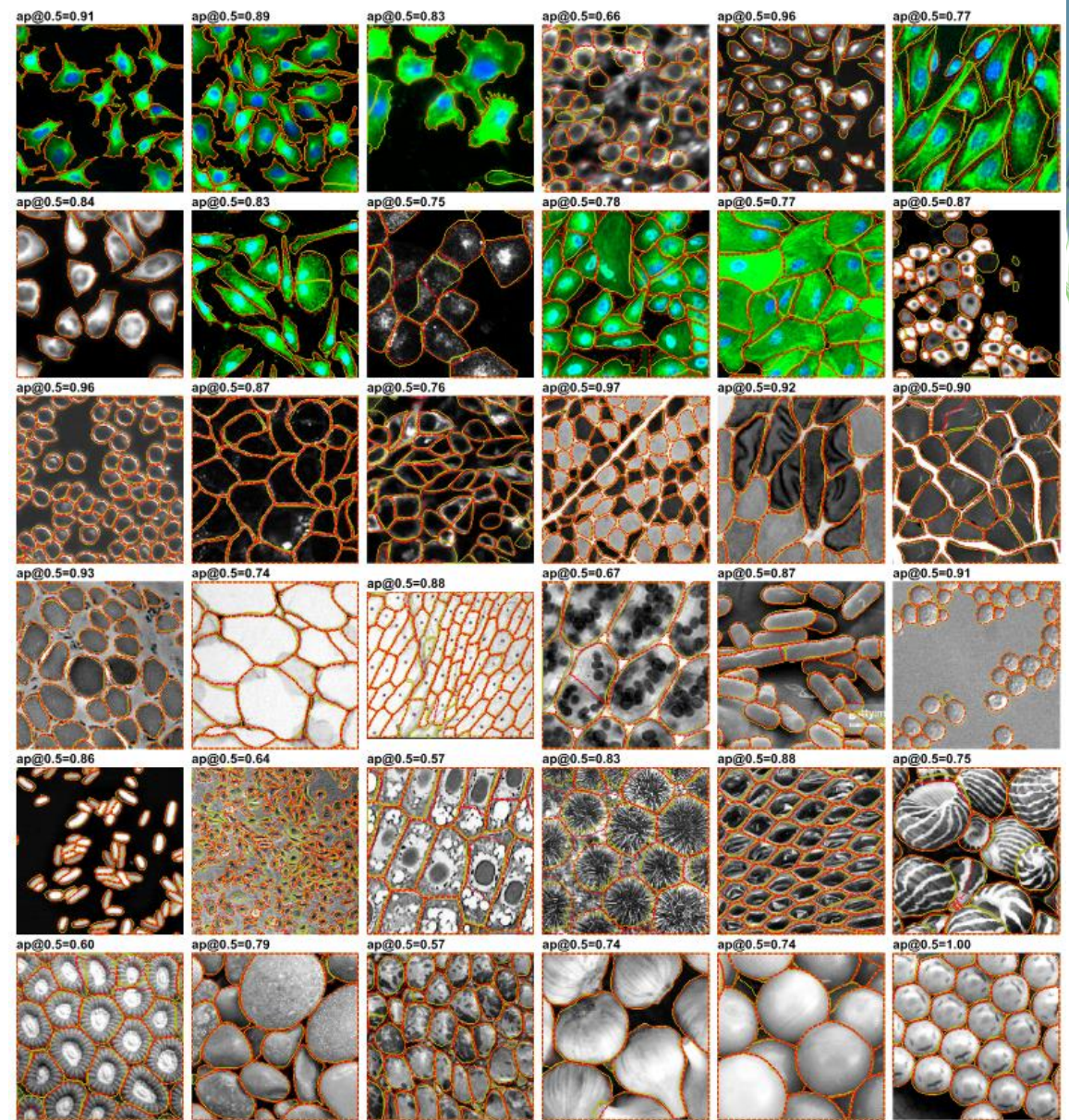
StarDist: Business model

Serving clinical scientists using software based on open-source code



CellPose

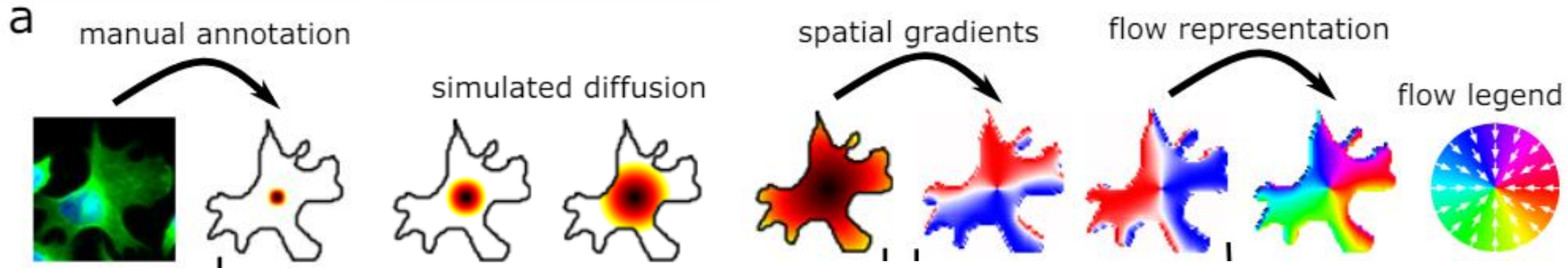
- Processing more diverse datasets and object shapes



CellPose

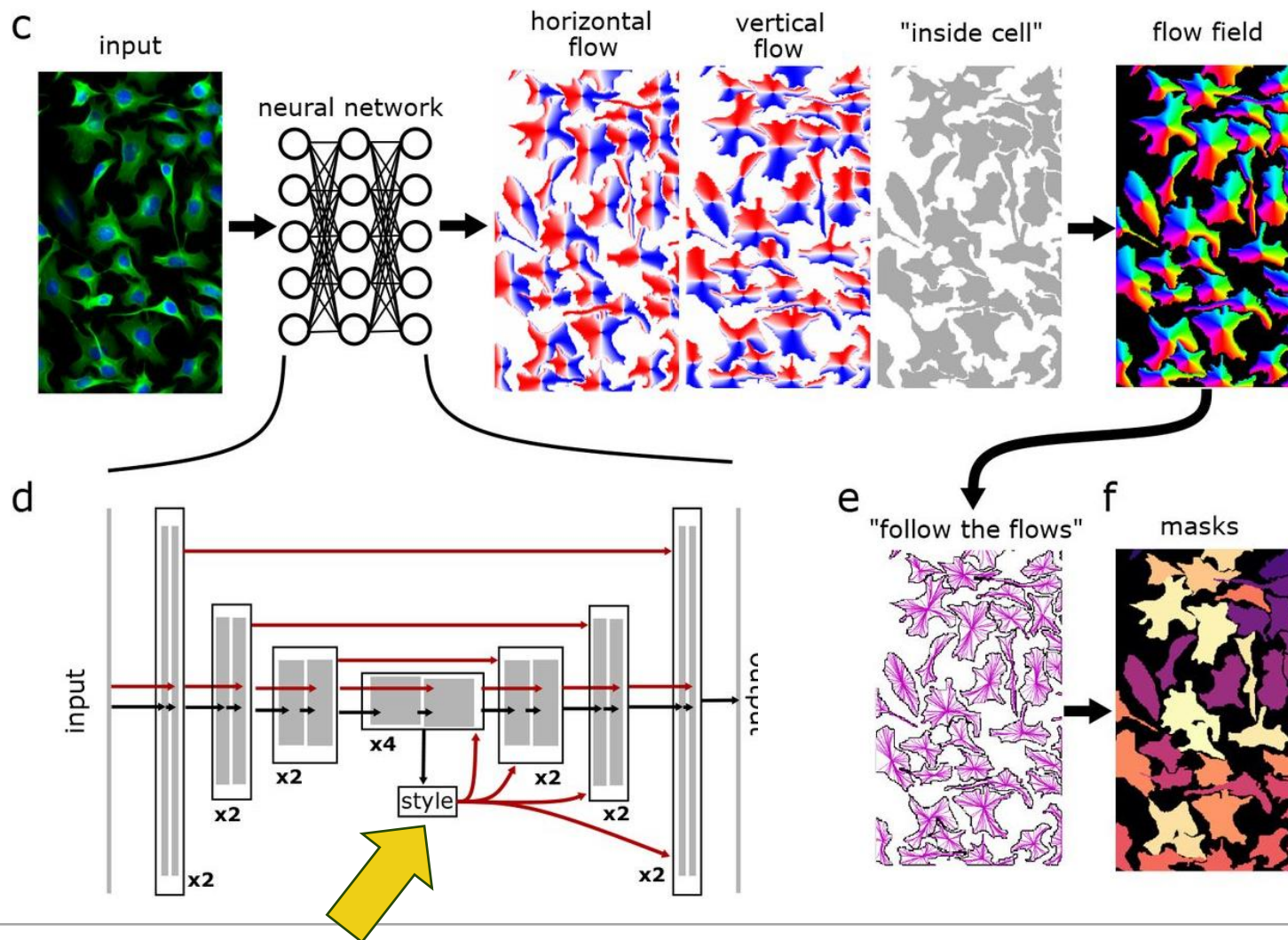
Compute „flow fields“ from images

- using classical image processing during training
- using neural networks during prediction



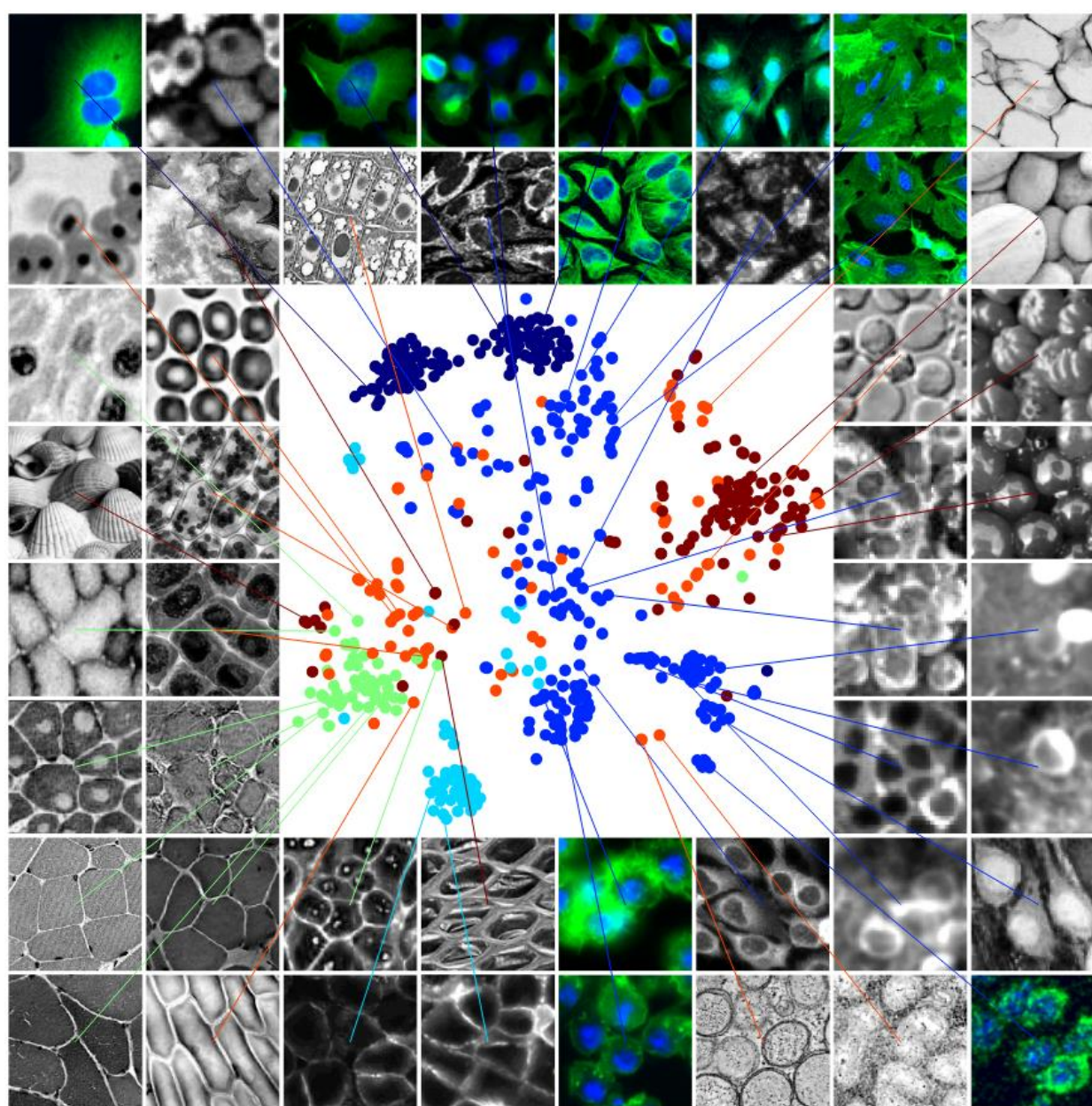
CellPose

- Cell/Nuclei – segmentation based on flow-fields
- Technically similar to Watershed, but with a deep-learning based altitude-image

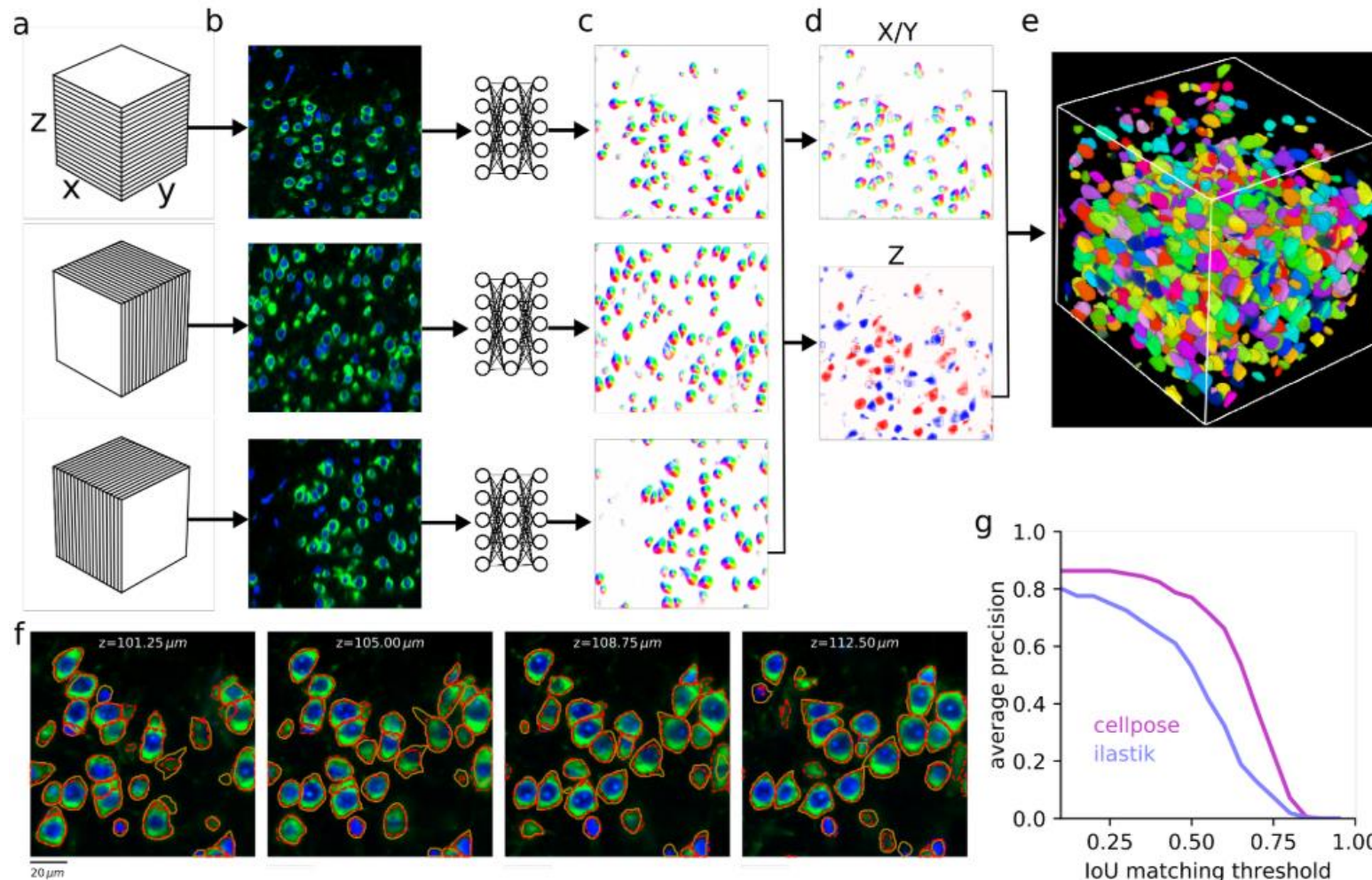


CellPose

- Image *style* is a parameter determined before prediction to guide segmentation.

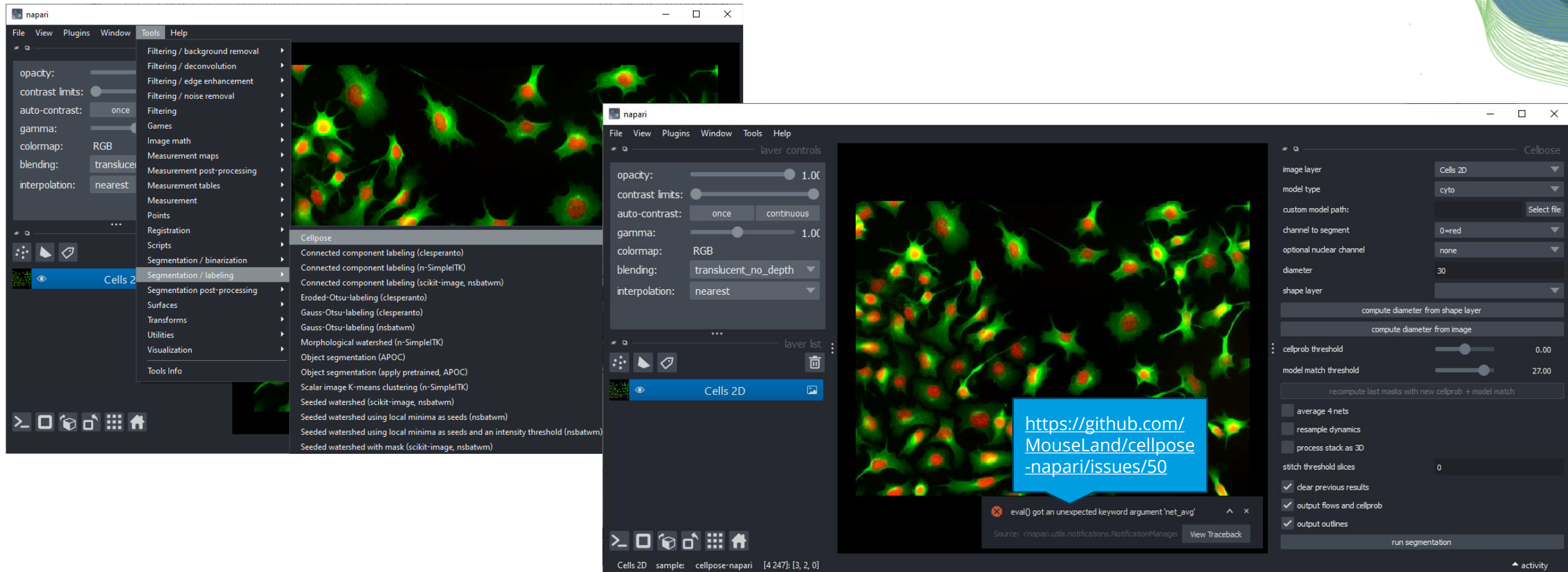


Cellpose 3D



Cellpose in Napari

mamba install cellpose-napari



Cellpose in Python

Initialize a pretrained model

```
[3]: model = cellpose.models.Cellpose(gpu=False, model_type='nuclei')
```

List available models

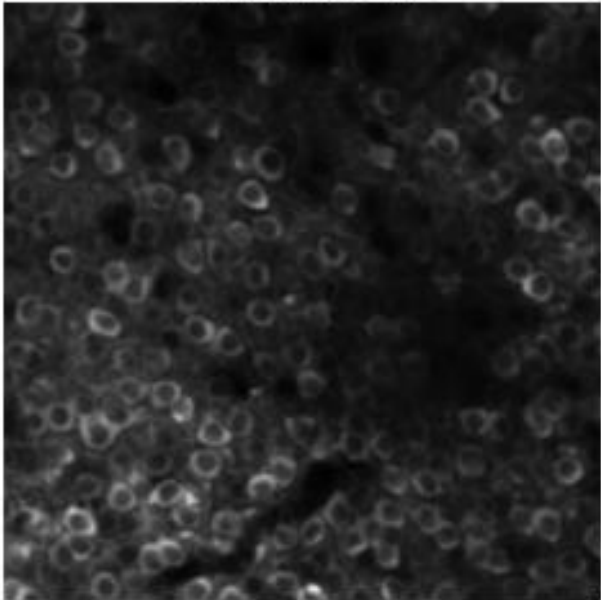
```
[8]: cellpose.models.MODEL_NAMES
```

```
[8]: ['cyto3',  
      'nuclei',  
      'cyto2_cp3',  
      'tissuenet_cp3',  
      'livecell_cp3',  
      'yeast_PhC_cp3',  
      'yeast_BF_cp3',  
      'bact_phase_cp3',  
      'bact_fluor_cp3',  
      'deepbacs_cp3',  
      'cyto2',  
      'cyto',  
      'transformer_cp3',  
      'neurips_cellpose_default',  
      'neurips_cellpose_transformer',  
      'neurips_grayscale_cyto2']
```

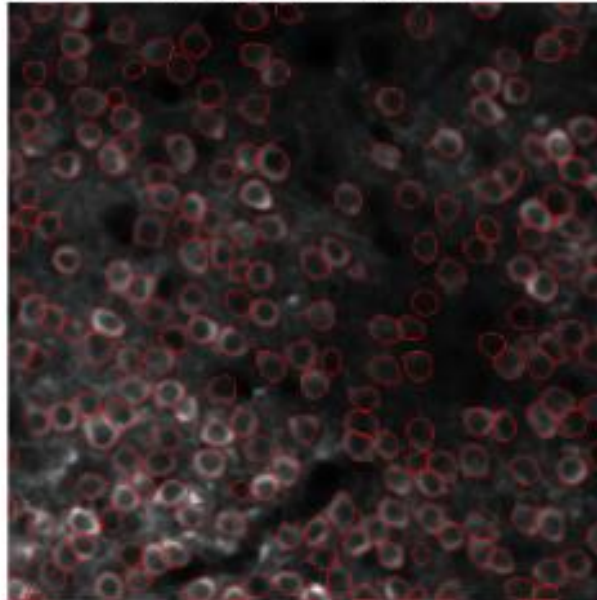

Cellpose in Python

```
[4]: channels = [0,0] # This means we are processing single-channel greyscale images.  
masks, flows, styles, diams = model.eval(image, diameter=None, channels=channels)
```

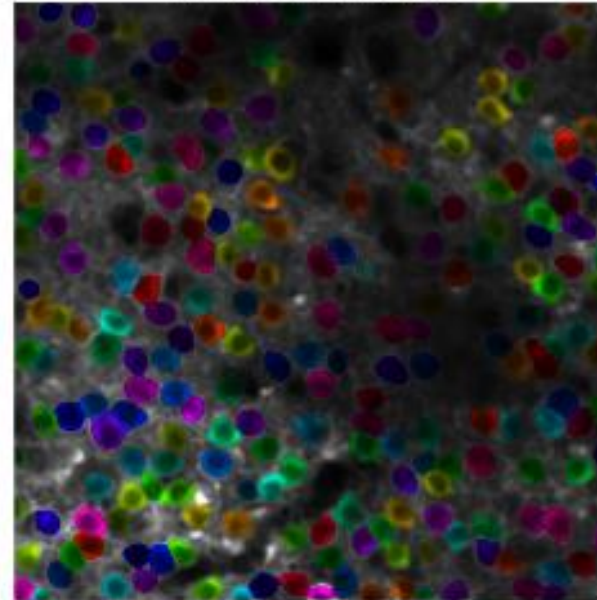
original image



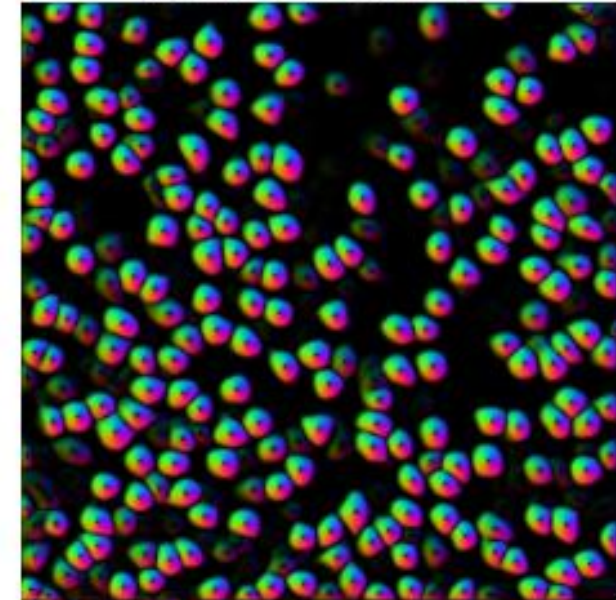
predicted outlines



predicted masks



predicted cell pose



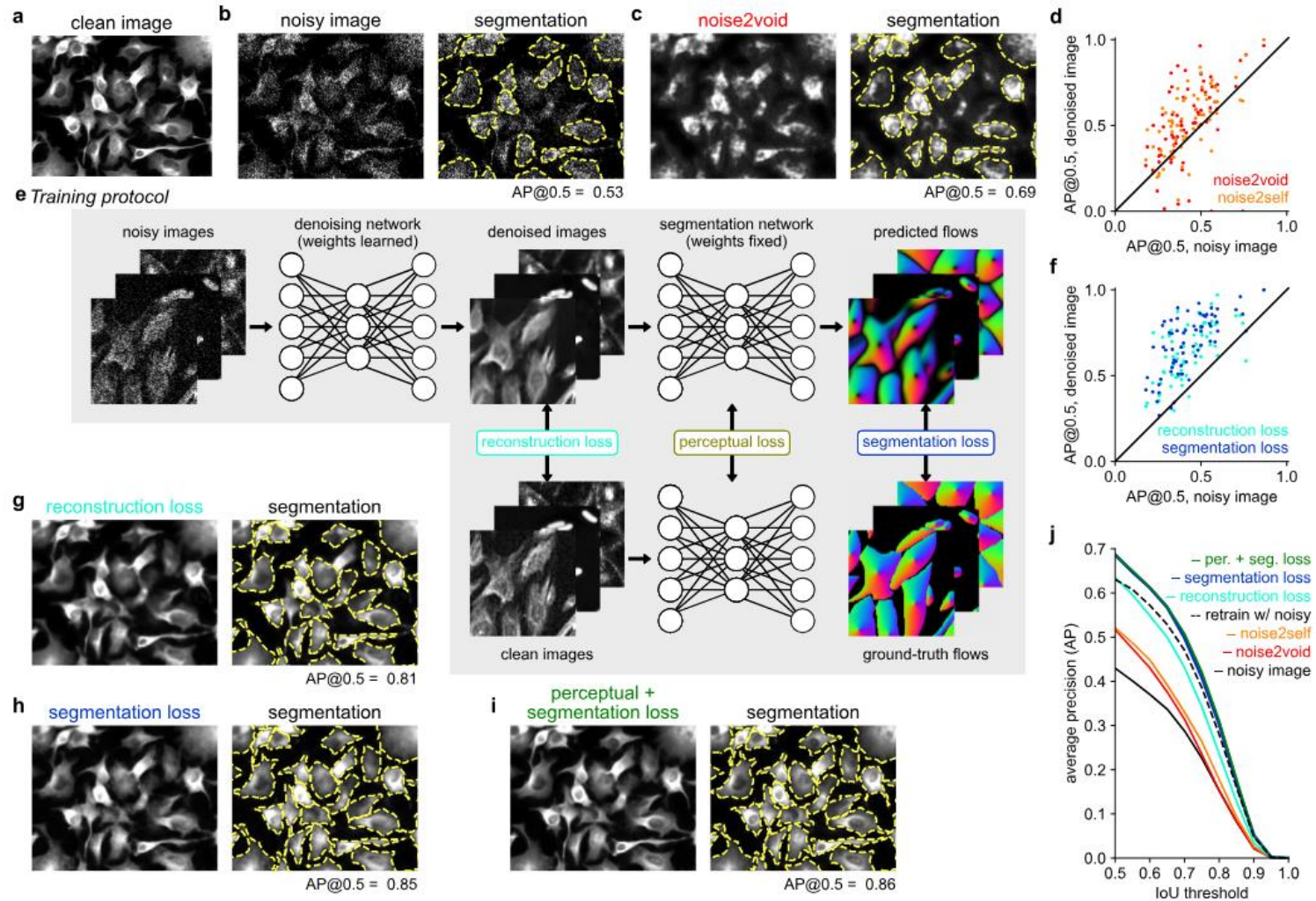
Cellpose 2

Train your own model

The screenshot shows a web browser displaying the article "Cellpose 2.0: how to train your own model" on the Nature Methods website. The browser's address bar shows the URL "nature.com/articles/s41592-022-01663-4". The page header includes the "nature methods" logo, navigation links for "View all journals", "Search", and "Log in", and a "Sign up for alerts" button. The article title is prominently displayed, along with the authors "Marius Pachitariu" and "Carsen Stringer". The publication date is "07 November 2022". A "Download PDF" button is visible. The abstract text is partially visible, starting with "Pretrained neural network models for biological segmentation can provide good out-of-the-box results for many image types. However, such models do not allow users to adapt the segmentation style to their specific needs and can perform suboptimally for test images that are very different from the training images. Here we introduce Cellpose 2.0, a new package that includes an ensemble of diverse pretrained models as well as a human-in-the-loop pipeline for rapid prototyping of new custom models. We show that models pretrained on the Cellpose dataset can be fine-tuned with only 500–1,000 user-annotated regions".

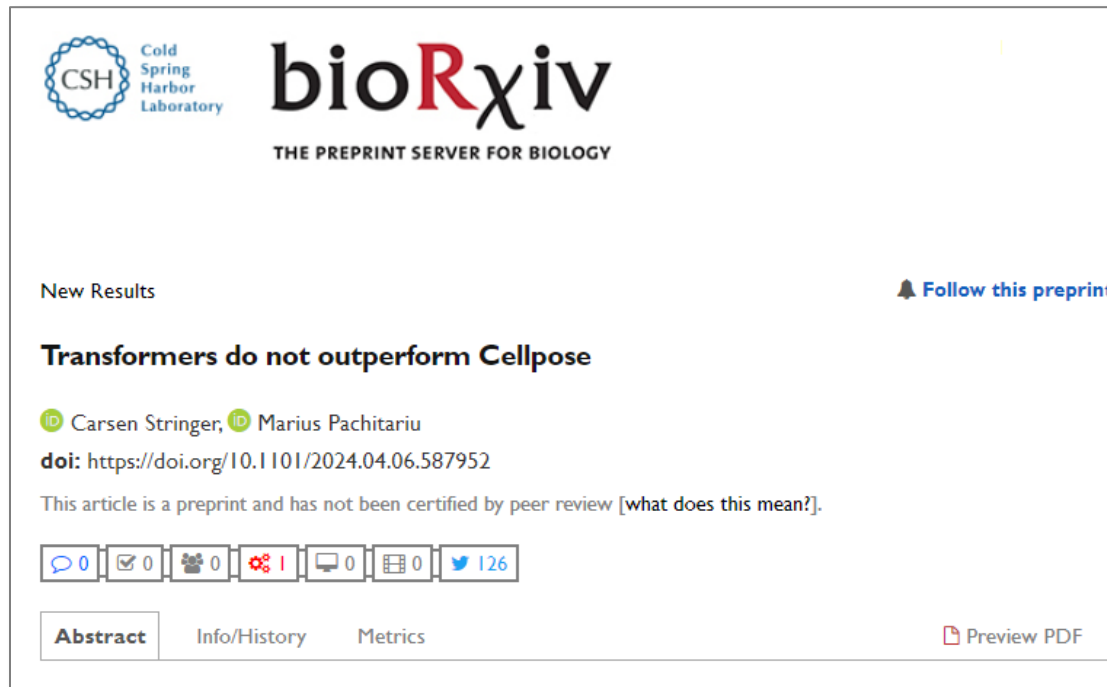
Cellpose 3

Cellpose 3 includes denoising and deblurring, to improve image segmentation quality



Cellpose

State-of-the-art despite the rise of transformers



The screenshot shows the bioRxiv preprint interface. At the top left is the Cold Spring Harbor Laboratory logo. The title is "Transformers do not outperform Cellpose" by Carsen Stringer and Marius Pachitariu. The DOI is https://doi.org/10.1101/2024.04.06.587952. Below the title are social media sharing icons for comments, email, print, a gear icon, a speech bubble, a calendar, and a retweet icon with a count of 126. At the bottom of the preprint area are tabs for "Abstract", "Info/History", and "Metrics", along with a "Preview PDF" button.

Abstract

In a recent publication, Ma et al (2024) claim that a transformer-based cellular segmentation method called Mediar - which won a Neurips challenge - outperforms Cellpose (0.897 vs 0.543 median F1 score). Here we show that this result was obtained by artificially impairing Cellpose in multiple ways. When we removed these impairments, Cellpose outperformed Mediar (0.861 vs 0.826 median F1 score on the updated test set). To further investigate the performance of transformers for cellular segmentation, we replaced the Cellpose backbone with a transformer. The transformer-Cellpose model also did not outperform the standard Cellpose (0.848 median F1 test score). Our results suggest that transformers do not advance the state-of-the-art in cellular segmentation.

Competing Interest Statement

The authors have declared no competing interest.

Copyright The copyright holder for this preprint is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a [CC-BY-NC 4.0 International license](#).

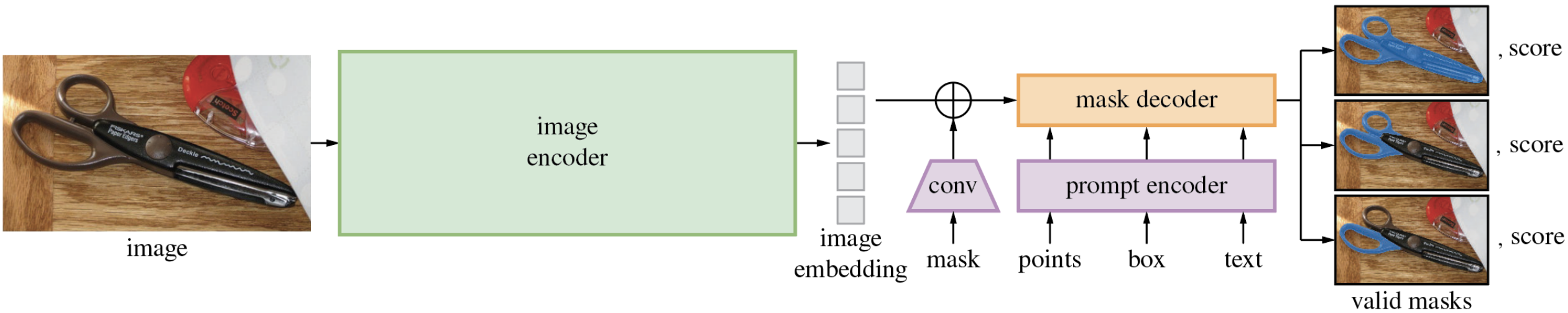
Segment Anything Model

New approach to DL-based image segmentation involving prompts



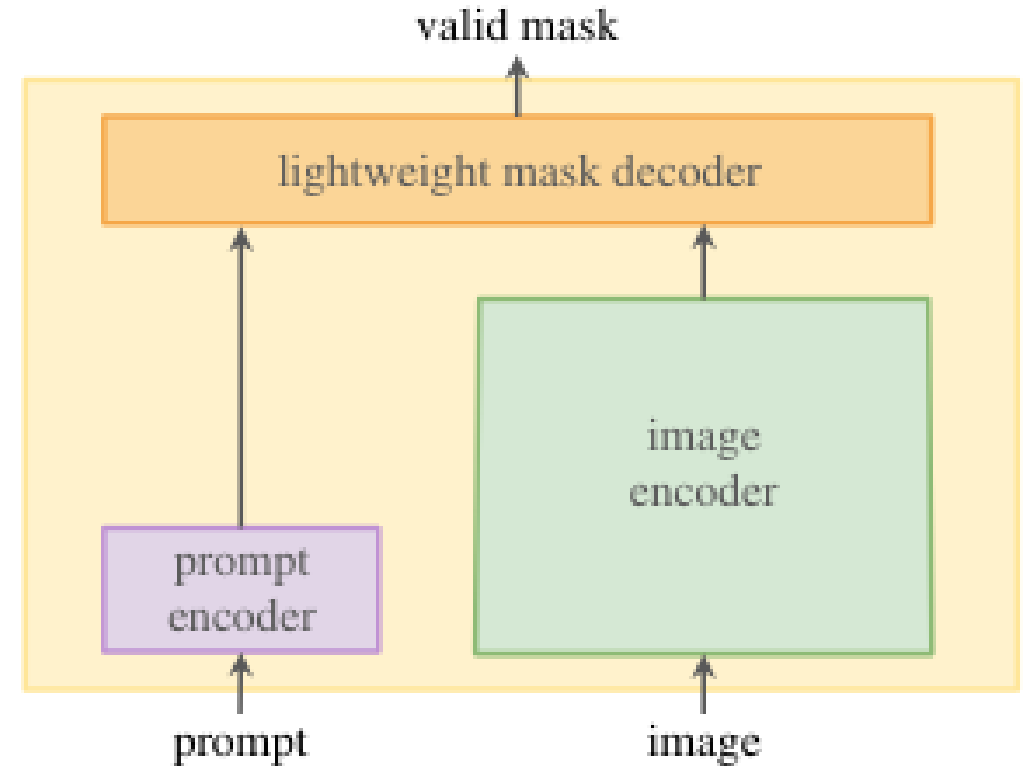
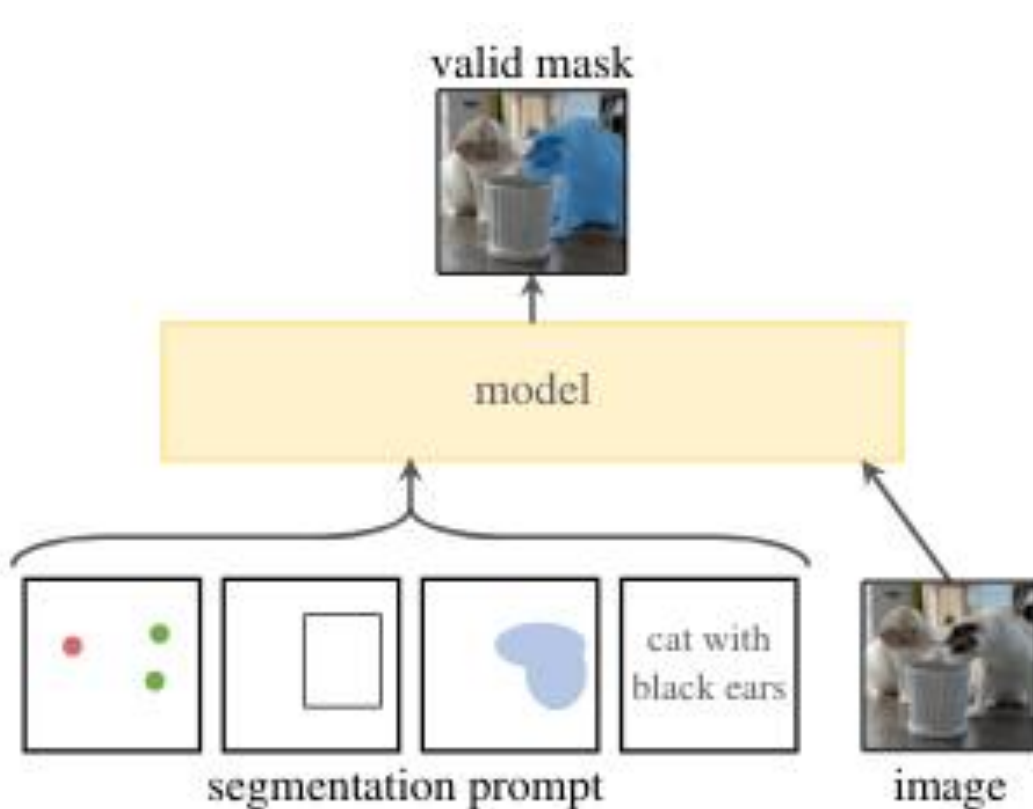
Segment Anything Model

New approach to DL-based image segmentation involving prompts



Segment Anything Model

New approach to DL-based image segmentation involving prompts



Segment Anything Model

Trained on mostly natural images

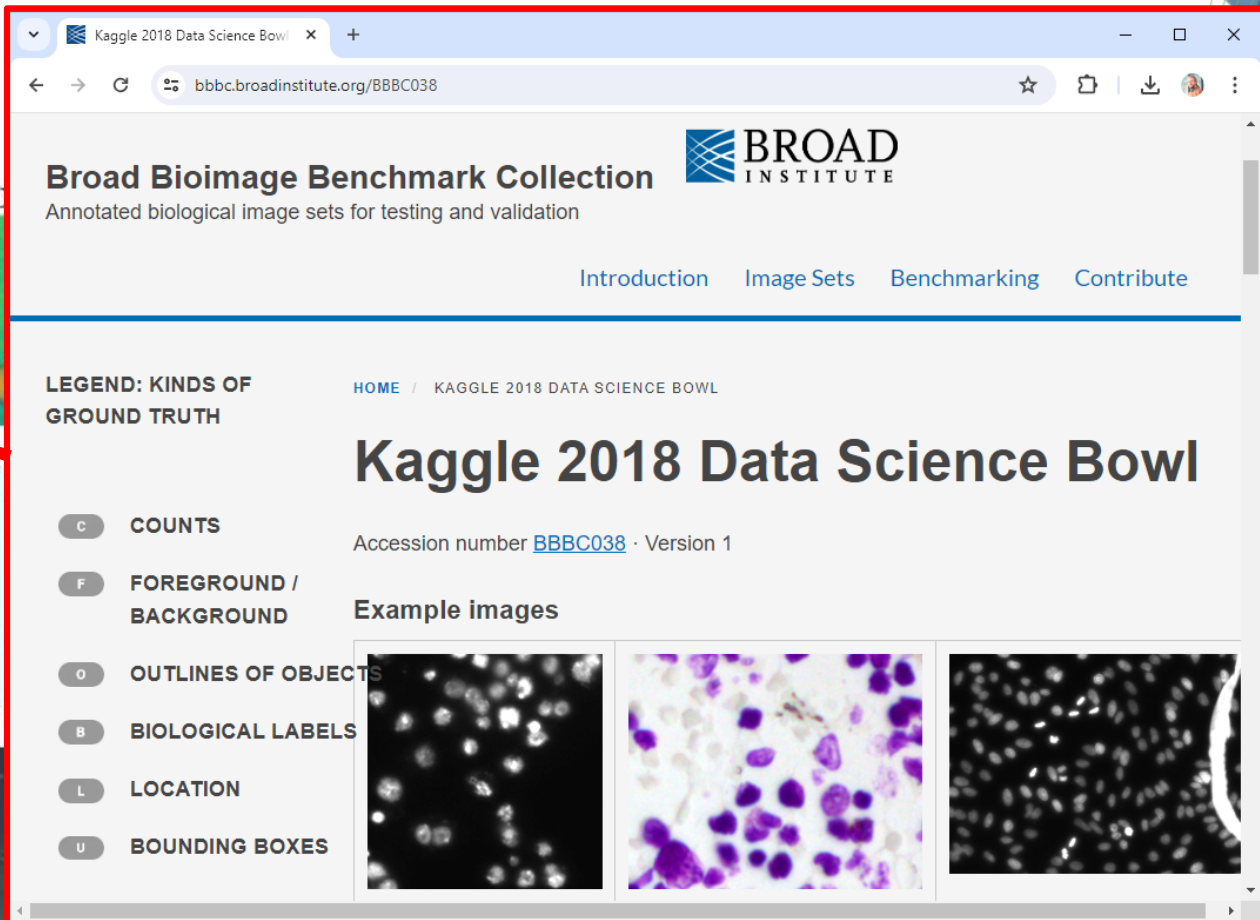
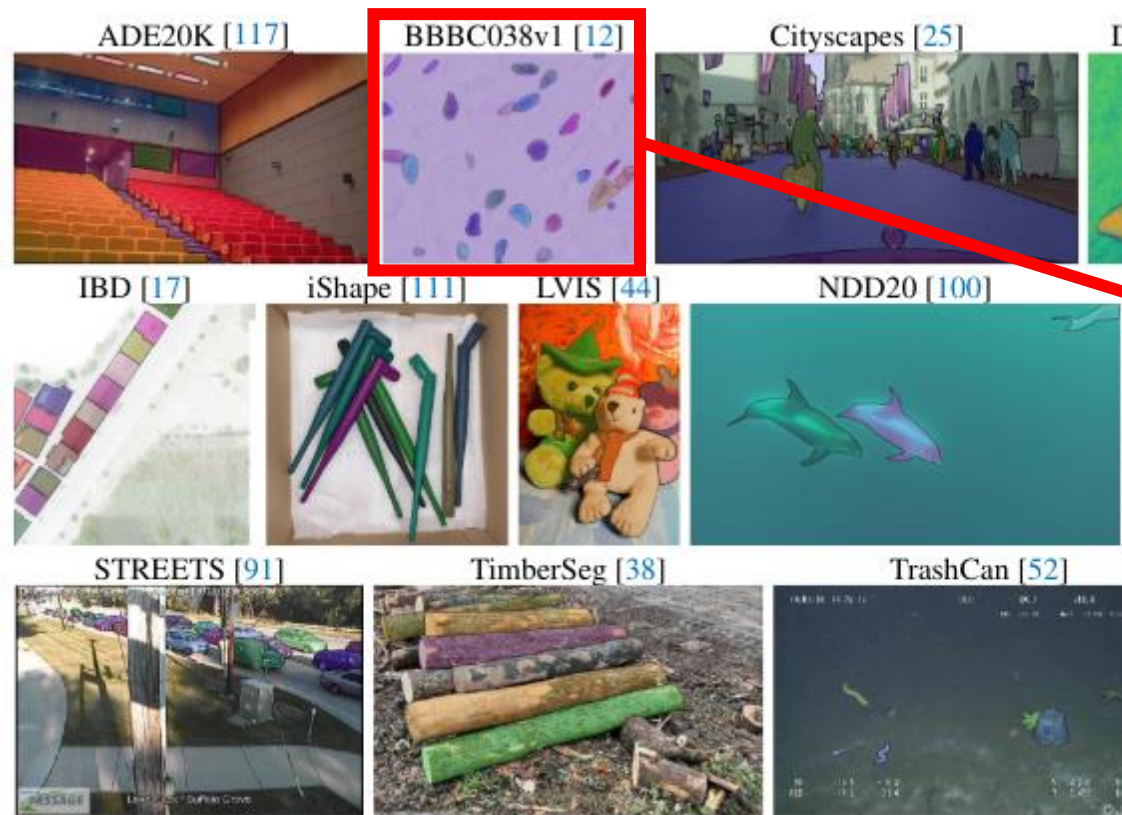


Figure 8: Samples from the 23 diverse segmentation datasets used to evaluate SAM's zero-shot transfer capabilities.

Segment Anything for Microscopy

Popping up napari plugins, some within 24h after SAM was published

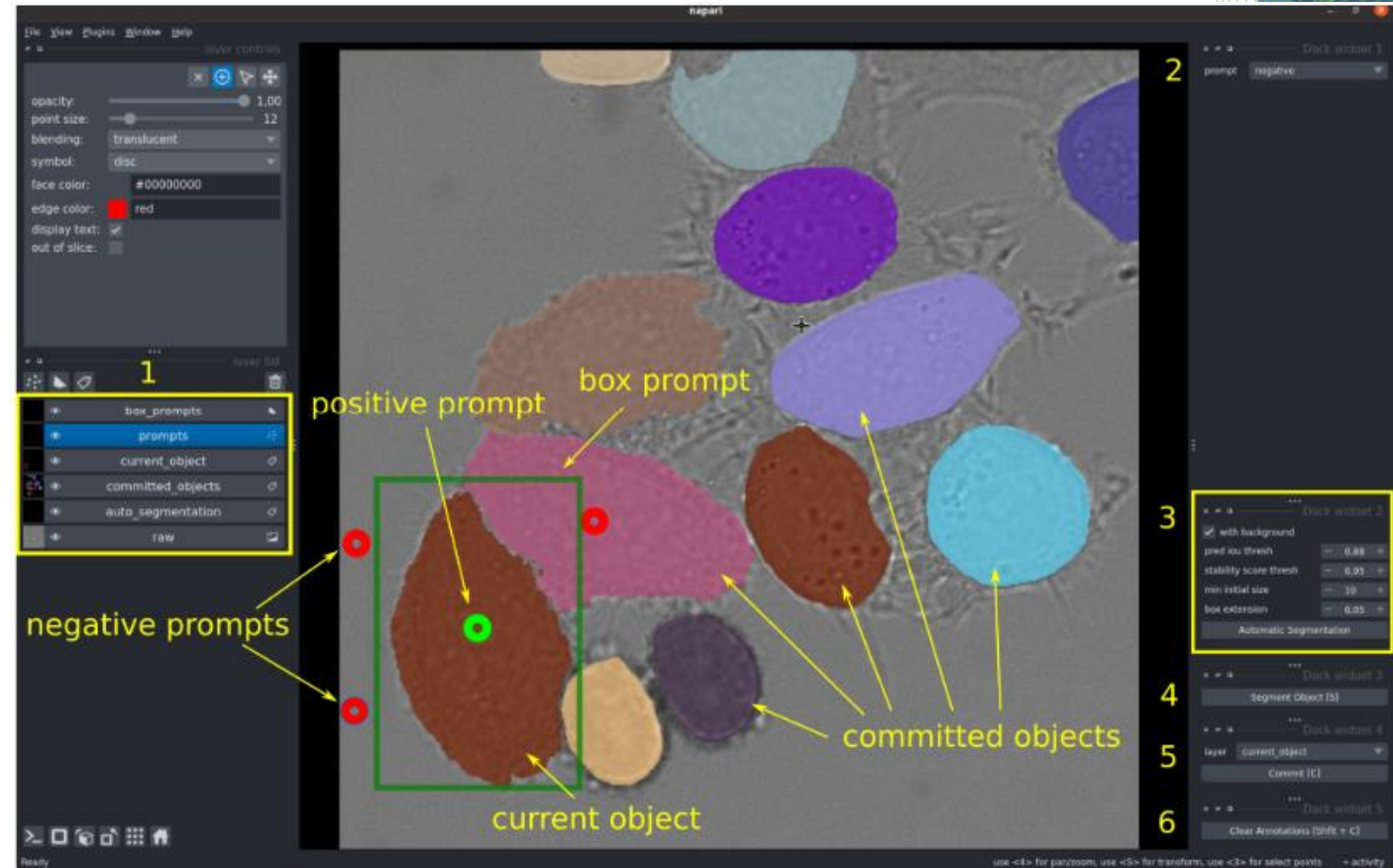
The image displays four browser windows showing GitHub repository pages for various napari plugins:

- h1roalchem/napari-SAM4IS**: A GitHub repository page for a napari plugin for instance and semantic segmentation using the Segment Anything Model (SAM). It includes a README, license (Apache-2.0), and installation instructions.
- royerlab/napari-segment-anything**: A GitHub repository page for a napari plugin of the Segment Anything Model (SAM). It includes a README, license (Apache-2.0), and a video demonstrating the plugin's functionality.
- MIC-DKFZ/napari-sam**: A GitHub repository page for a Segment Anything Model (SAM) in Napari. It includes a README, license (Apache-2.0), and a video demonstrating the plugin's functionality.
- computational-cell-analytics/micro-sam**: A GitHub repository page for Segment Anything for Microscopy. It includes a README, license (MIT), and a list of tools for segmentation and tracking in microscopy.

Segment Anything for Microscopy

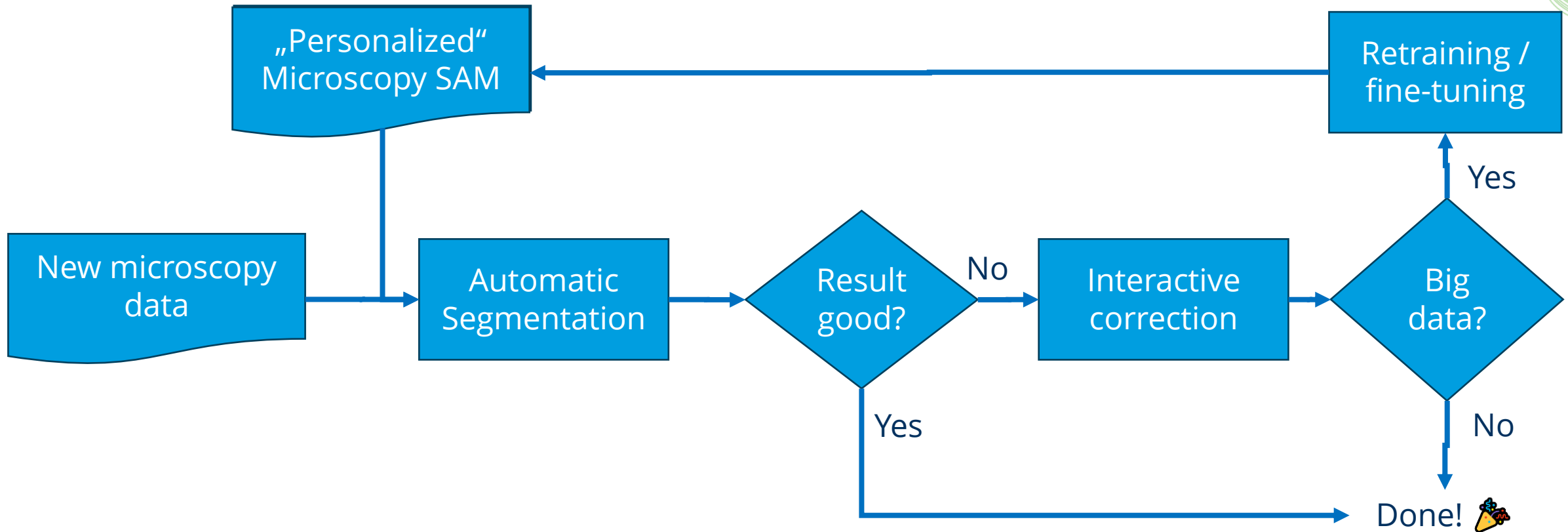
Downsides:

- Original code did not contain the procedure for iterative training
- Instance segmentation not ideal (watershed-implementation added in micro-sam)
- Fine-tuning for microscopy data necessary

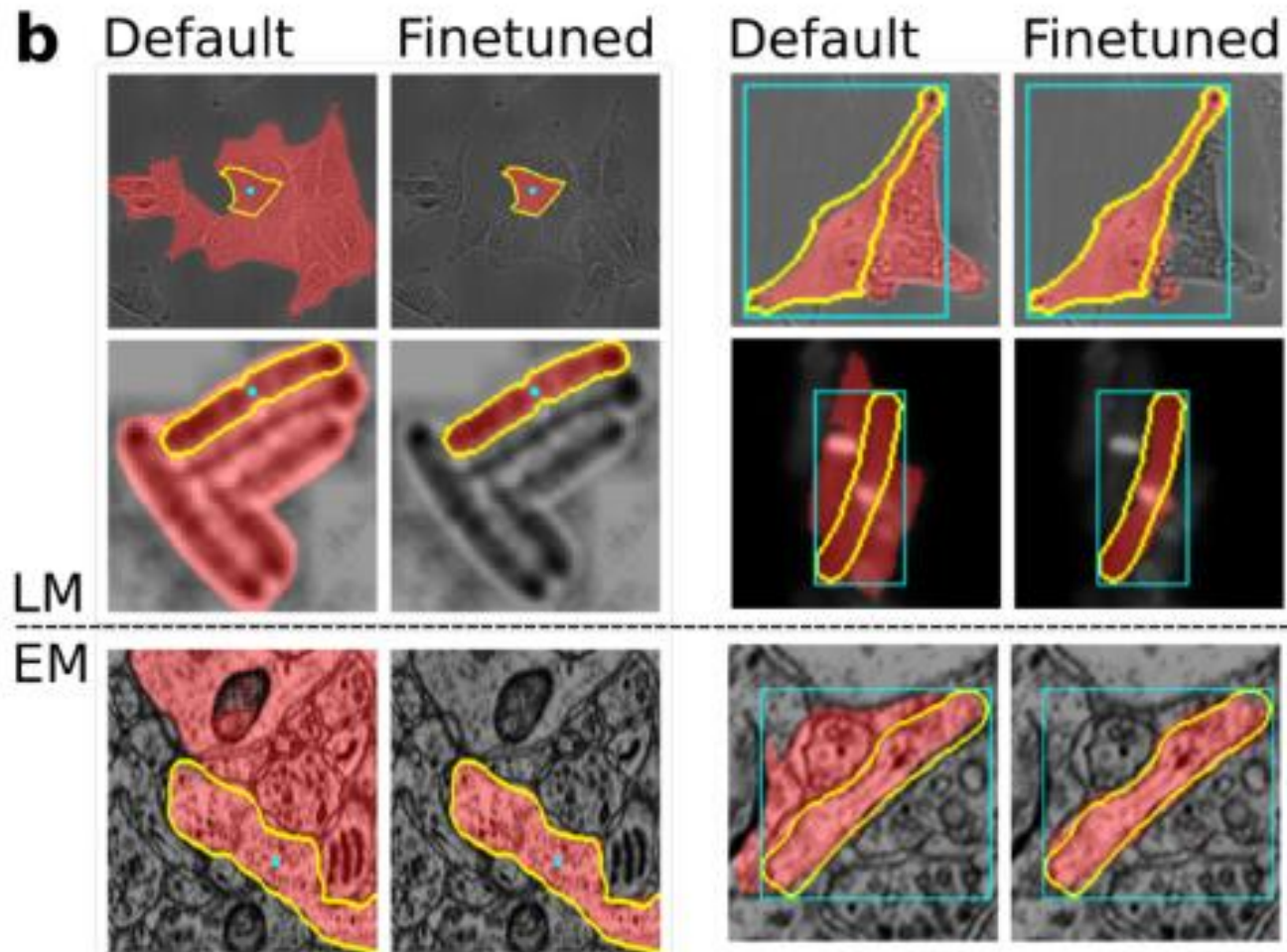


Segment Anything for Microscopy

Real-world scenarios: human-in-the-loop



Segment Anything for Microscopy



Summary: Deep Learning for Bio-image Analysis

- [Convolutional] Neural Networks is a decade old technology that enabled breakthroughs recently.
 - Image Denoising
 - Image Segmentation
- Common scheme: Smart algorithms for processing input/output of neural networks + standard NN architectures
 - ~~Image in, instance segmentation out~~
- Training these models is
 - computationally expensive,
 - needs large amounts of training data (~~single images~~),
 - requires a certain level of [python] expertise
- If Voronoi-Otsu-Labeling does the job, don't dive into deep learning!

Exercises

Robert Haase

Funded by



Bundesministerium
für Bildung
und Forschung

SACHSEN



Diese Maßnahme wird gefördert durch die Bundesregierung
aufgrund eines Beschlusses des Deutschen Bundestages.
Diese Maßnahme wird mitfinanziert durch Steuermittel auf
der Grundlage des von den Abgeordneten des Sächsischen
Landtags beschlossenen Haushaltes.

Exercises

Make noise2void, stardist, cellpose [and micro-sam] work.

Hint: This may screw up your conda environment.

In case of weird errors:

1. Don't panic
2. Recreate you environment
3. Install the thing you want to use, preferably using mamba/conda.

Creating conda environments is not a big deal, just like grabbing a spoon before eating soup.

```
Command Prompt - pip install n2v tensorflow==2.5.1
matplotlib, xarray, google-auth-oauthlib, csbdeep, bioimageio.s
Attempting uninstall: wrapt
Found existing installation: wrapt 1.16.0
Uninstalling wrapt-1.16.0:
  Successfully uninstalled wrapt-1.16.0
Attempting uninstall: typing-extensions
Found existing installation: typing_extensions 4.11.0
Uninstalling typing_extensions-4.11.0:
  Successfully uninstalled typing_extensions-4.11.0
Attempting uninstall: six
Found existing installation: six 1.16.0
Uninstalling six-1.16.0:
  Successfully uninstalled six-1.16.0
Attempting un
Found existi
Uninstalling
  Successful
Attempting un
Found existi
Uninstalling
  Successful
Attempting un
Found existi
Uninstalling
  Successful
Attempting un
Found existi
Uninstalling
  Successful
```

```
Command Prompt
Uninstalling matplotlib-3.8.4:
Successfully uninstalled matplotlib-3.8.4
ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behavior
is the source of the following dependency conflicts:
google-cloud-speech 1.28.1 requires shapely<3.0.0dev, which is not installed.
grpc-google-iam-v1 0.13.0 requires grpcio<2.0.0dev,>=1.44.0, but you have grpcio 1.34.1 which is incompatible.
grpcio-status 1.60.0 requires grpcio<=1.60.0, but you have grpcio 1.34.1 which is incompatible.
grpcio-status 1.60.0 requires protobuf<=4.21.6, but you have protobuf 3.20.3 which is incompatible.
aiosqlite 0.20.0 requires typing_extensions<=4.0, but you have typing_extensions 3.7.4.3 which is incompatible.
apoc 0.12.0 requires numpy<=1.21, but you have numpy 1.19.5 which is incompatible.
devbio-napari 0.10.1 requires numpy<=1.21.4, but you have numpy 1.19.5 which is incompatible.
hdbscan 0.8.33 requires numpy>=1.20, but you have numpy 1.19.5 which is incompatible.
magicgui 0.8.2 requires typing_extensions>=4.1.0, but you have typing_extensions 3.7.4.3 which is incompatible.
napari 0.4.18 requires numpy>=1.21, but you have numpy 1.19.5 which is incompatible.
napari 0.4.18 requires typing_extensions>=4.2.0, but you have typing_extensions 3.7.4.3 which is incompatible.
napari-animation 0.0.0 requires napari>=0.4.19rc5, but you have napari 0.4.18 which is incompatible.
napari-clusters-plotter 0.7.3 requires numpy<=1.23.5,>=1.21, but you have numpy 1.19.5 which is incompatible.
```

```
[1]: import n2v
from skimage.io import imread

-----
ValueError                                Traceback (most recent call last)
Cell In[1], line 2
      1 import n2v
----> 2 from skimage.io import imread

File ~\mambaforge\envs\tea2\lib\site-packages\skimage\_init_.py:122
    118 # We are not importing the rest of the scikit during the build
    119 # process, as it may not be compiled yet
    120 else:
    121     try:
--> 122         from ._shared import geometry
    123         del geometry
    124     except ImportError as e:

File geometry.pyx:1, in init skimage._shared.geometry()

ValueError: numpy.ndarray size changed, may indicate binary incompatibility. Expected 96 from C header,
got 80 from PyObject
```