



CENTER FOR SCALABLE DATA ANALYTICS AND
ARTIFICIAL INTELLIGENCE

Deep Learning for Bio-image Analysis

Robert Haase

Funded by



Bundesministerium
für Bildung
und Forschung



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.

Exam Bio-image Data Science

Day: July 15th 2025

Time: 9:15

Room: S 202

Schillerstraße 6
04109 Leipzig

Quiz

Supervised machine learning requires...

A human supervisor



Labeled data



A binary image



Dimensionality reduction



Quiz

An embedding is ...

N-dimensional
space



Result of
dimensionality
reduction



Result of
clustering



Result of model
training



Deep Learning Basics

Robert Haase

Funded by



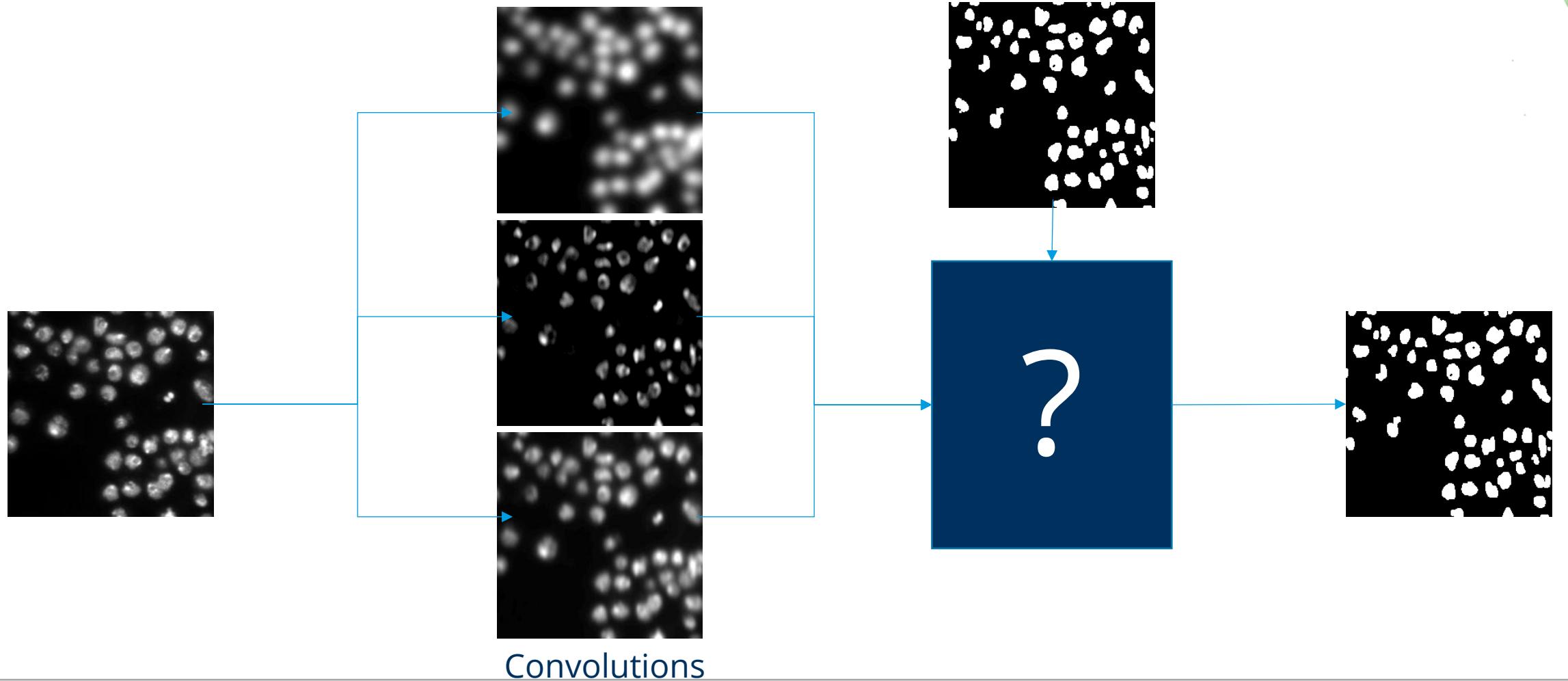
Bundesministerium
für Bildung
und Forschung



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.

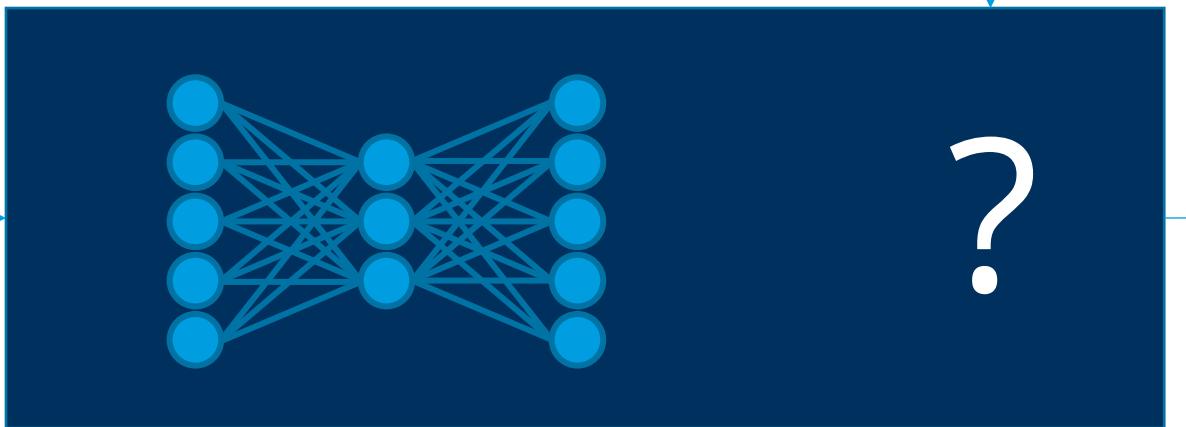
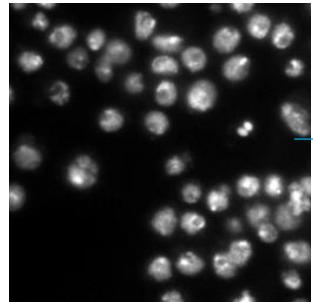
Machine learning for image analysis

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

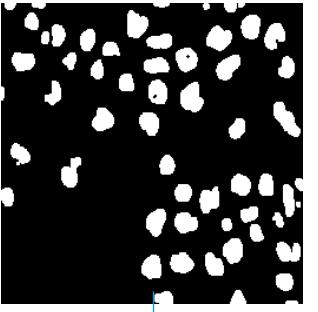


Outlook: Deep learning for image analysis

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

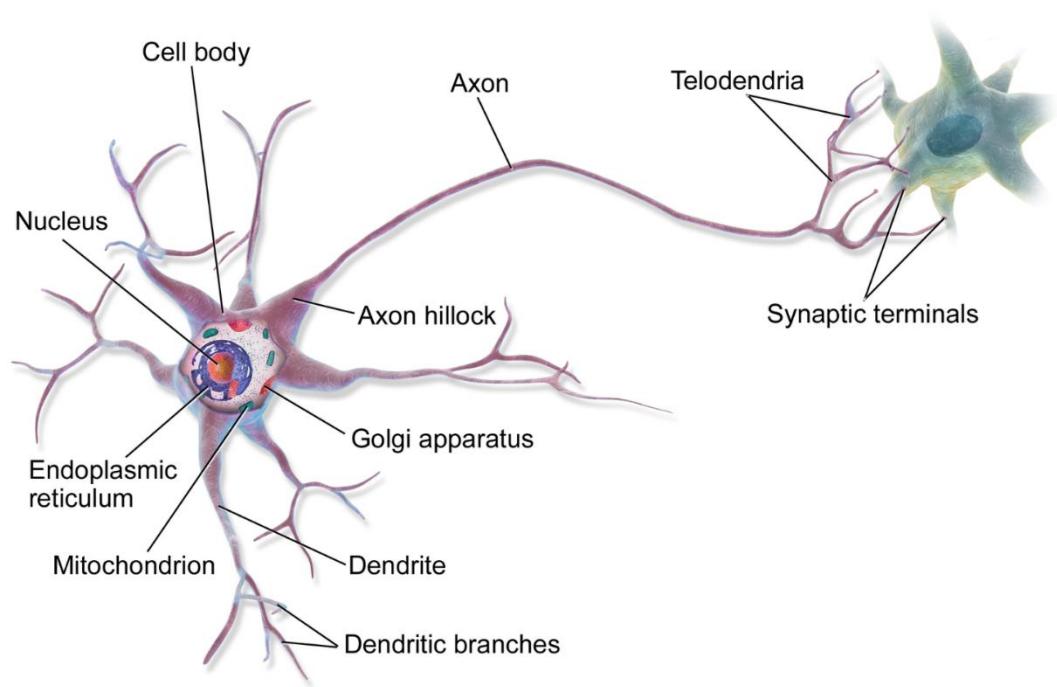


Convolutional neural
networks

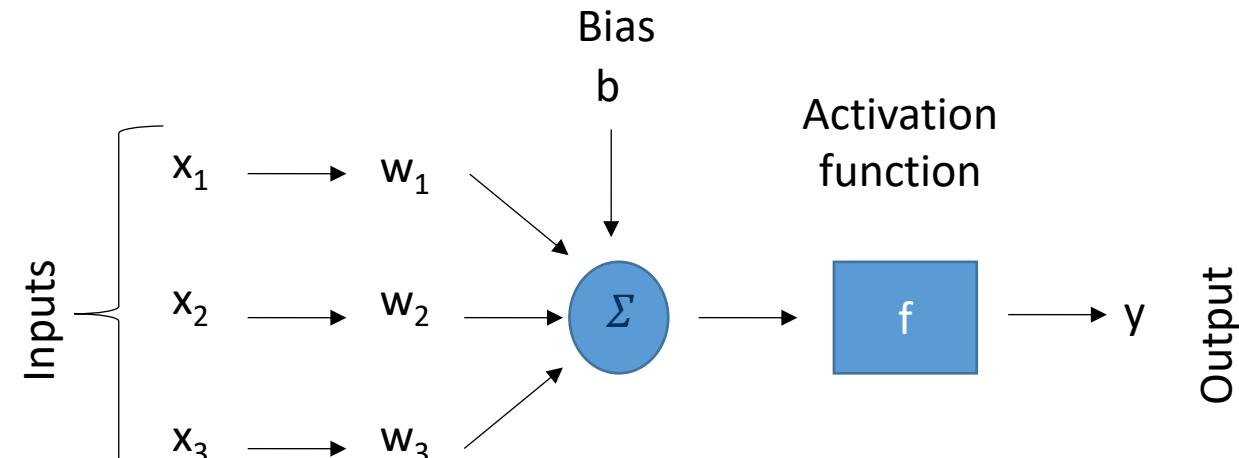


Neural networks

- How biologists see neurons



- How computer scientists see neurons
“perceptron”

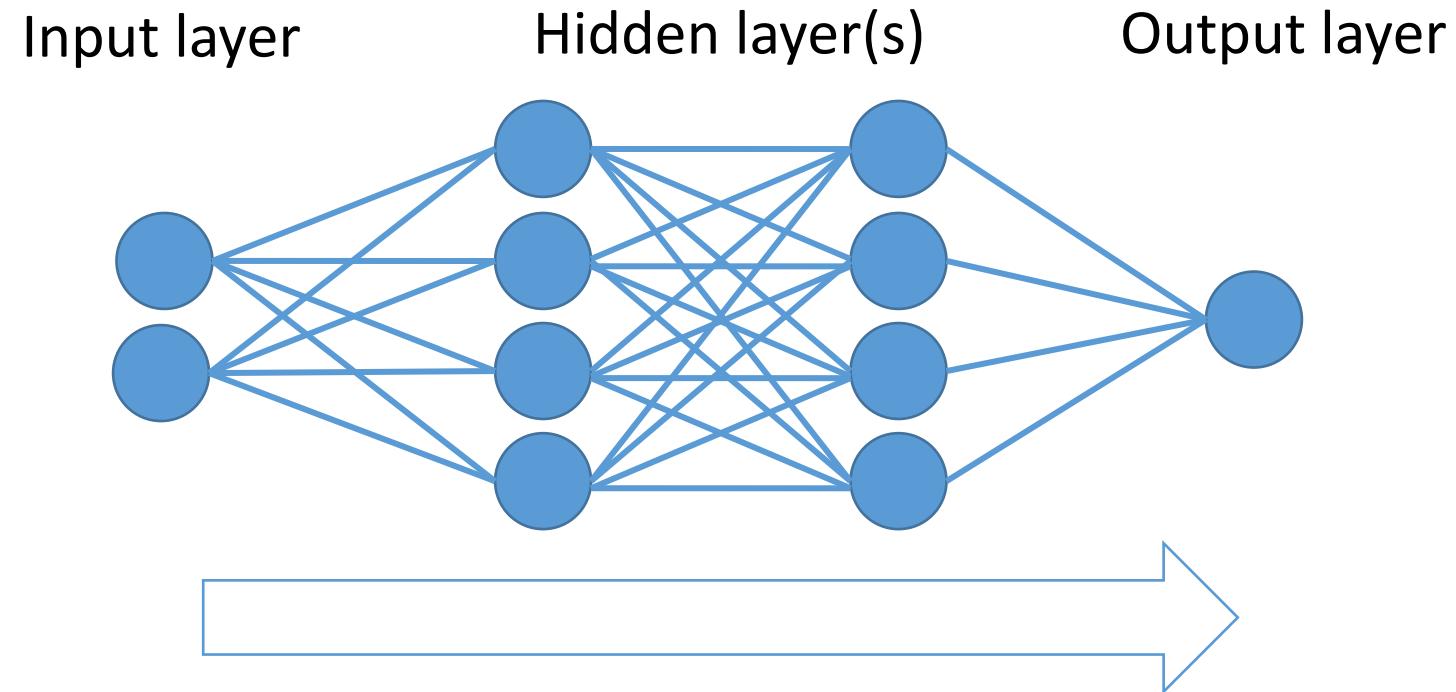


$$y = f(w \cdot |1x_1 + w_2x_2 + w_3x_3 + b|)$$

Neural Networks

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

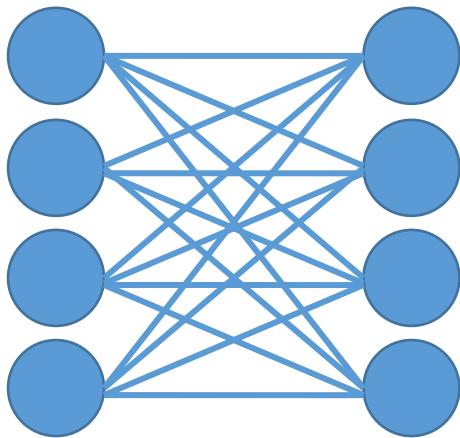
If there are *many* hidden layers, we speak of a *deep neural network*



Convolutional neural networks

- Layer types

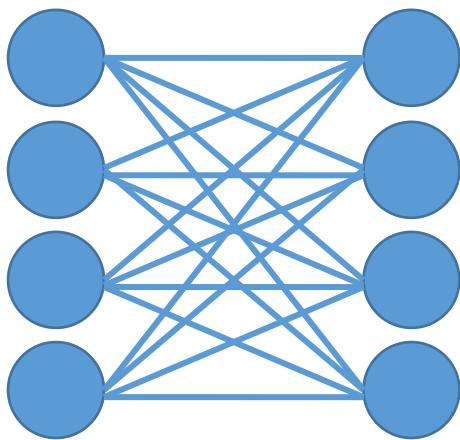
Fully connected layer



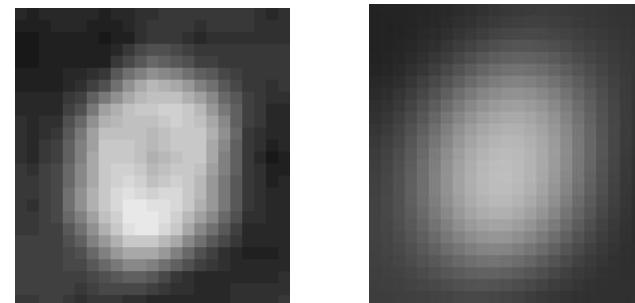
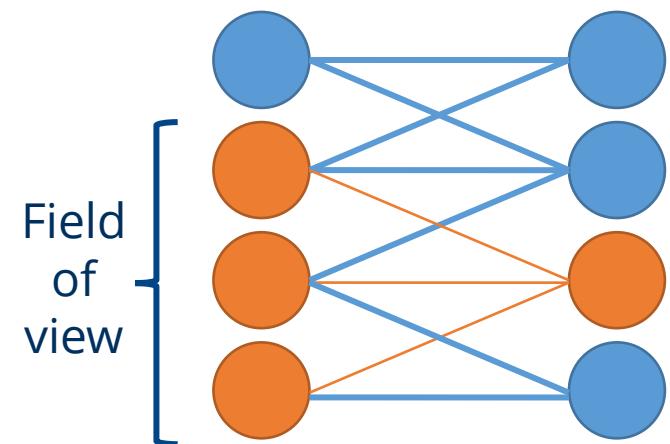
Convolutional neural networks

- Layer types

Fully connected layer



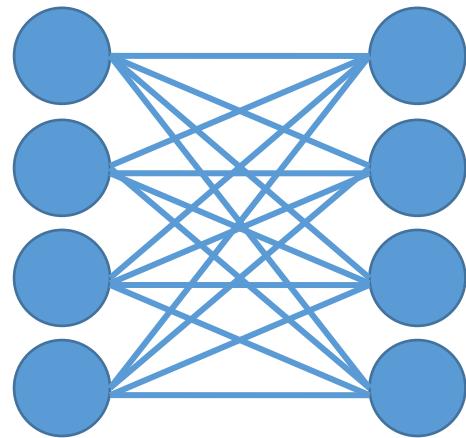
Convolutional layer



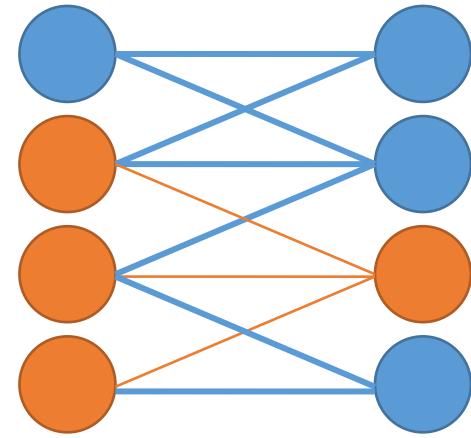
Convolutional neural networks

- Layer types

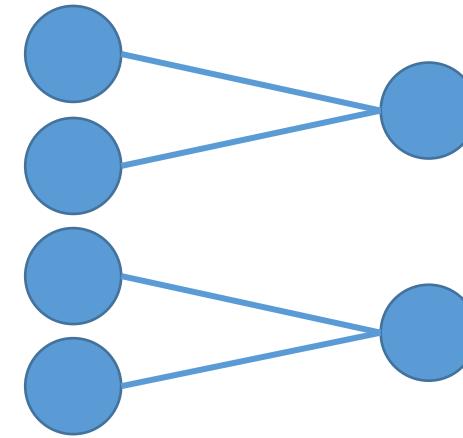
Fully connected layer



Convolutional layer



Pooling layer
("Max pool", "Average pool")



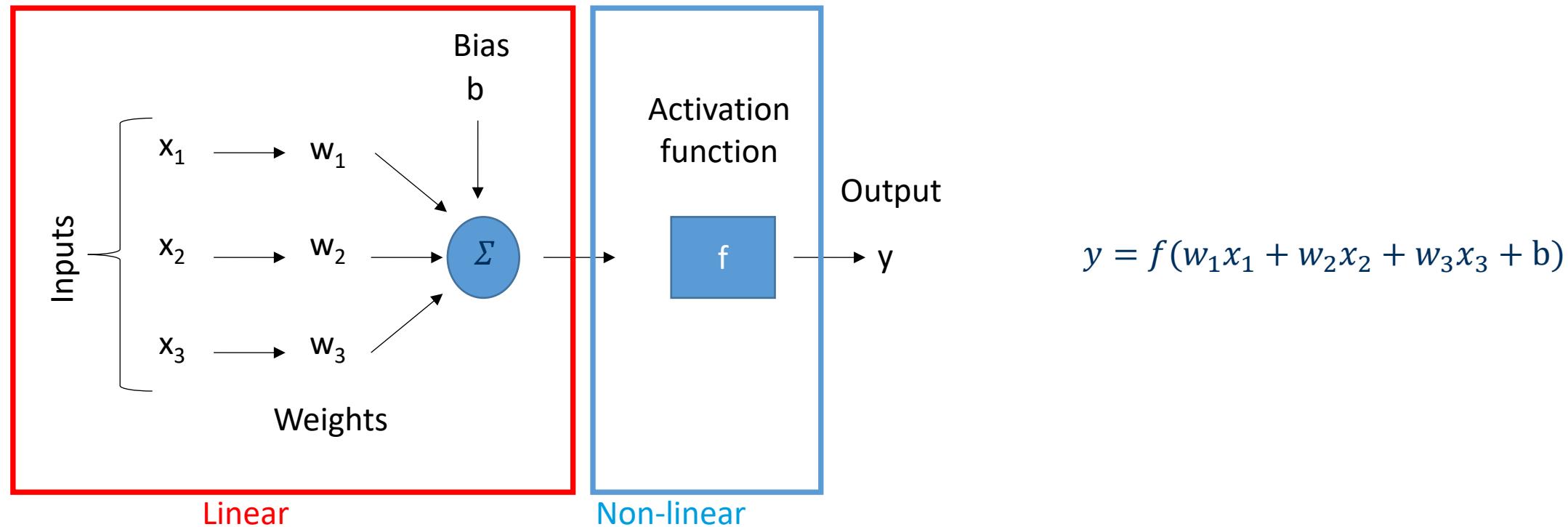
3	15	1	13
9	7	0	10
11	5	5	3
1	8	9	6

Max pooling →

15	13
11	9

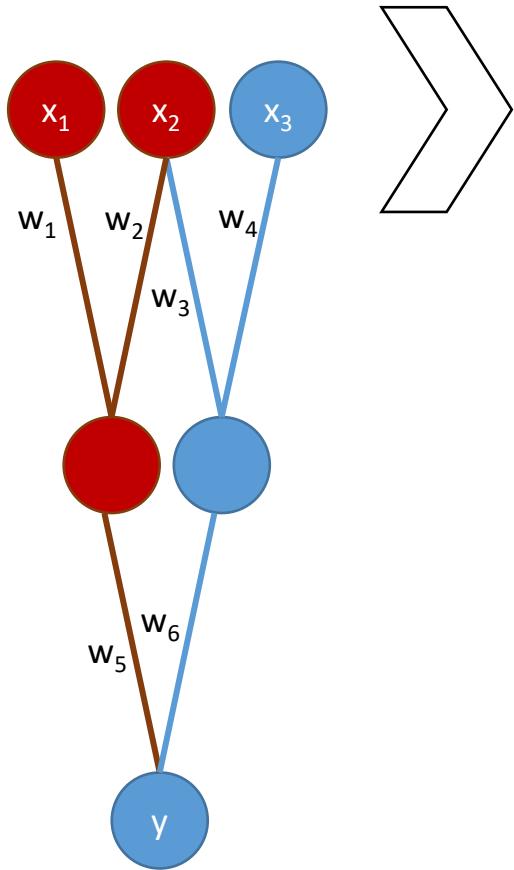
Activation functions

- Introduction of *non-linearity* and *activation functions* enabled what we call *deep-learning* today.

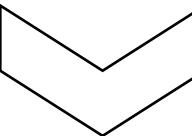


Convolutional neural networks

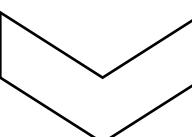
- Assuming we had **no activation functions** in the network, layers can be reduced by eliminating brackets!



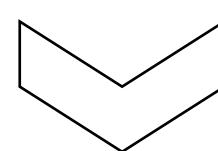
$$y = w_5(w_1x_1 + w_2x_2) + w_6(w_3x_2 + w_4x_3)$$



$$y = w_5w_1x_1 + w_5w_2x_2 + w_6w_3x_2 + w_6w_4x_3$$



$$y = w_5w_1x_1 +$$

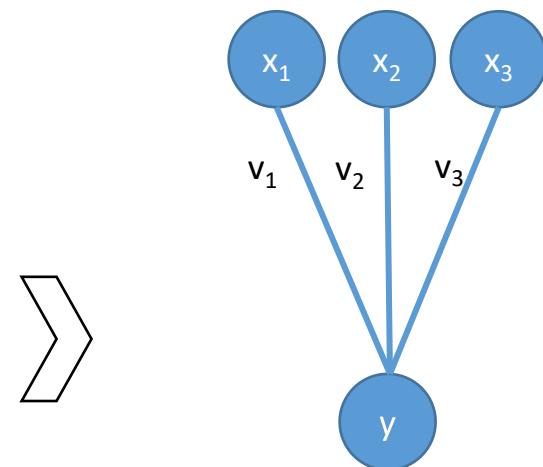


$$v_1 = w_5w_1$$

$$v_2 = w_5w_2 + w_6w_3$$

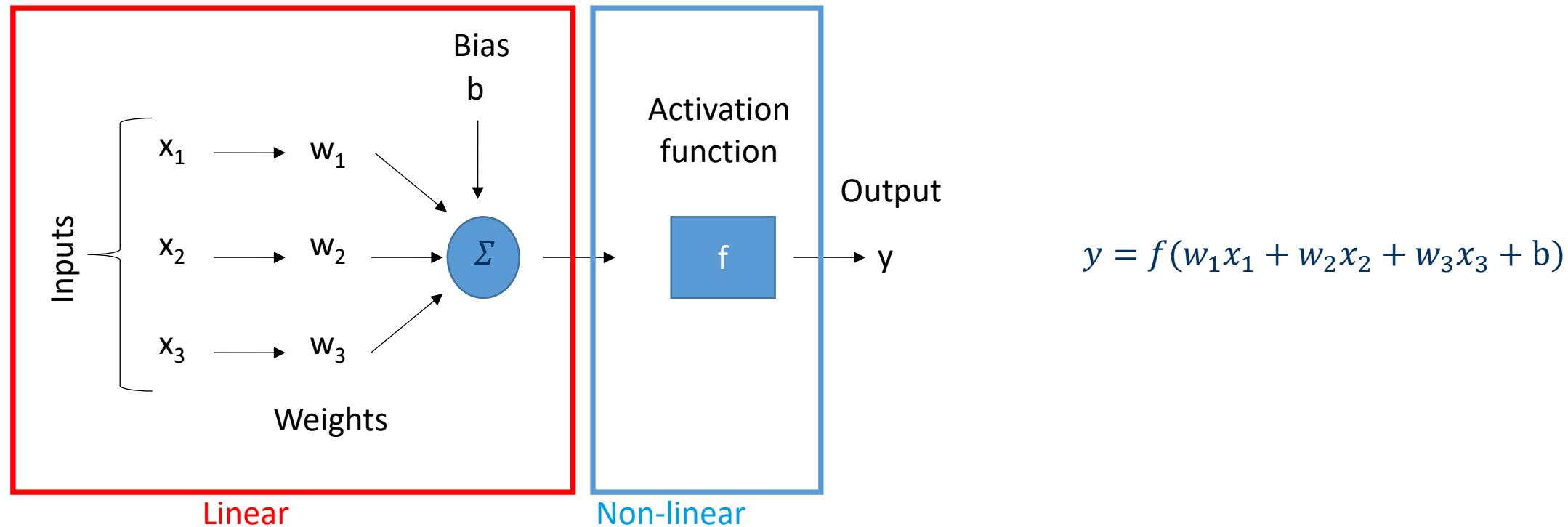
$$v_3 = w_6w_4$$

$$y = v_1x_1 + v_2x_2 + v_3x_3$$



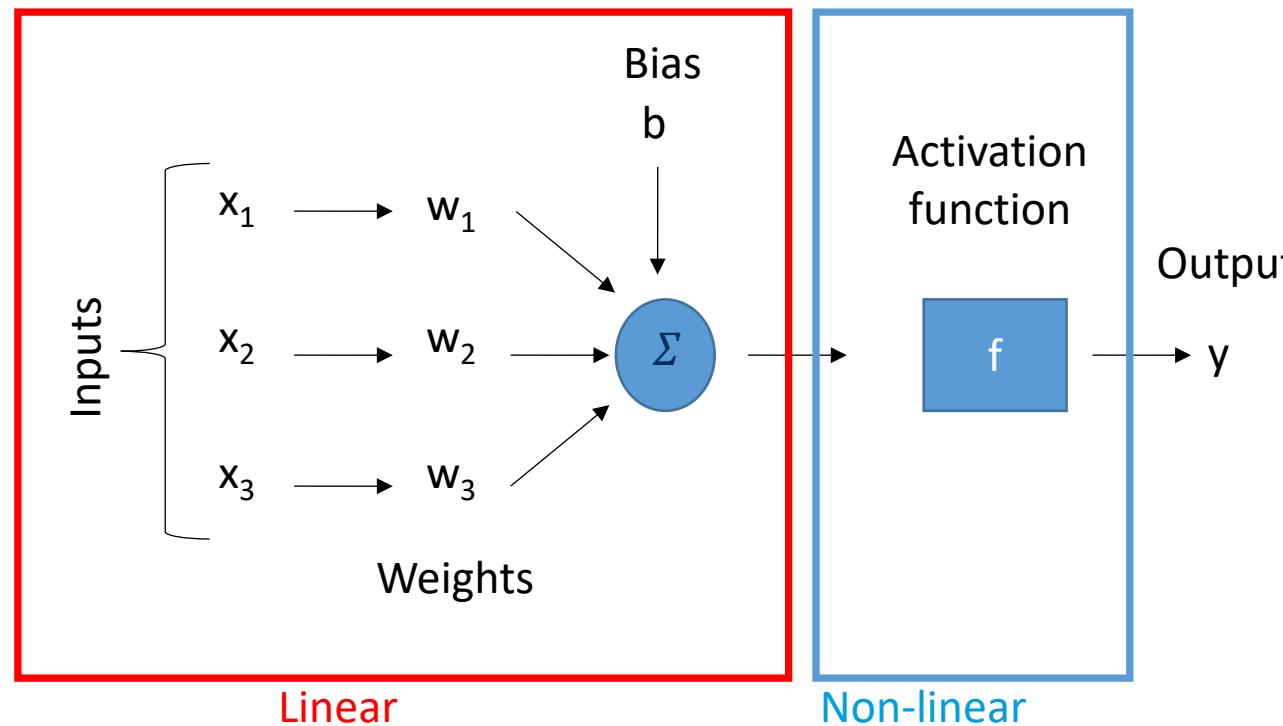
Activation functions

- Introduction of *non-linearity* and *activation functions* enabled what we call *deep-learning* today.



Activation functions

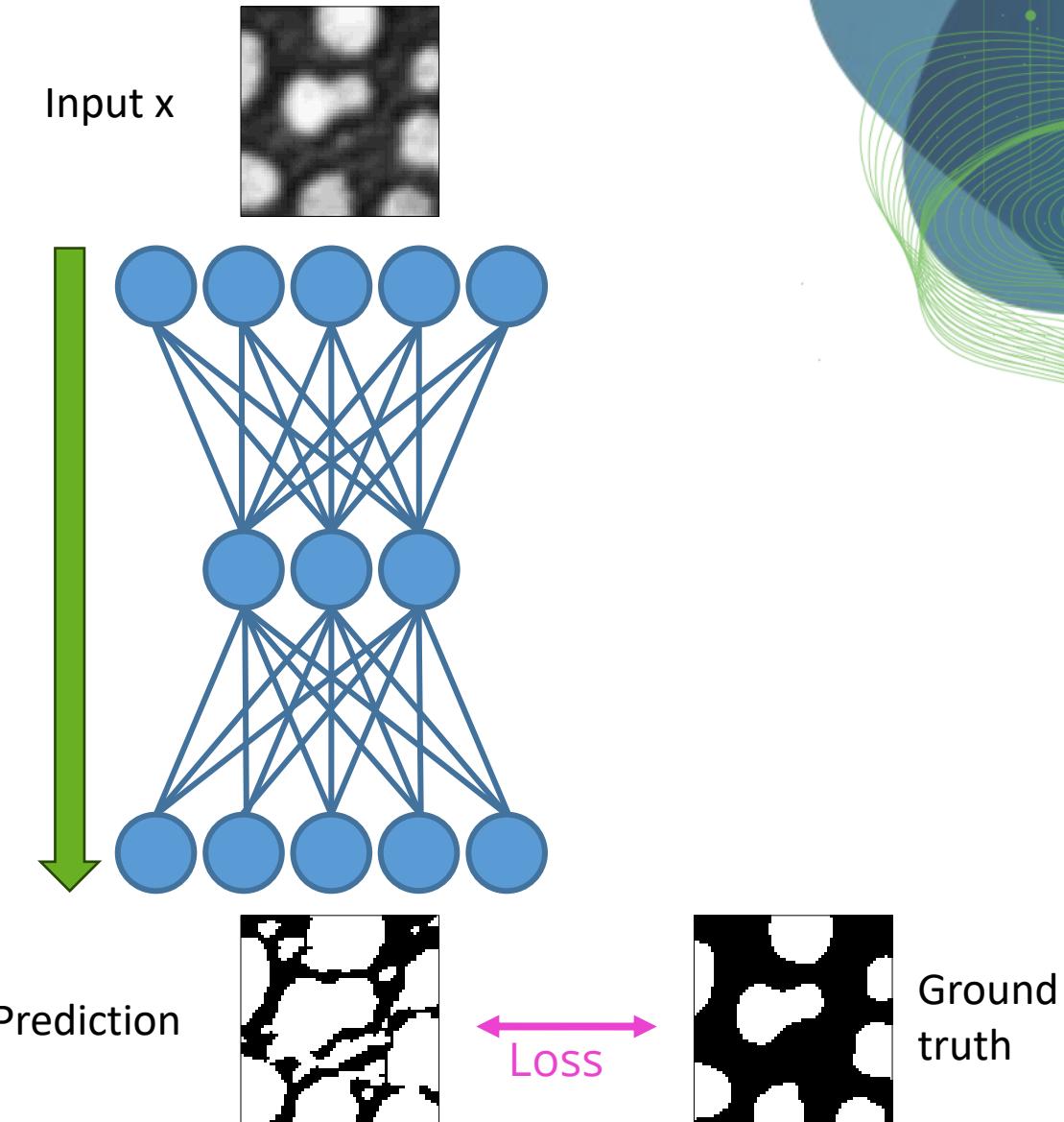
- Introduction of *non-linearity* and *activation functions* enabled what we call *deep-learning* today.



Identity		x
Binary step		$\begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases}$
Logistic, sigmoid, or soft step		$\sigma(x) \doteq \frac{1}{1 + e^{-x}}$
Rectified linear unit (ReLU) ^[8]		$(x)^+ \doteq \begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases} = \max(0, x) = x\mathbf{1}_{x>0}$

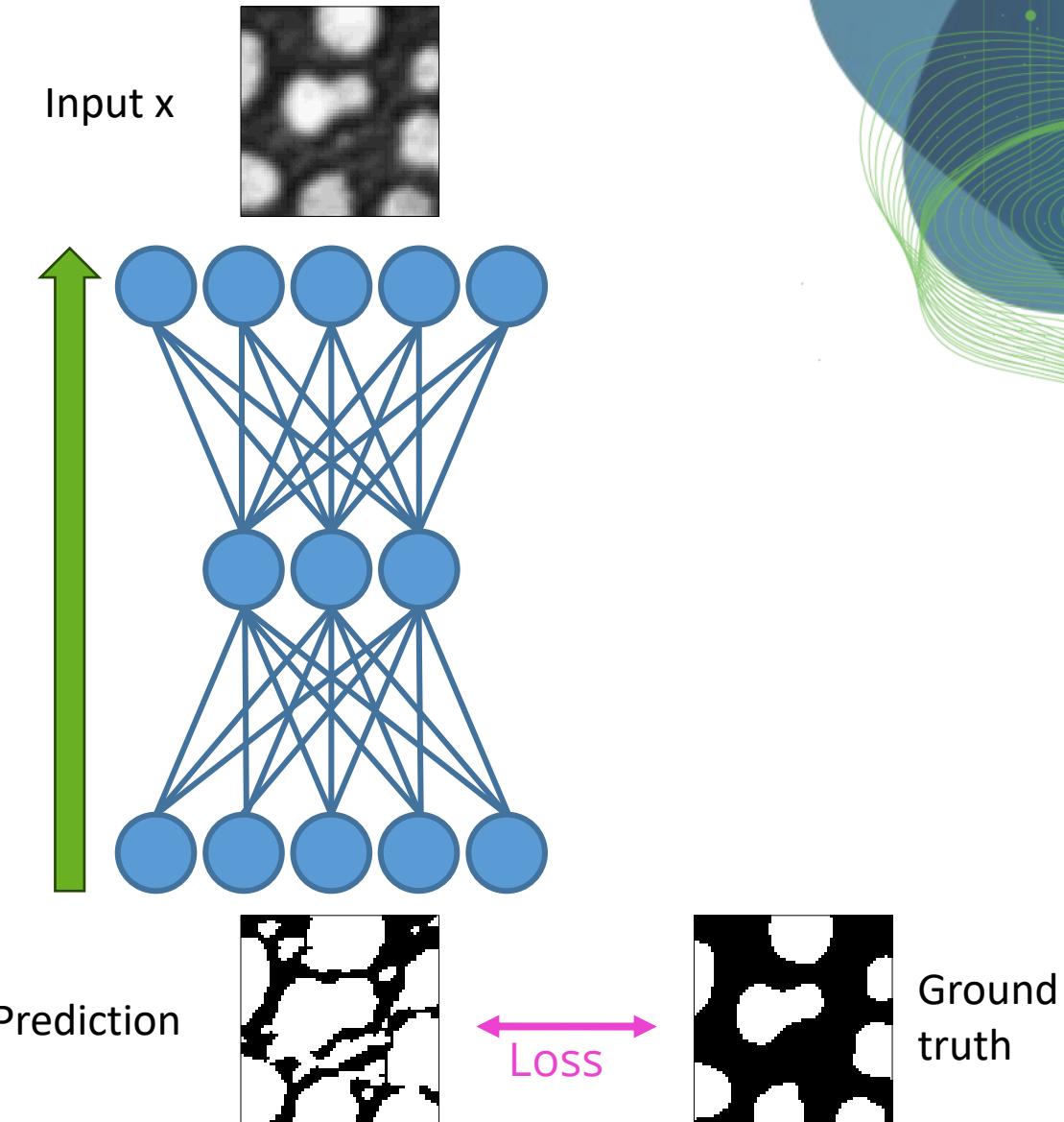
Learning: Back propagation

- Step 0: Initialize the network randomly (weights, bias)
- Step 1: **Forward pass** the input through the network, get an initial prediction
- Step 2: Compare the output with the ground truth, compute the error (loss function)
 - The **loss** function can be *freely* defined.
 - Example: mean squared error
- Step 3: Update weights



Learning: Back propagation

- Updating weights:
 - Set output to the error (per-parameter gradient)
 - **Backward-pass**: add/subtract gradients from weights, to push the network towards giving the right answer.
- Execute the same procedure for next sample
- Execute the same for multiple *epochs*



Model validation

Train dataset (e.g. 80% of the data)

- Used for training directly

Validation dataset (10% of the data)

- After every epoch, determine if model *still generalizes*

Test dataset (10% of the data)

- Final evaluation after training is finished (once)

Underfitting

- A trained model that is not even able to properly process the data it was trained on

Overfitting

- A model that is able to process data it was trained on well
- It processes other data poorly

Underfitting

Loss (lower is better)

Overfitting

Training duration (epochs)

Explainable Deep Learning

Robert Haase

Funded by



Bundesministerium
für Bildung
und Forschung



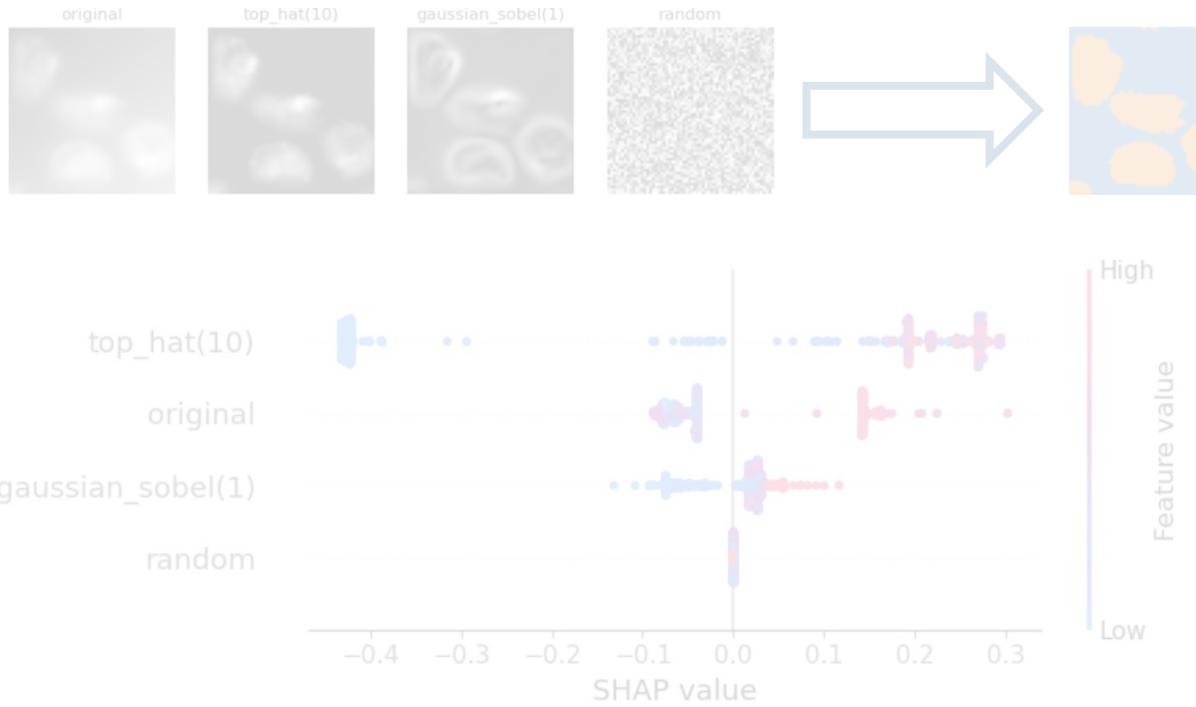
Diese Maßnahme wird gefördert durch die Bundesregierung
aufgrund eines Beschlusses des Deutschen Bundestages.
Diese Maßnahme wird mitfinanziert durch Steuermittel auf
der Grundlage des von den Abgeordneten des Sächsischen
Landtags beschlossenen Haushaltes.

Interpretability of Deep-Learning algorithms

Visualization of intermediate results and their influence on results

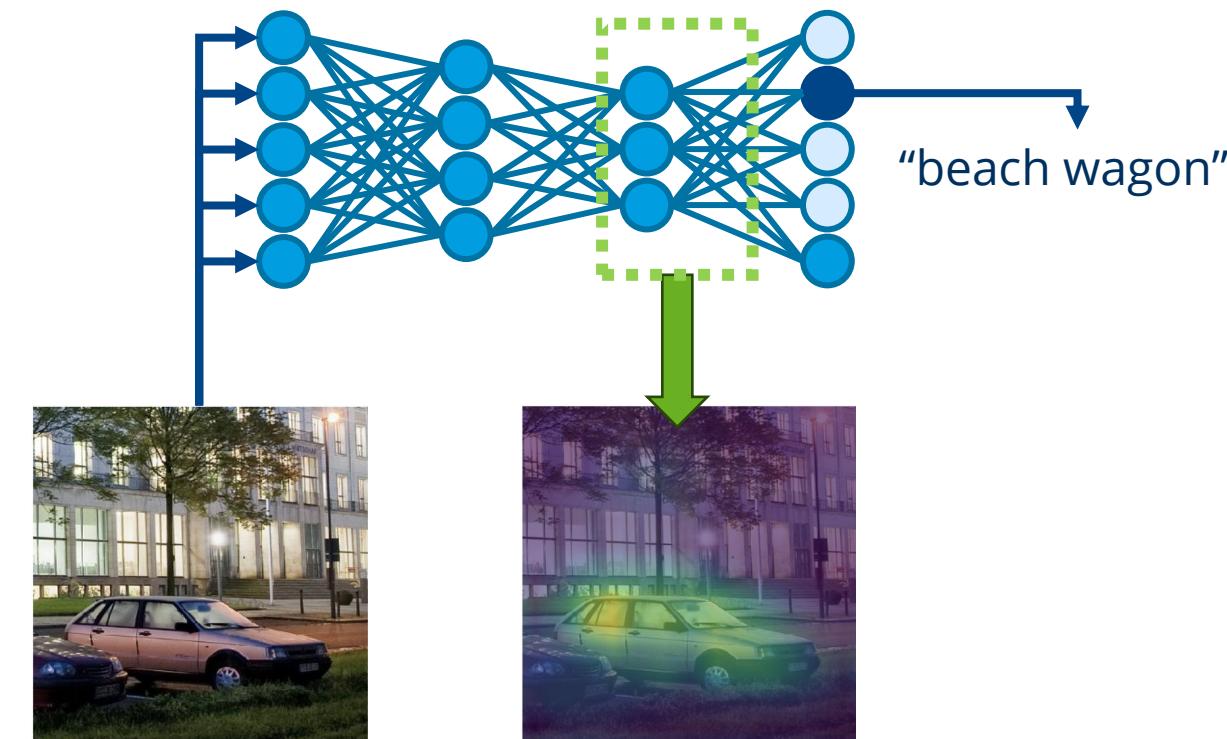
Model-agnostic methods

Example: Shapley's Additive exPlanations (SHAP)



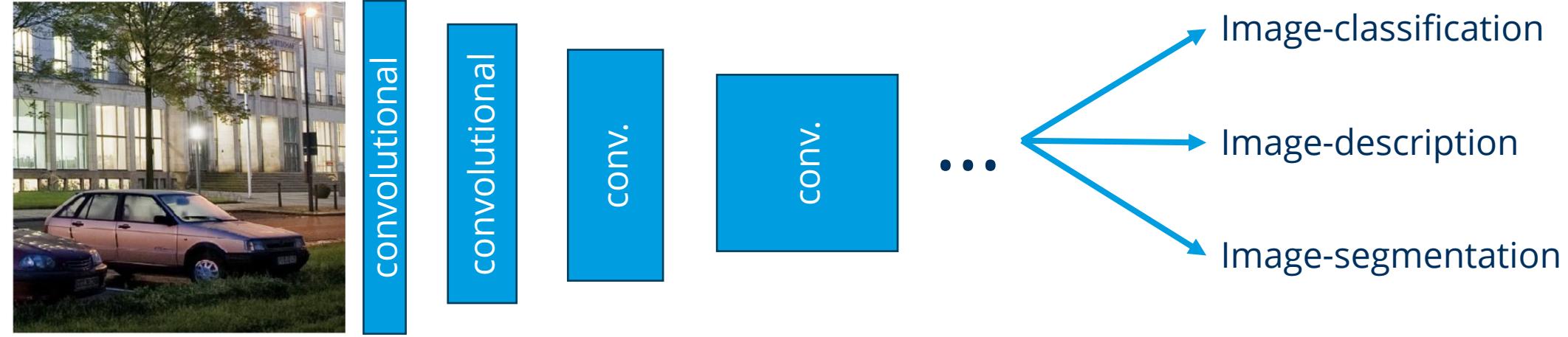
Modell-specific Methods

Example : Gradient Class Activation Maps (Grad-CAM)



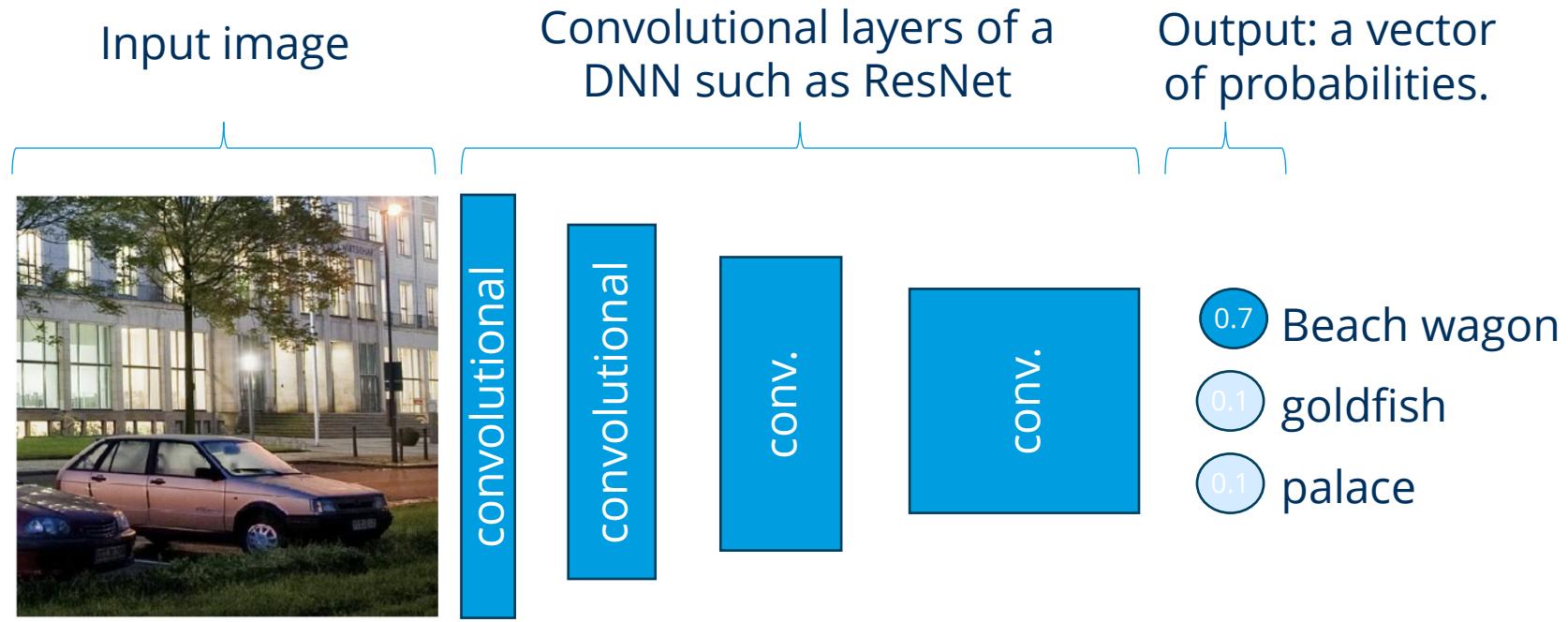
Gradient Class-Activation Maps (Grad-CAM)

- Works only with NN algorithms that first process input data with convolutional layers. (model-specific)
- Independent of right half of the NN (model-agnostic)
- Visualizes intermediate results to make decision-making in the AI system interpretable



Gradient Class-Activation Maps (Grad-CAM)

Is applied to existing network ; no modification of the architecture necessary (post-hoc method).



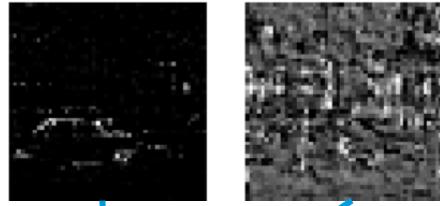
Gradient Class-Activation Maps (Grad-CAM)

Applied to existing network; no adaptation of the architecture necessary (post-hoc method).

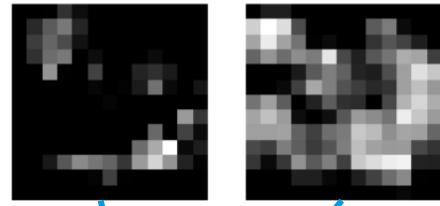
Layer 1 (256, 100, 100)



Layer 2 (512, 50, 50)



Layer 4 (2048, 13, 13)



“2028 feature images
with each 13x13 pixels”

400x400



convolutional

convolutional

conv.

conv.

- Beach wagon
- goldfish
- palace

“embedding”

Quiz

“aspect ratio” is what kind of *feature*?

Feature
image



Image
feature



Object
feature



Feature
object



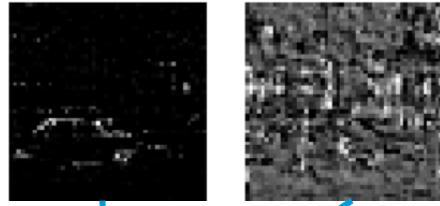
Gradient Class-Activation Maps (Grad-CAM)

Applied to existing network; no adaptation of the architecture necessary (post-hoc method).

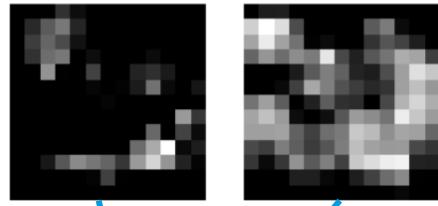
Layer 1 (256, 100, 100)



Layer 2 (512, 50, 50)



Layer 4 (2048, 13, 13)



400x400



convolutional

convolutional

conv.

conv.

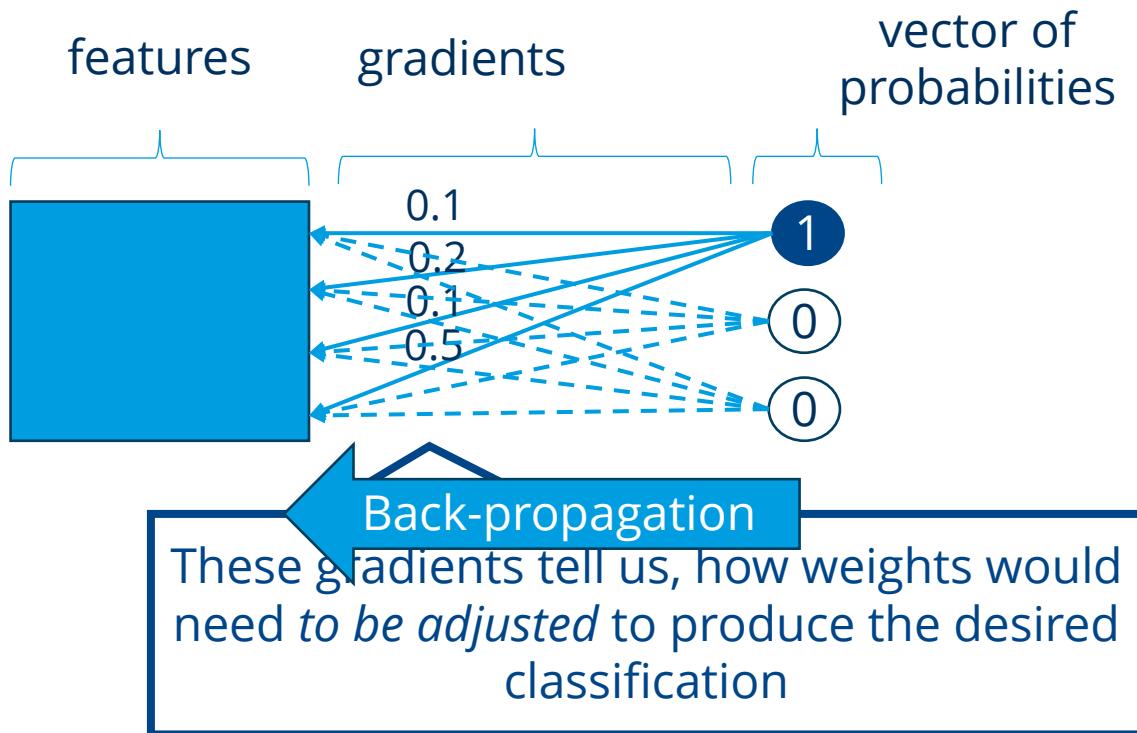
Grad-CAM happens here

- Beach wagon
- goldfish
- palace

None of these images directly says anything about image content.
There is no feature image “Beach wagon”

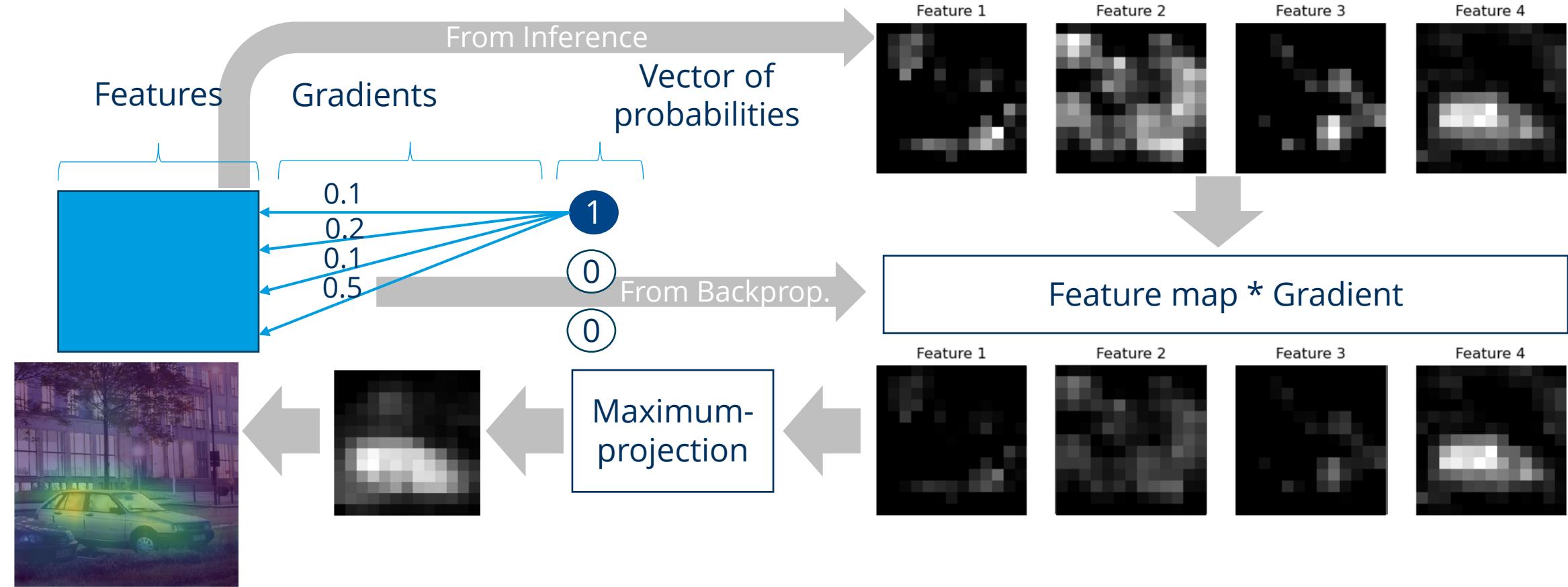
Gradient Class-Activation Maps (Grad-CAM)

Back-propagation of a perfect classification (1,0,0) gives us gradients (weight changes) to improve the classification.



Gradient Class-Activation Maps (Grad-CAM)

Back-propagation of a perfect classification (1,0,0) gives us gradients (weight changes) to improve the classification.



Gradient Class-Activation Maps (Grad-CAM)

Back-propagation of a perfect classification (1,0,0) gives us gradients (weight changes) to improve the classification.

This also works with other possible classifications, e.g. (0,1,0).

“beach waggon”



“palace”



“flagpole”



“great white shark”



Deep Learning for Microscopy Image Processing

Robert Haase

Funded by

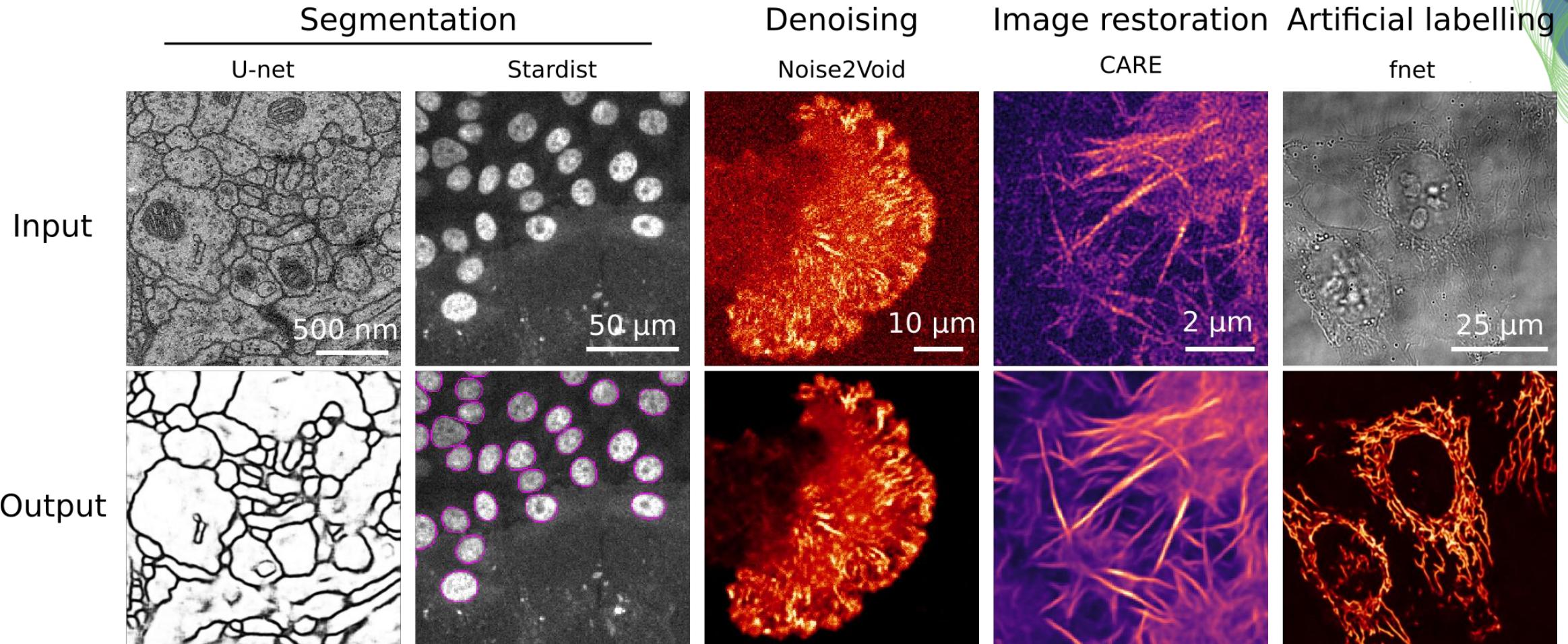


Bundesministerium
für Bildung
und Forschung

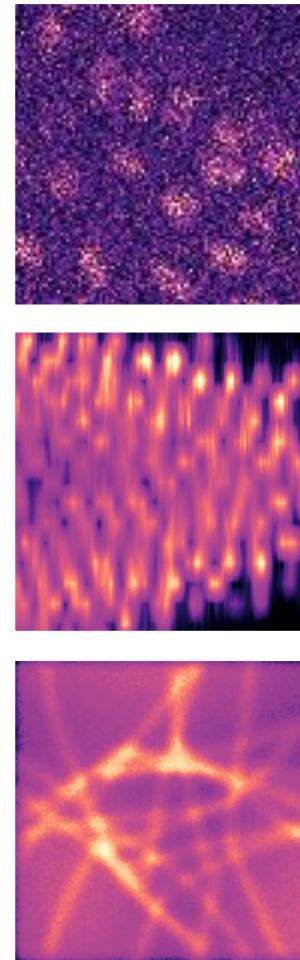
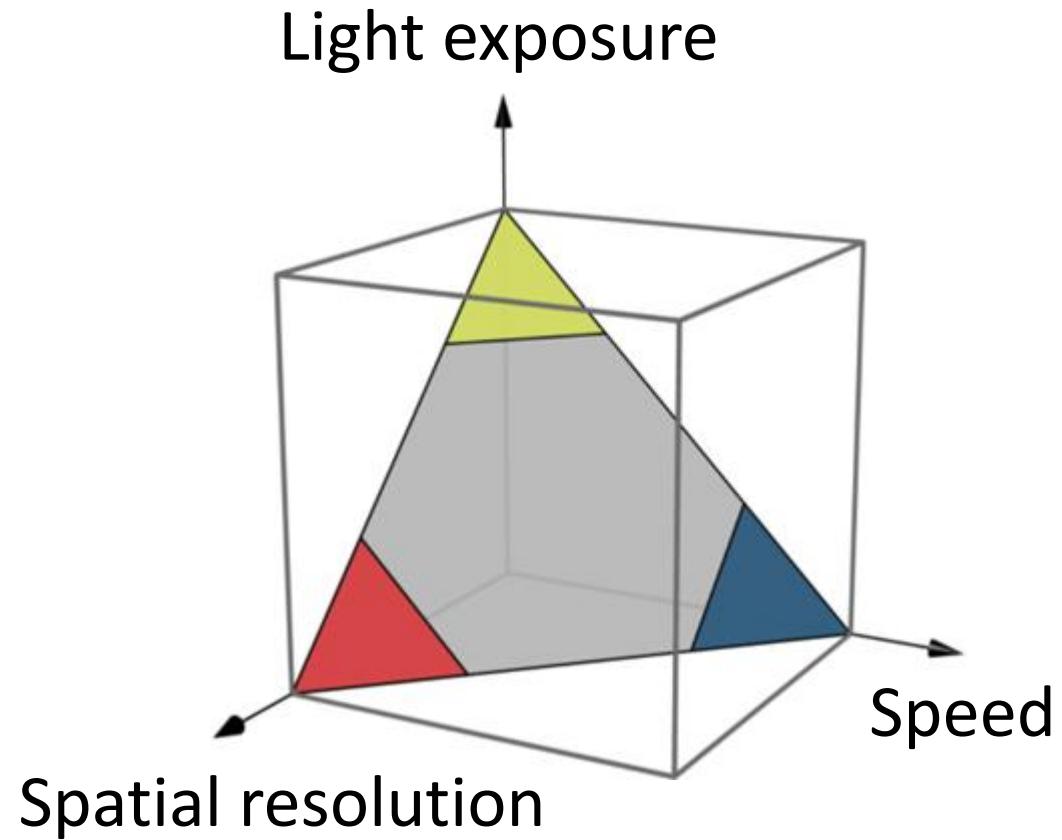


Diese Maßnahme wird gefördert durch die Bundesregierung aufgrund eines Beschlusses des Deutschen Bundestages.
Diese Maßnahme wird mitfinanziert durch Steuermittel auf der Grundlage des von den Abgeordneten des Sächsischen Landtags beschlossenen Haushaltes.

Deep Learning for Microscopy



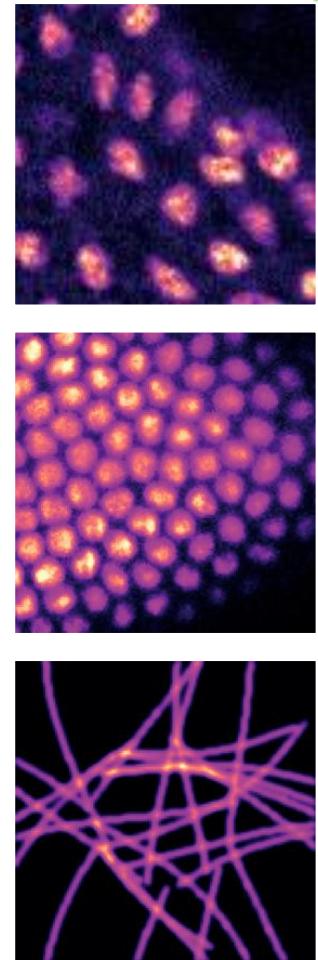
Trade-offs in live-cell imaging



Light Exposure

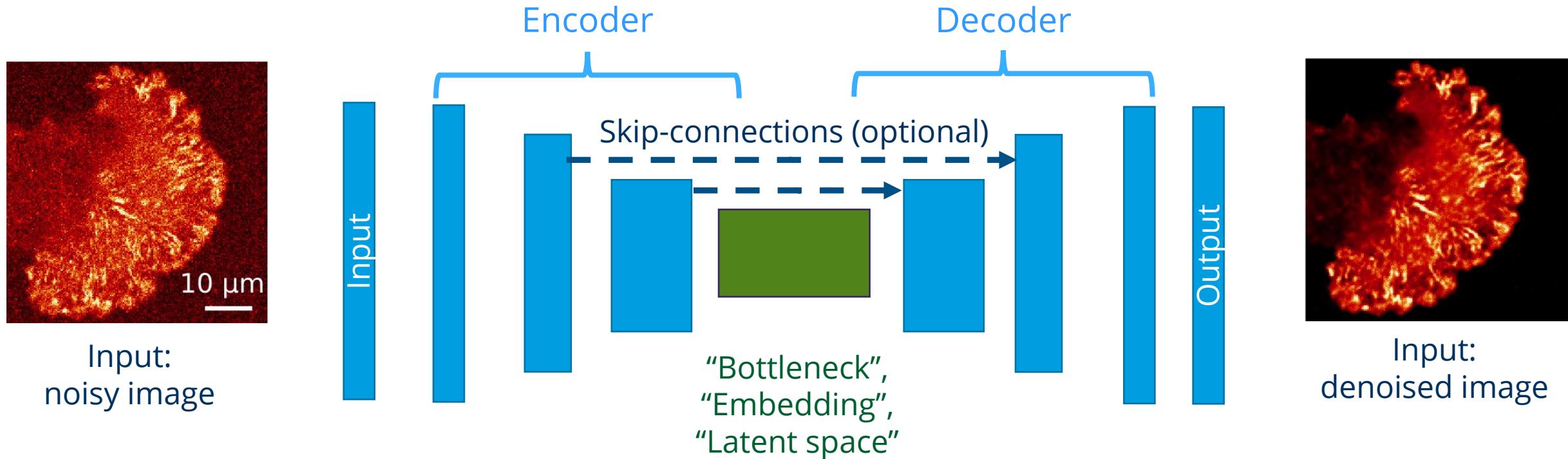
Speed

Spatial Resolution



Traditional architecture: Encoder-Decoder Networks

„Auto-encoder“, „Variational Auto-Encoder“, „U-Net“



Traditional architecture: Encoder-Decoder Networks

U-Net

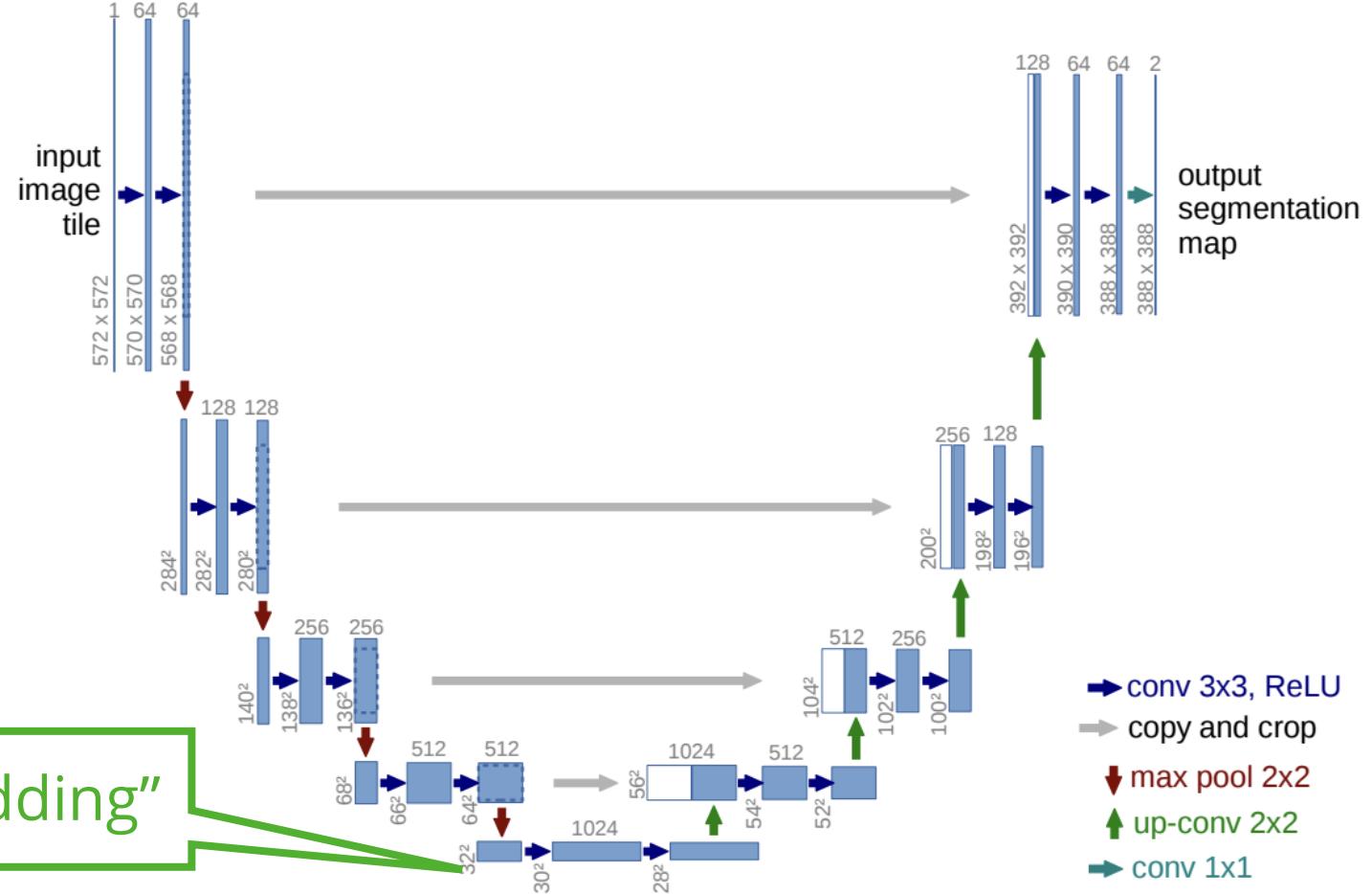


Image denoising: CARE

- 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!

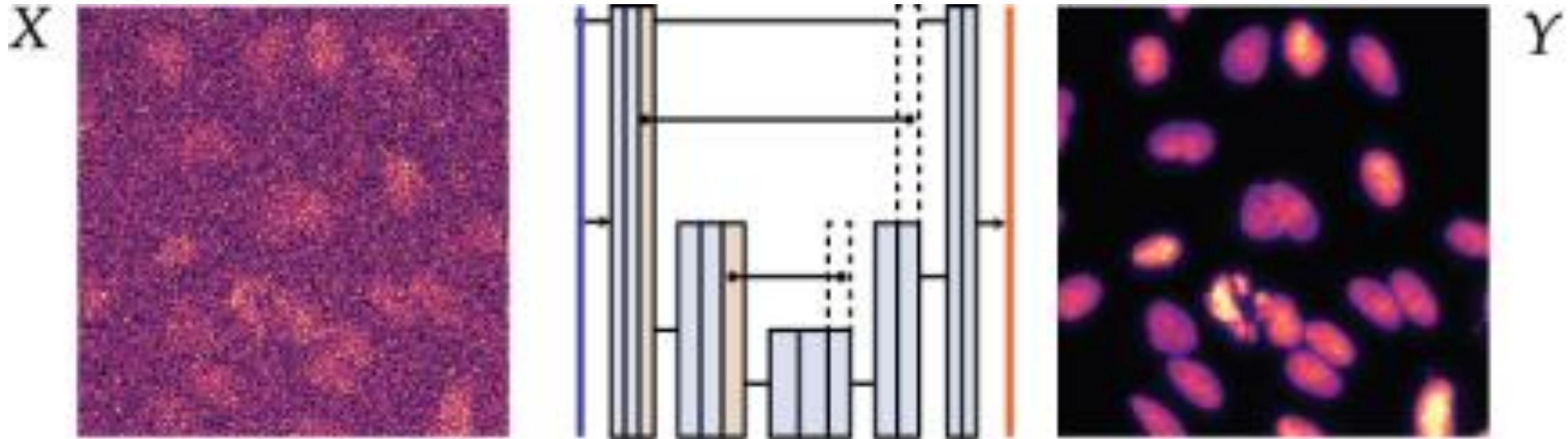
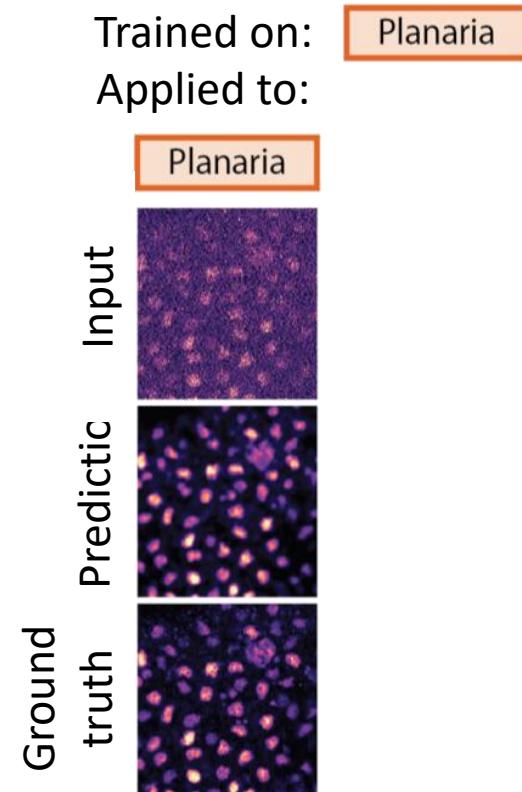


Image denoising: CARE

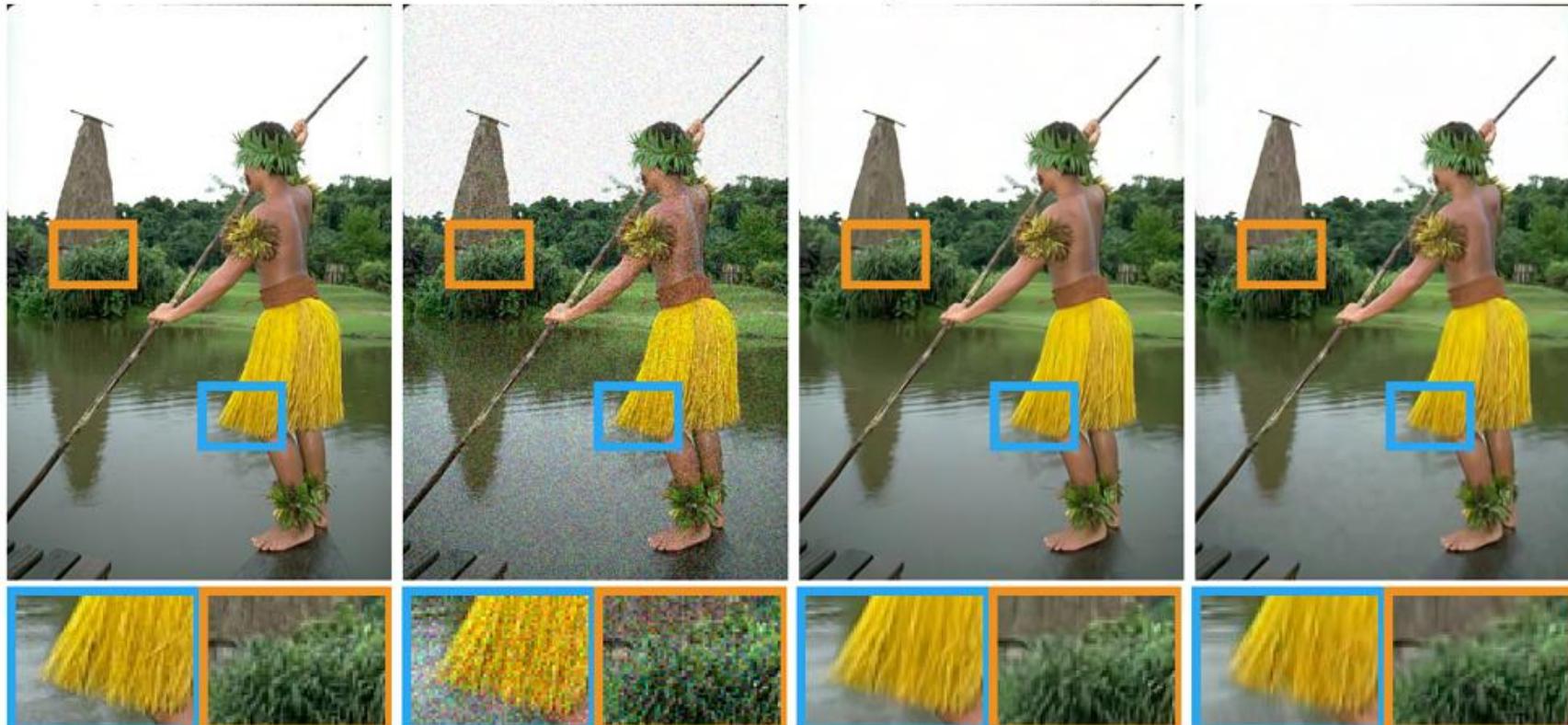
- Important to use on the same conditions/structures/staining that the networks were trained on!



Noise2noise

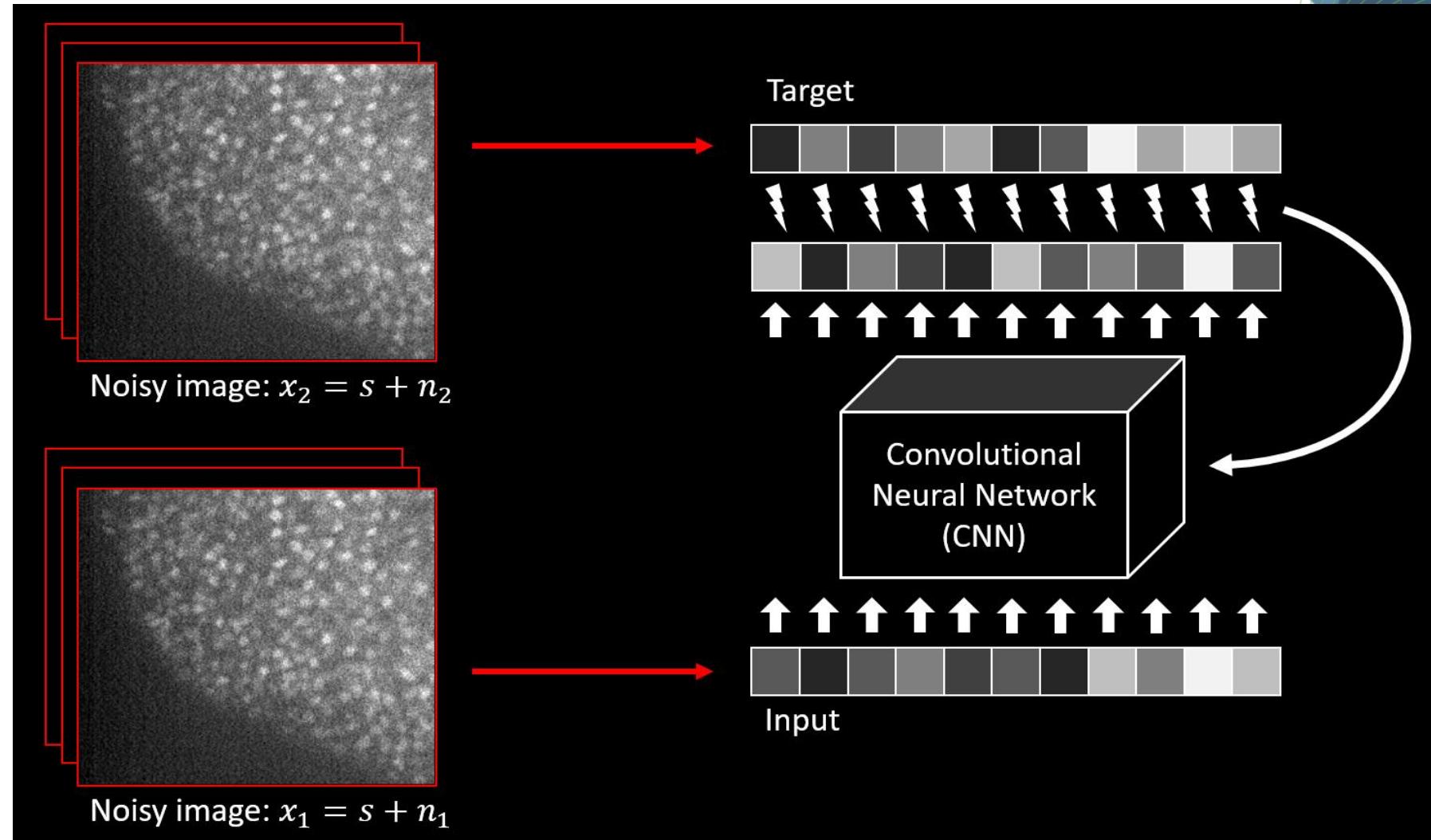
Added noise with gaussian distribution

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



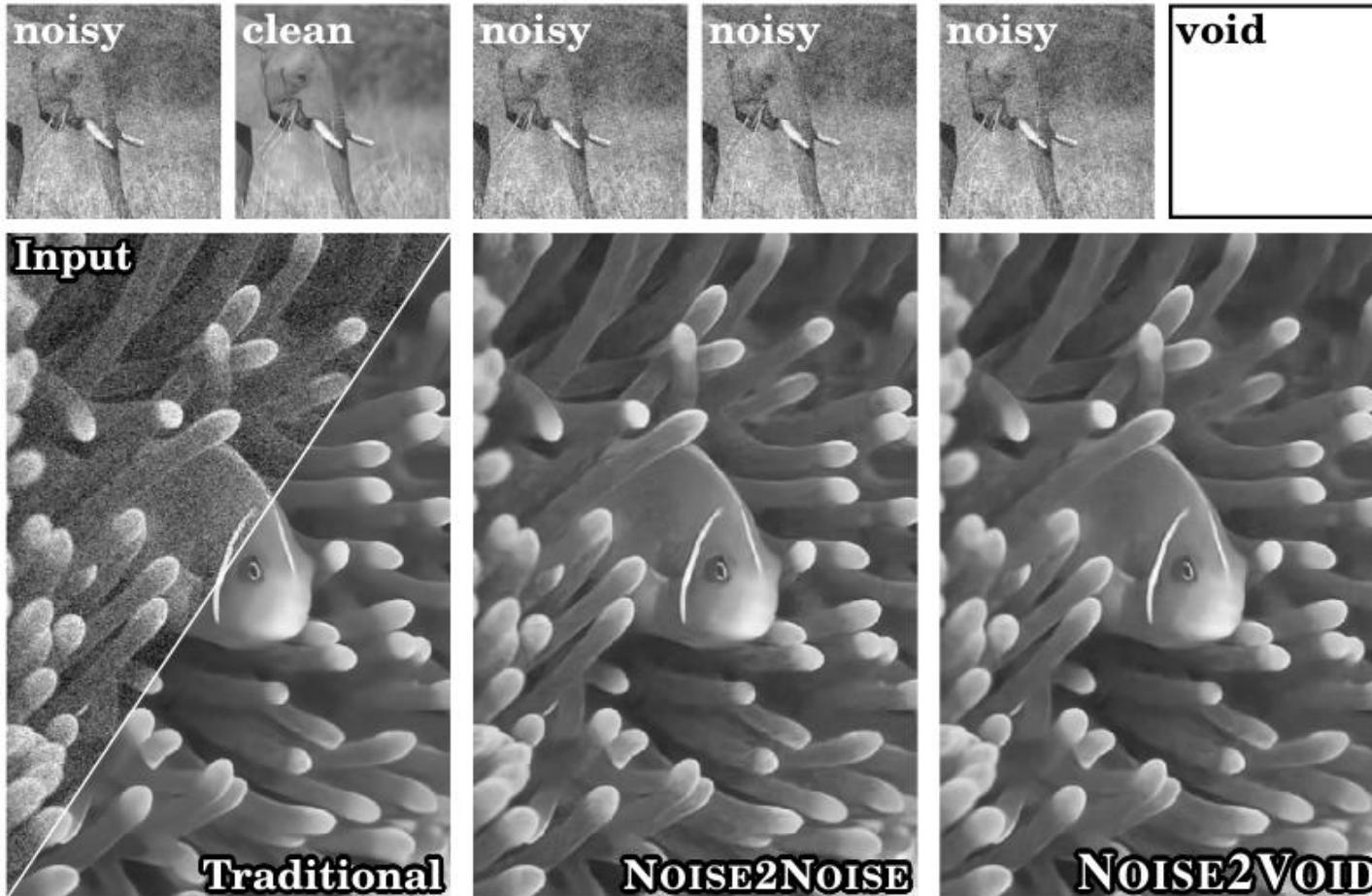
Noise2noise

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



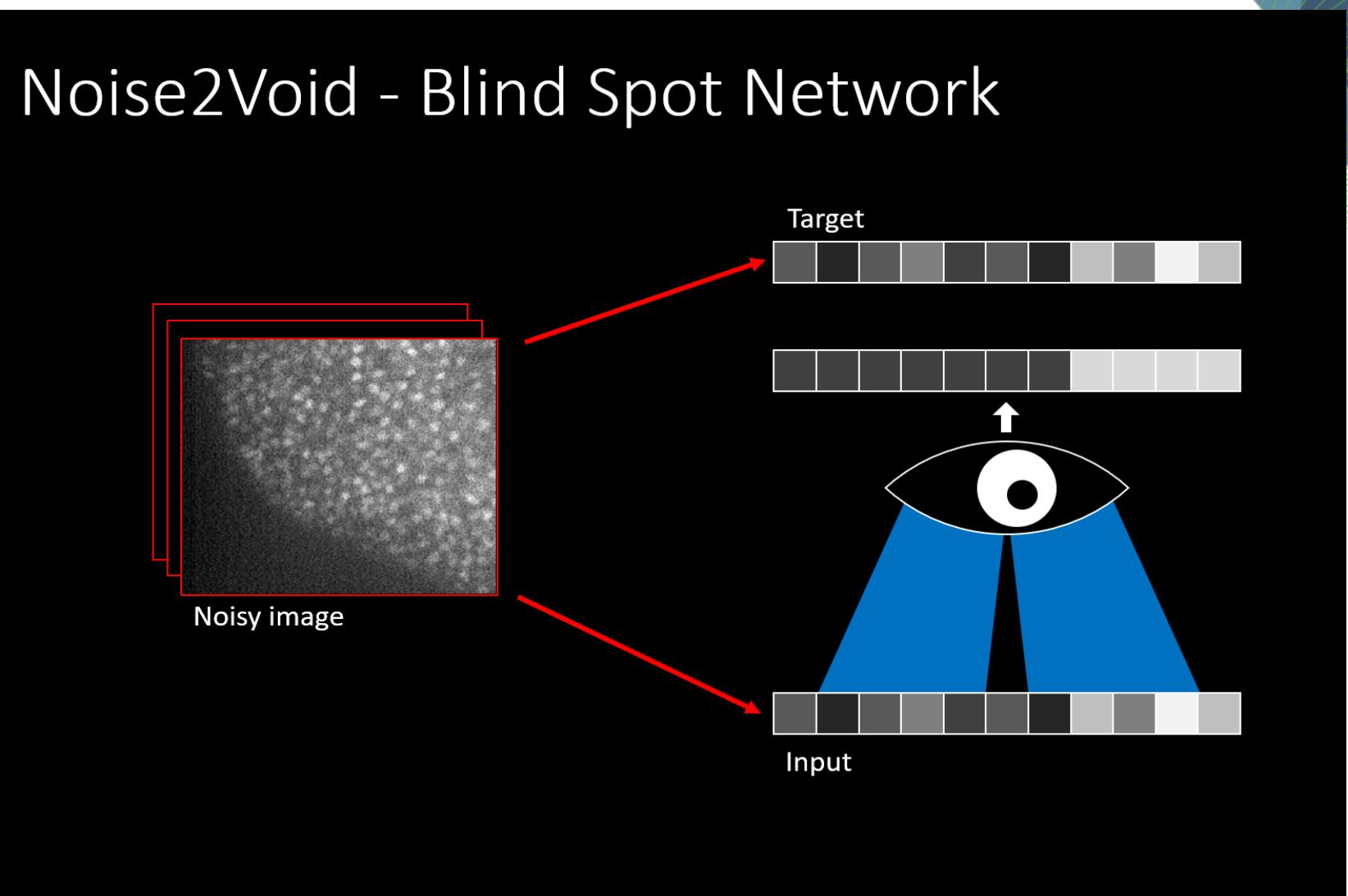
Noise2void

Image denoising without image pairs



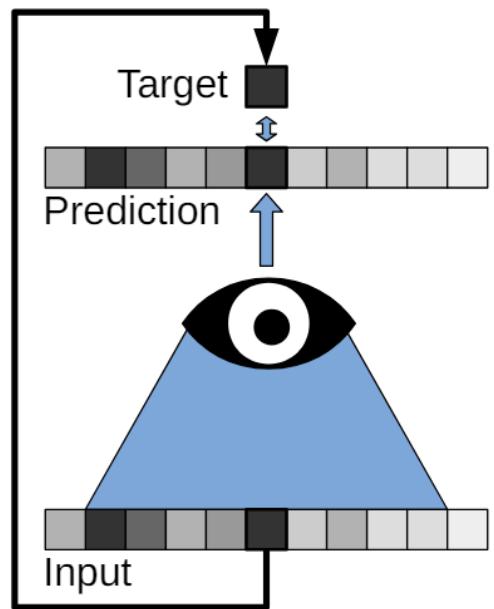
Noise2void

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

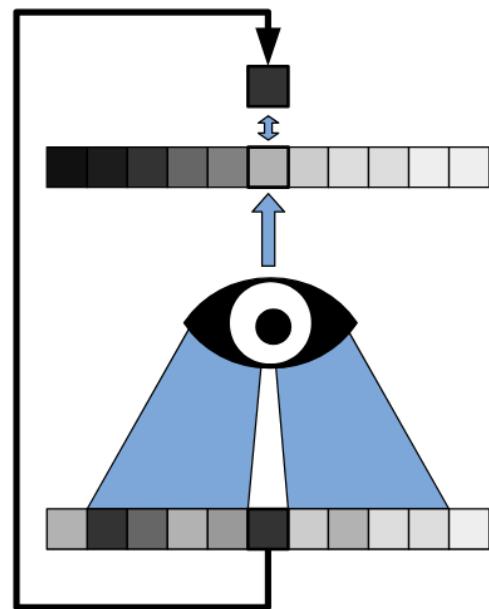


Noise2void

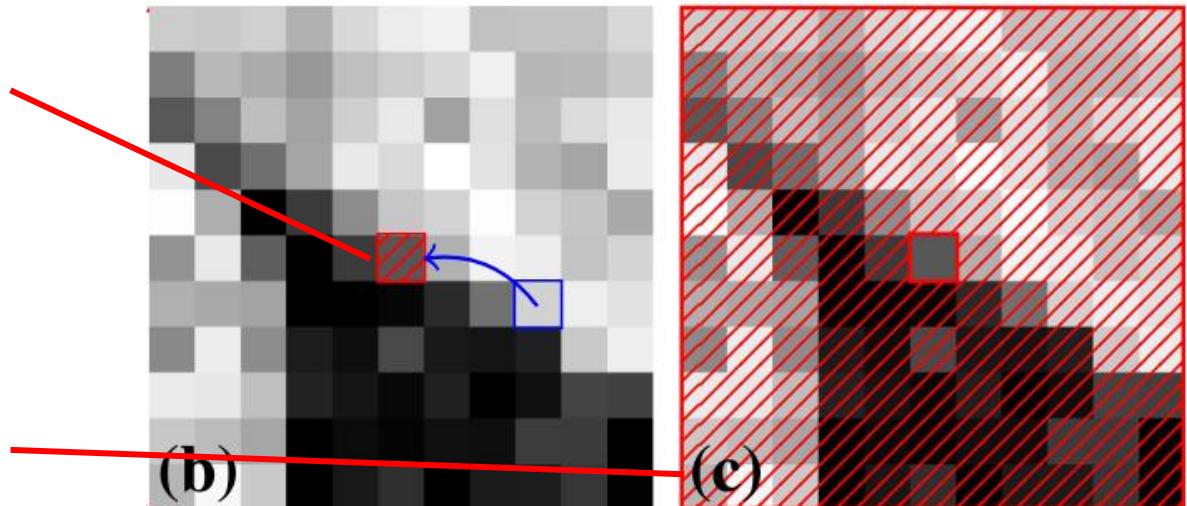
„Blind-spot-network“



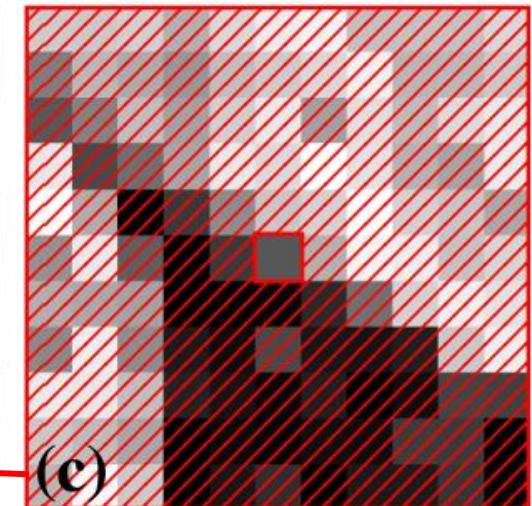
(a)



(b)



(b)



(c)

Noise2void

Napari-plugin

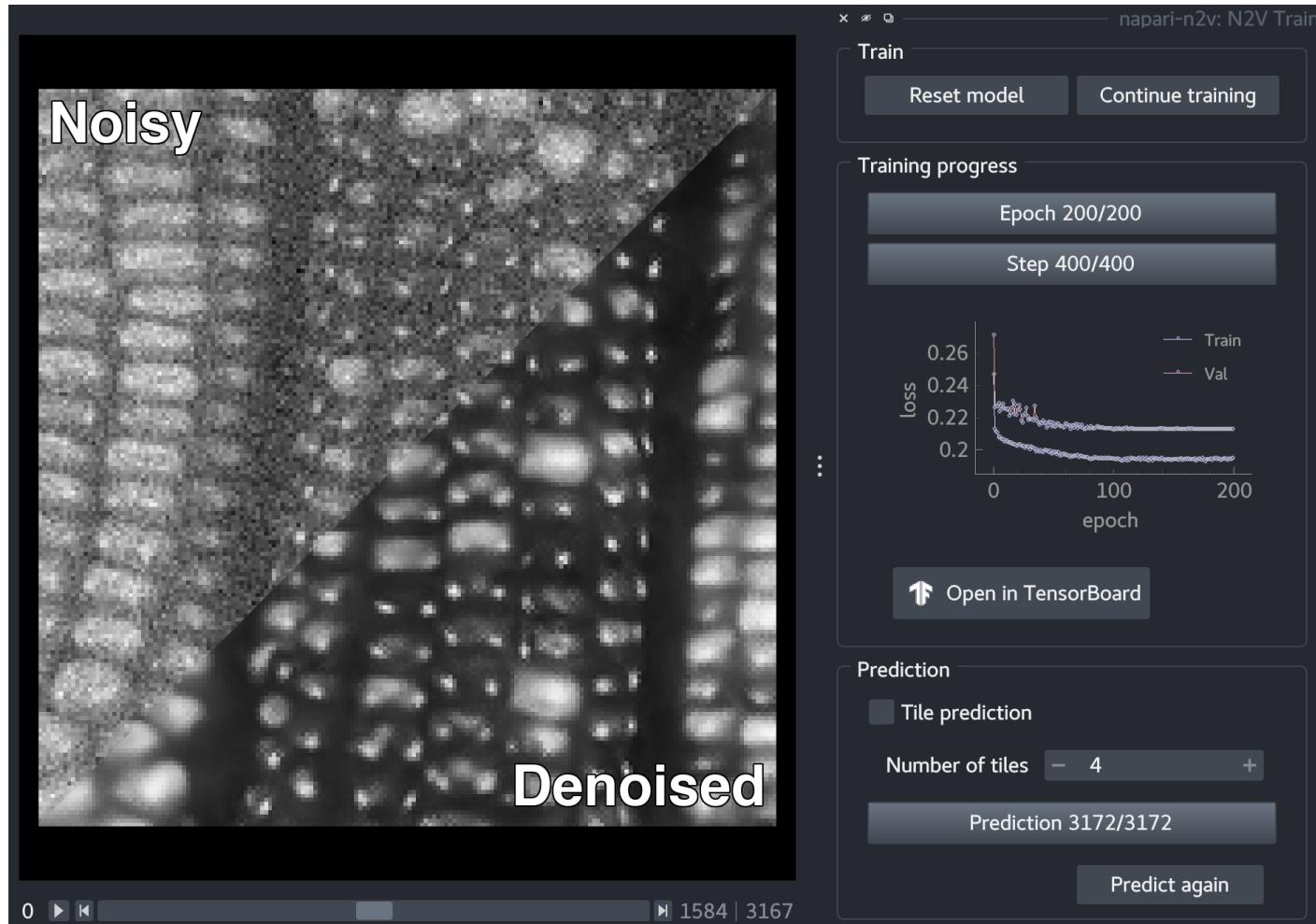
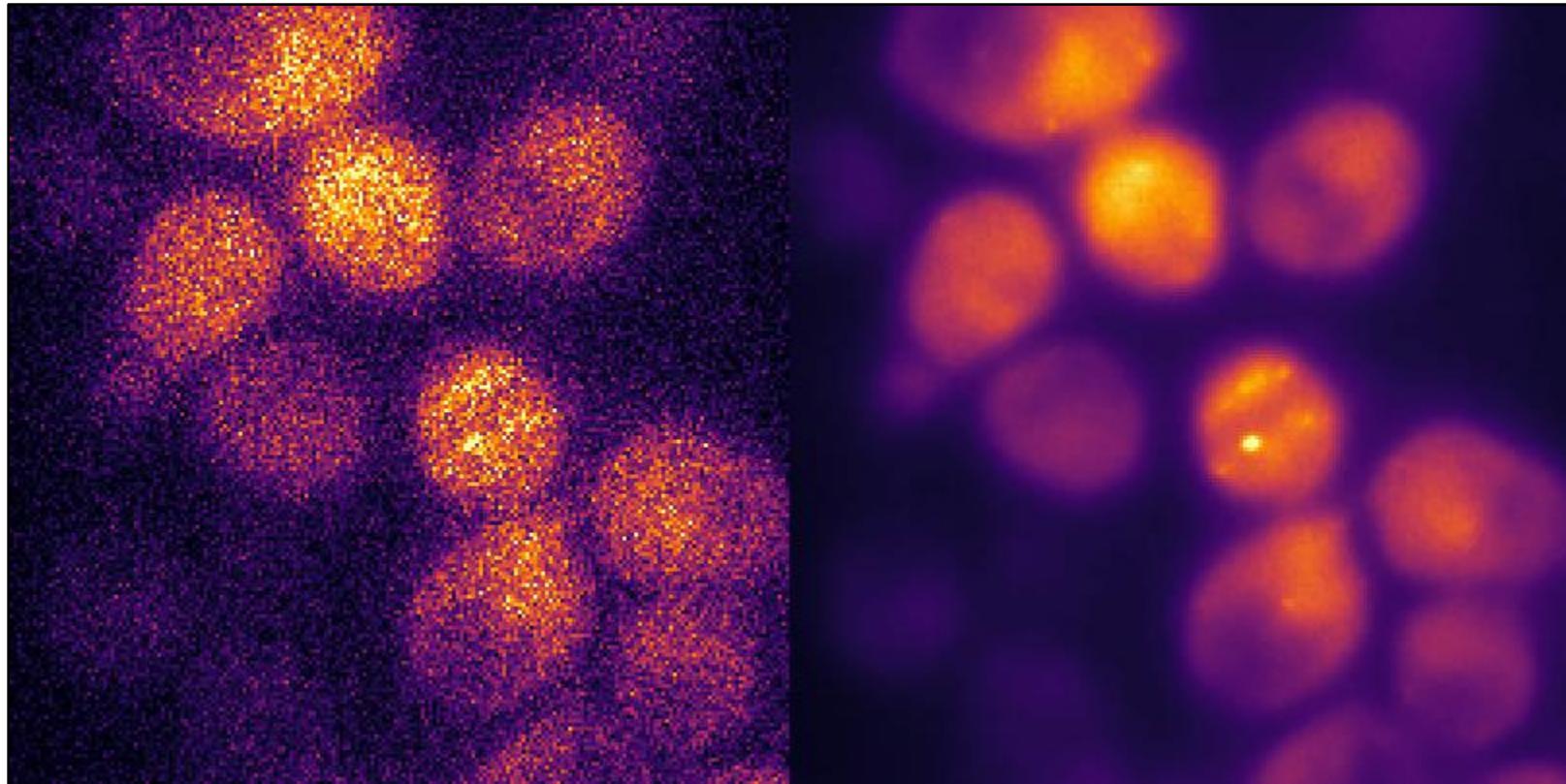


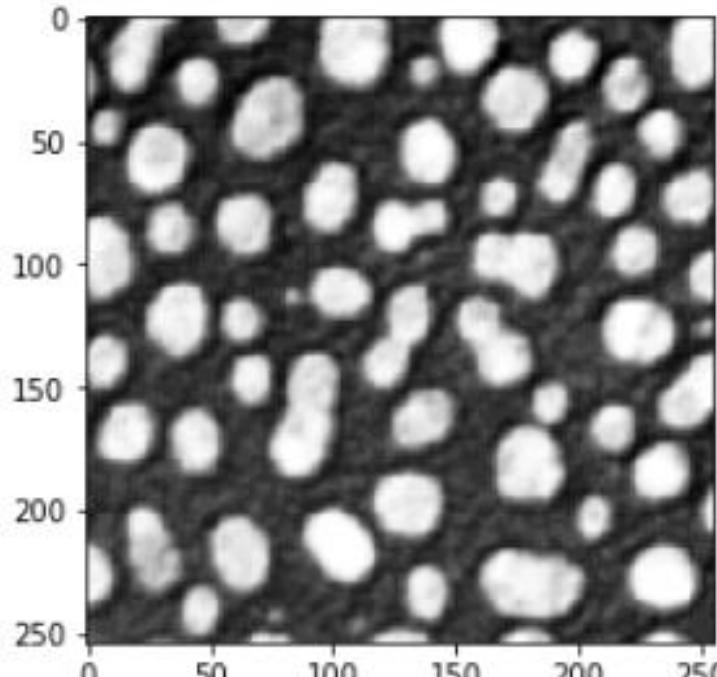
Image denoising

- Noise2Void

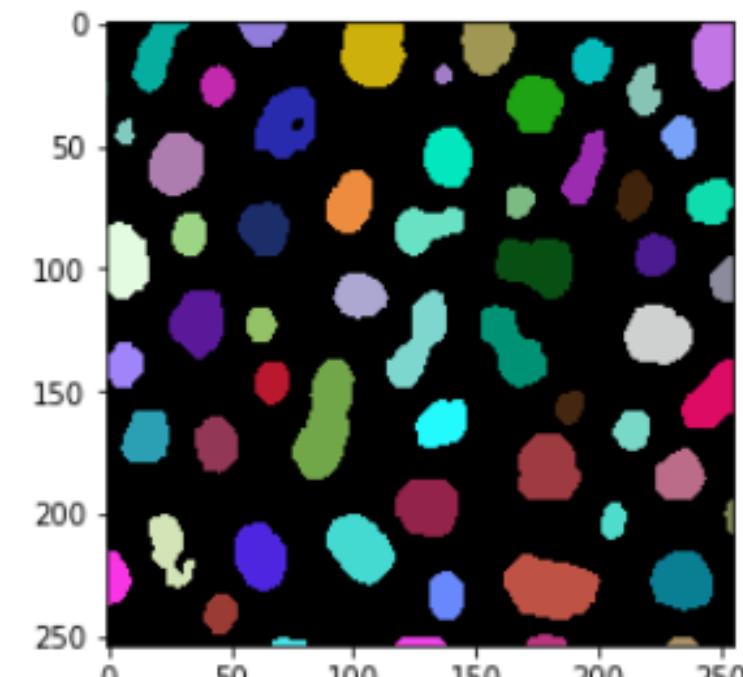
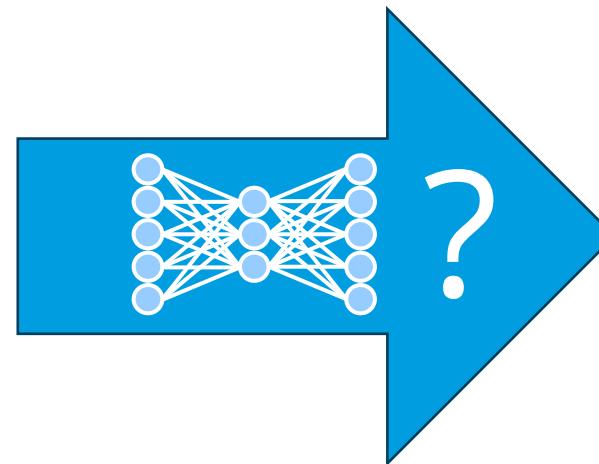


DL-based image segmentation

Quiz: Is it possible to train a neural network that performs this operation *directly*? Why yes? Why no?



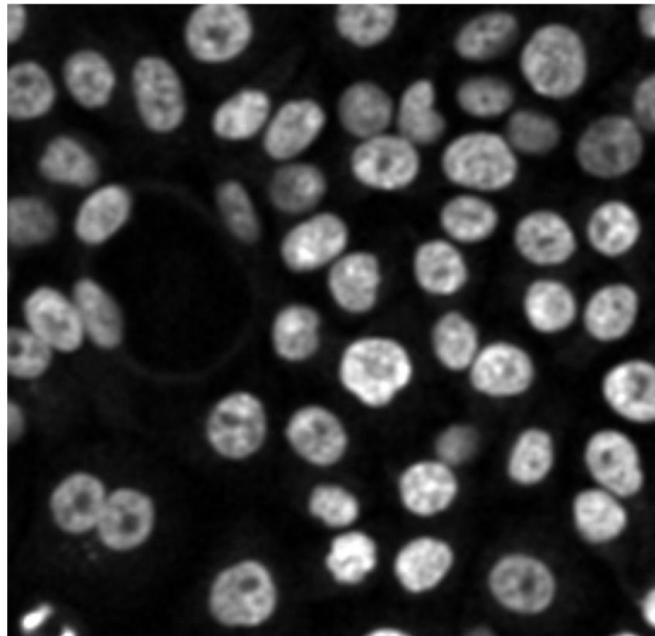
Intensity image



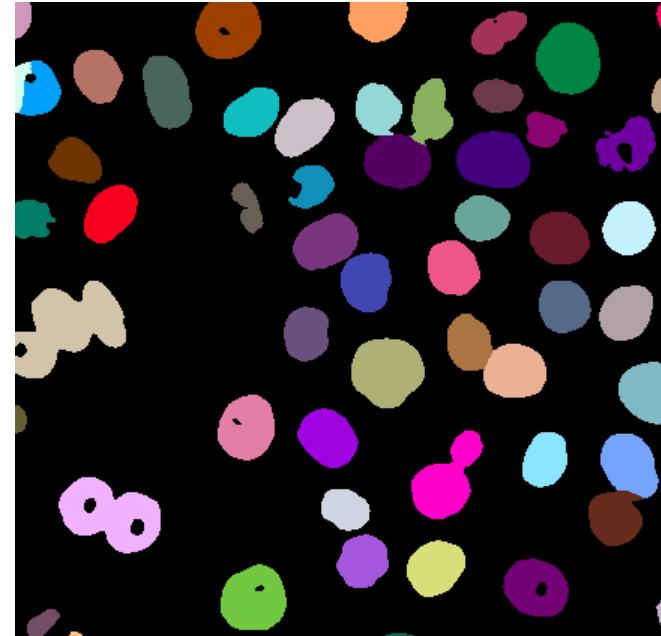
Label image

StarDist: Nuclei segmentation

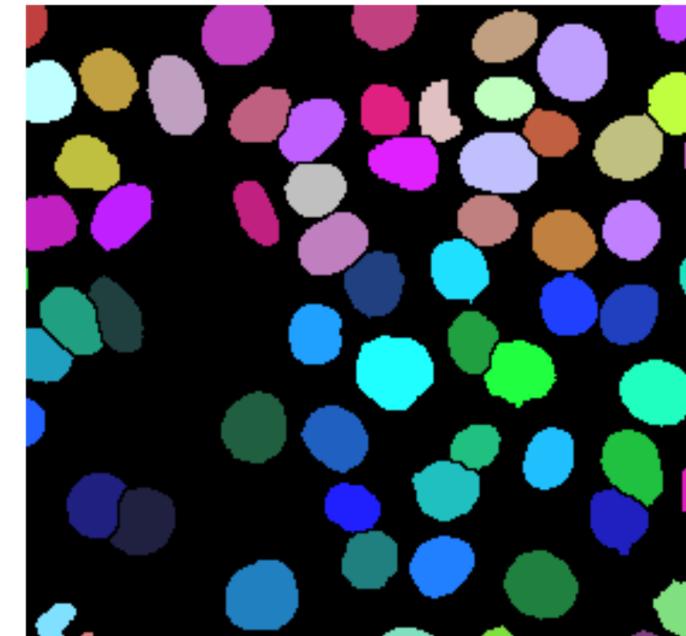
Advanced algorithms are necessary when nuclei become too dense.



Input image



Voronoi-Otsu-Labeling



StarDist (DL-based)

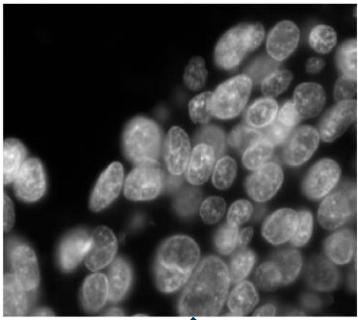
StarDist

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

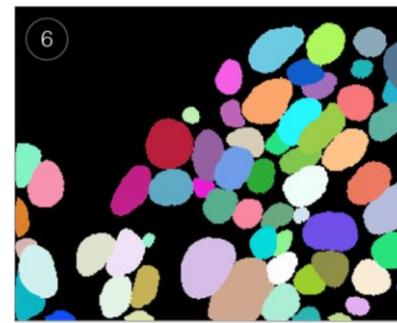
Class. Image proc.

Deep learning

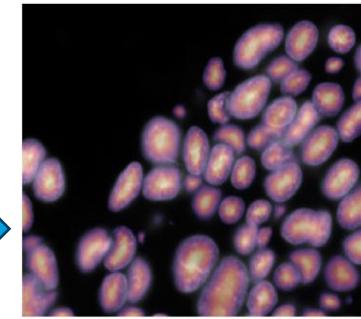
Input



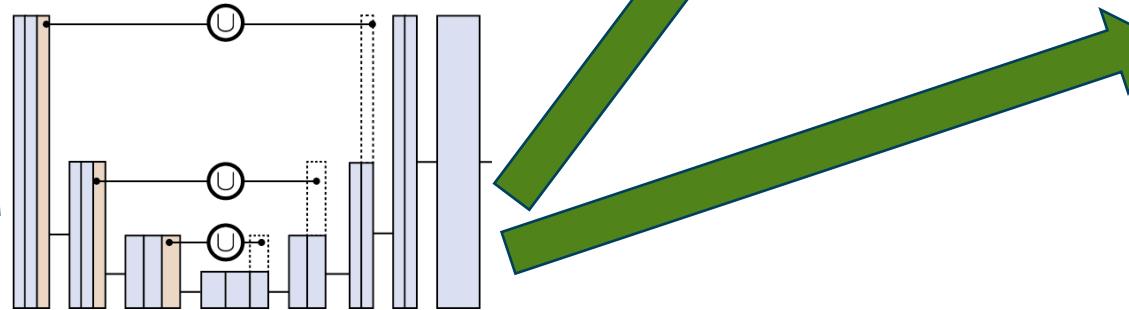
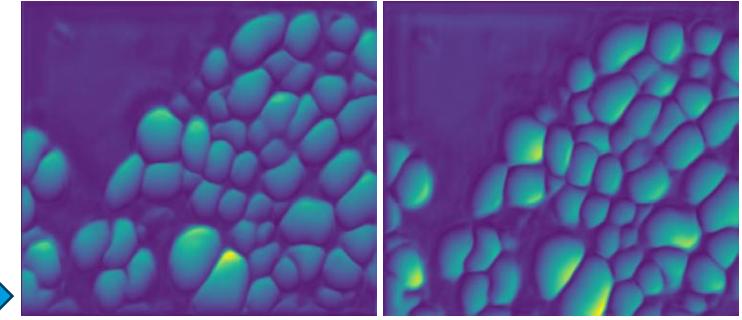
Ground truth



Object probabilities



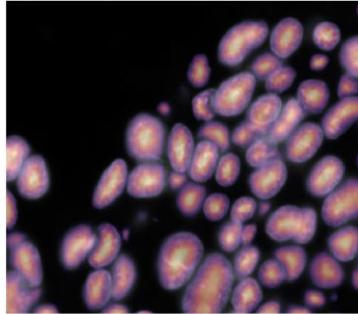
Directional distance maps (32x)



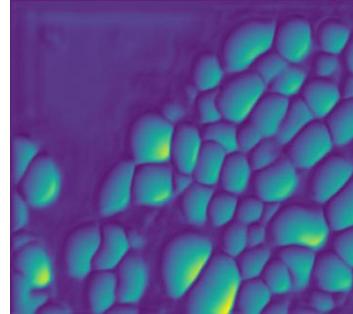
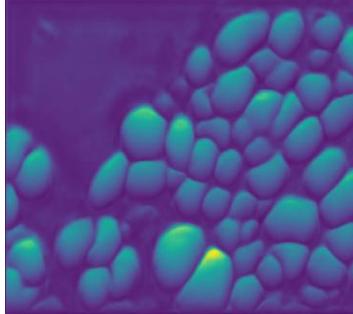
StarDist

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

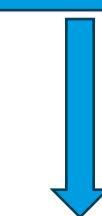
Object probabilities



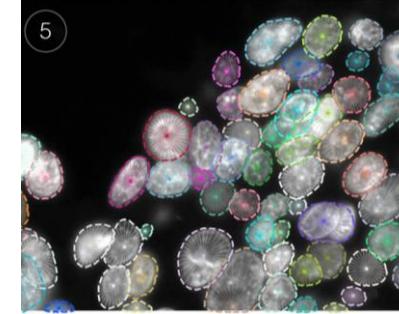
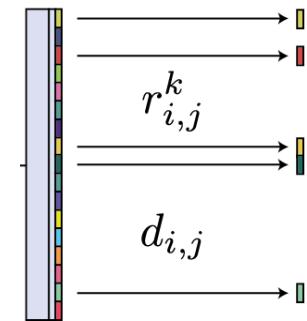
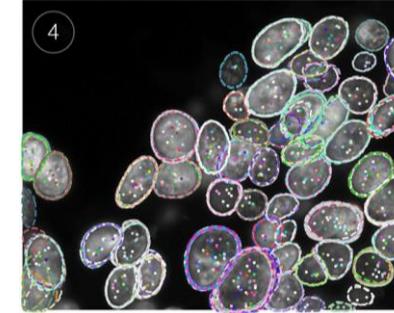
Directional distance maps



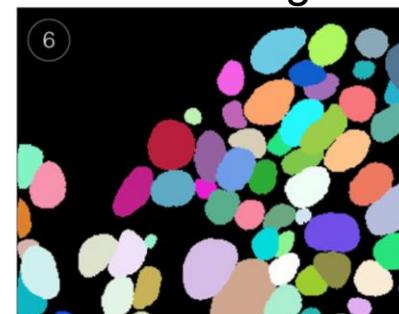
Local
maxima



Polygon candidates

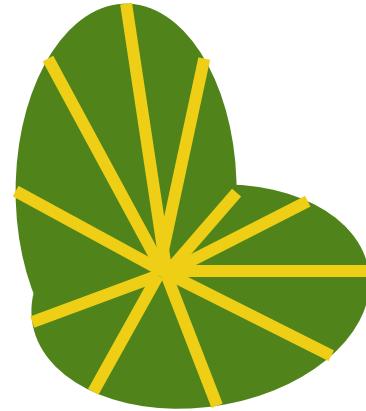
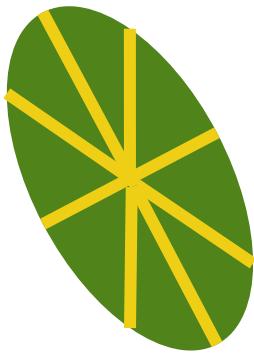
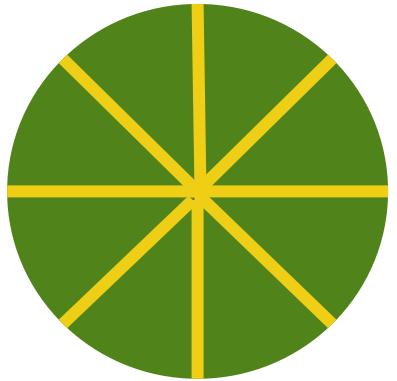


Final segmentation:
polygons



StarDist: Limitations

Star-convex shapes!



Good for nuclei, bad for cells.

Quiz

Which shape features may be compromised if we use StarDist for segmentation? Why?

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]

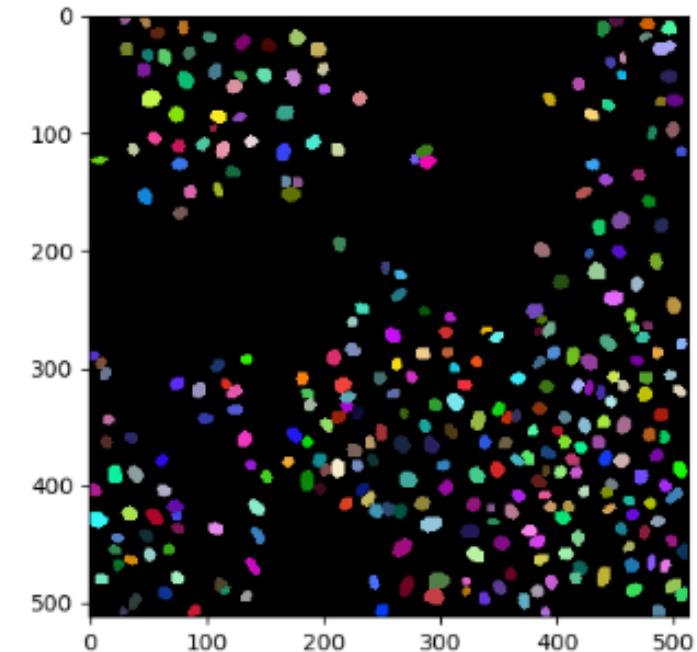
```
[4]: axis_norm = (0,1)  
image = normalize(image, 1,99.8, axis=axis_norm)
```

Quiz: Many DL-based segmentation algorithms perform a normalization before processing the image. Why?

Apply model

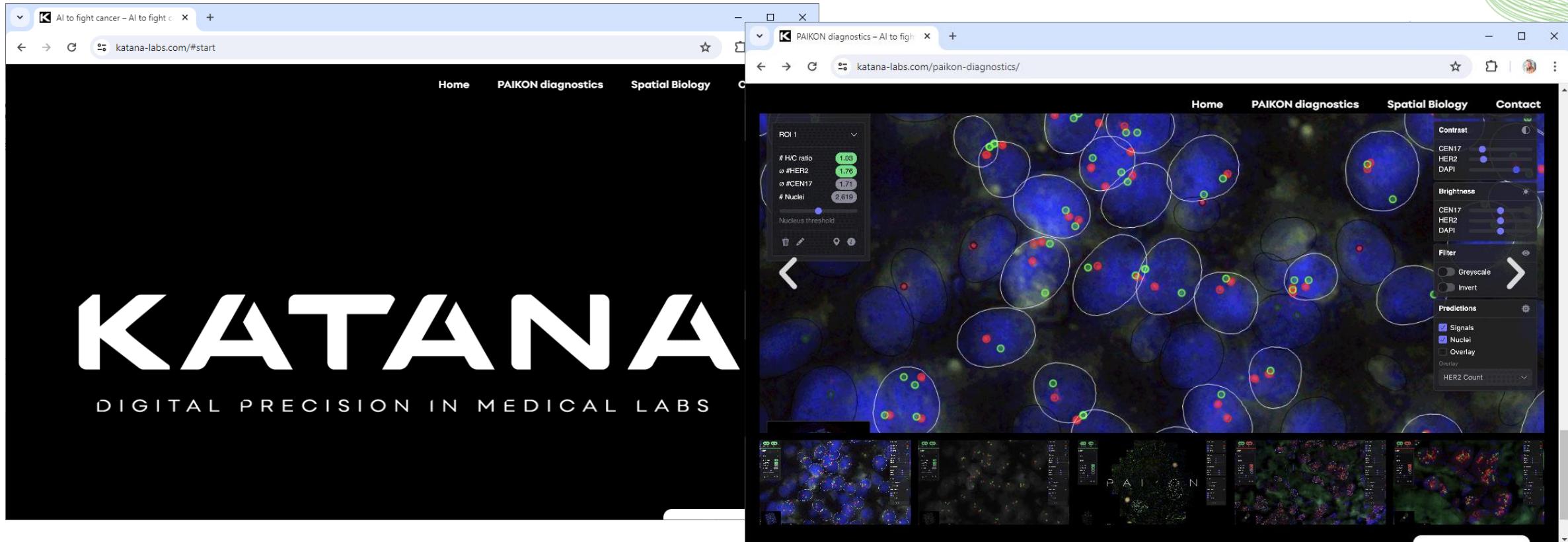
```
[5]: labels, details = model.predict_instances(image)  
  
stackview.insight(labels)
```

[5]:



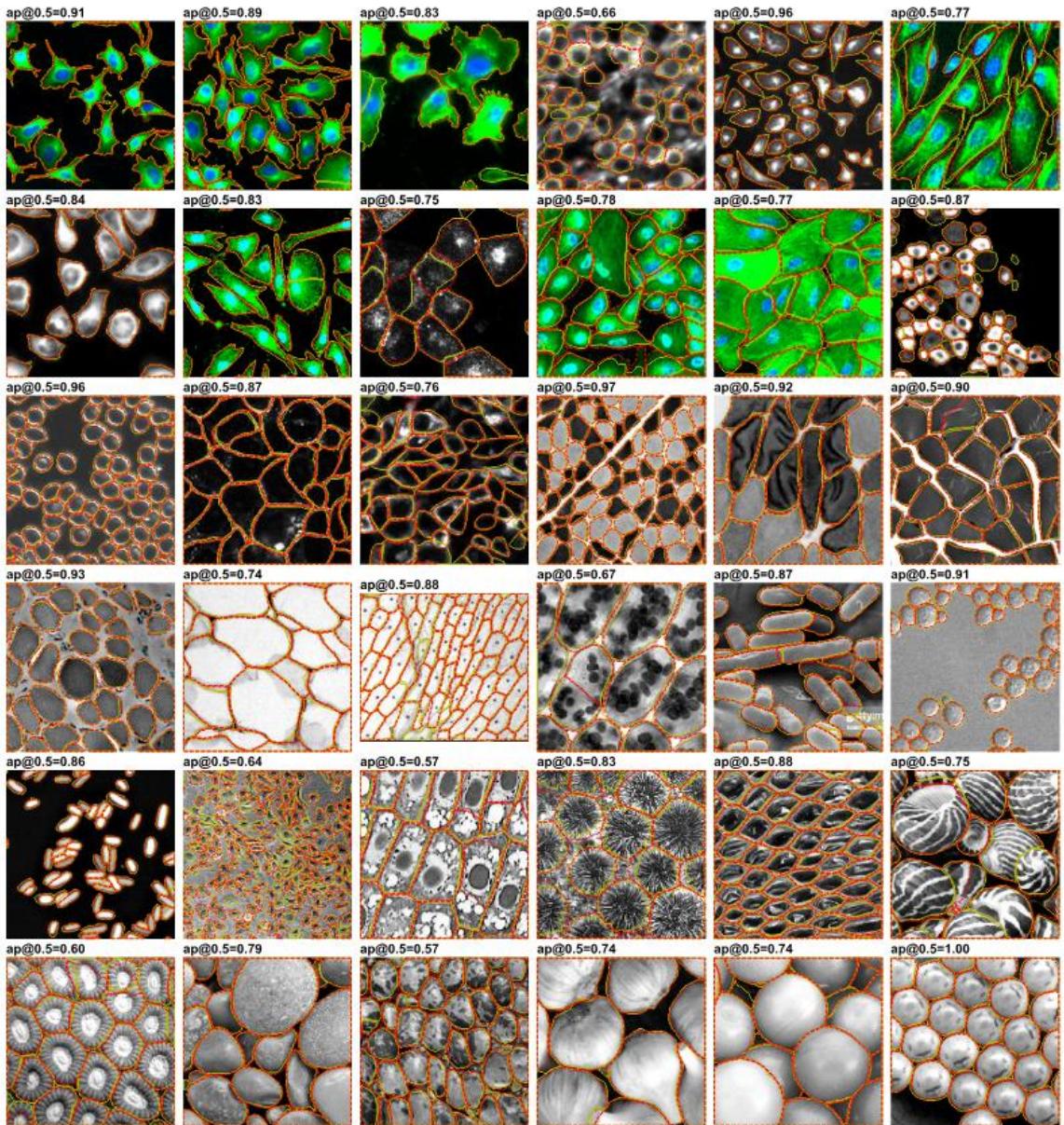
StarDist: Business model

Serving clinical scientists using software based on open-source code



CellPose

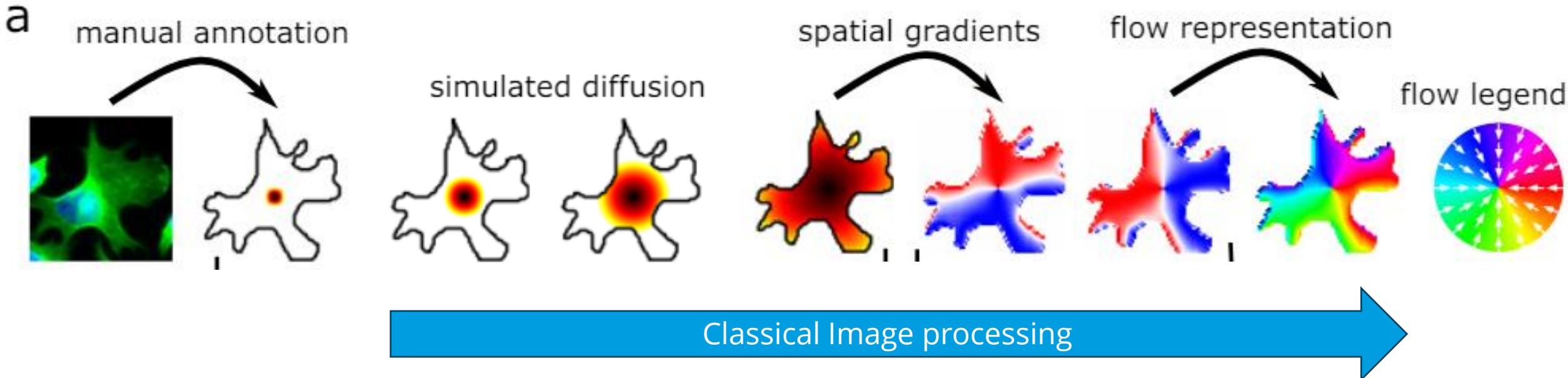
- Processing more diverse datasets and object shapes



CellPose: Training

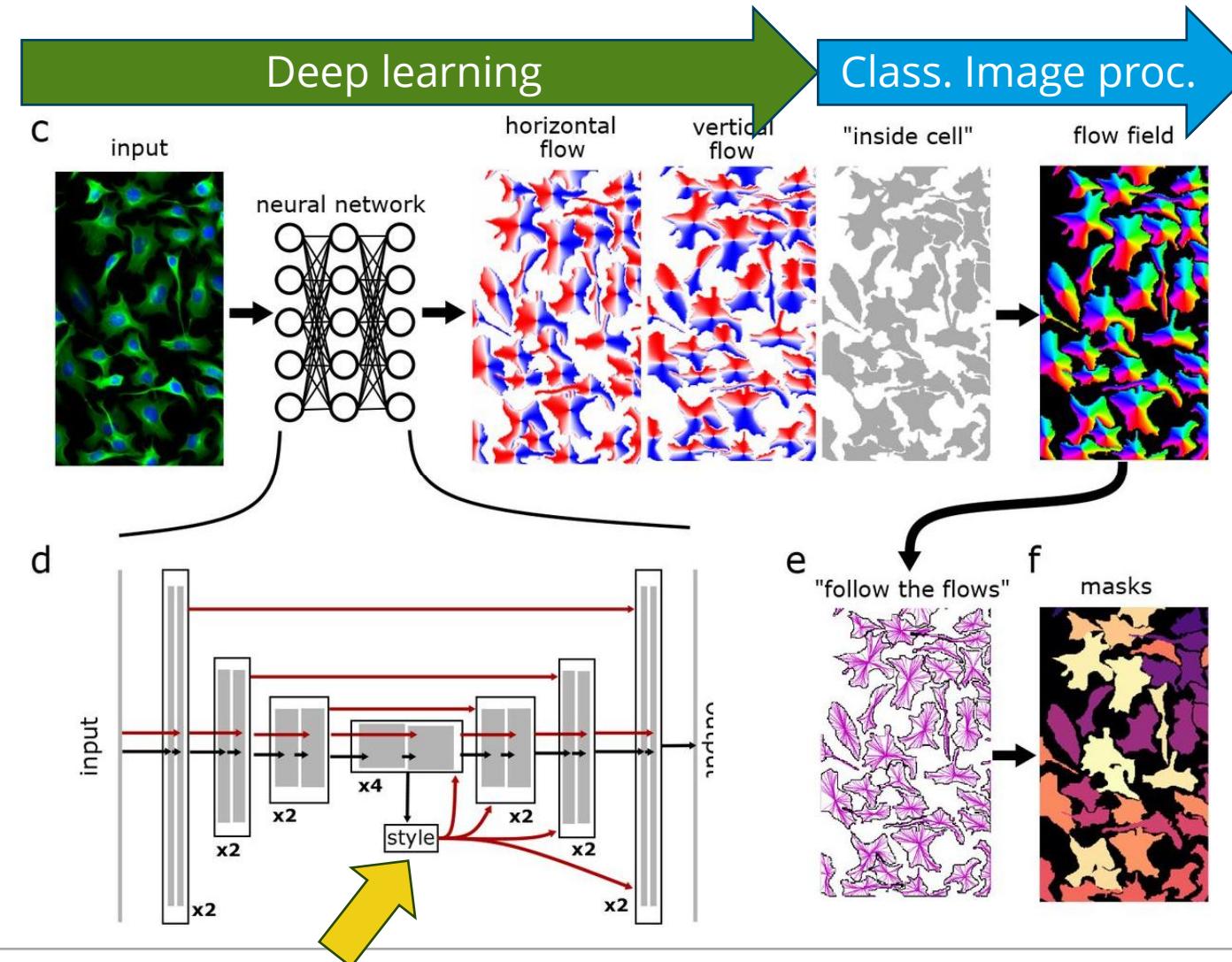
Compute „flow fields“ from images

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



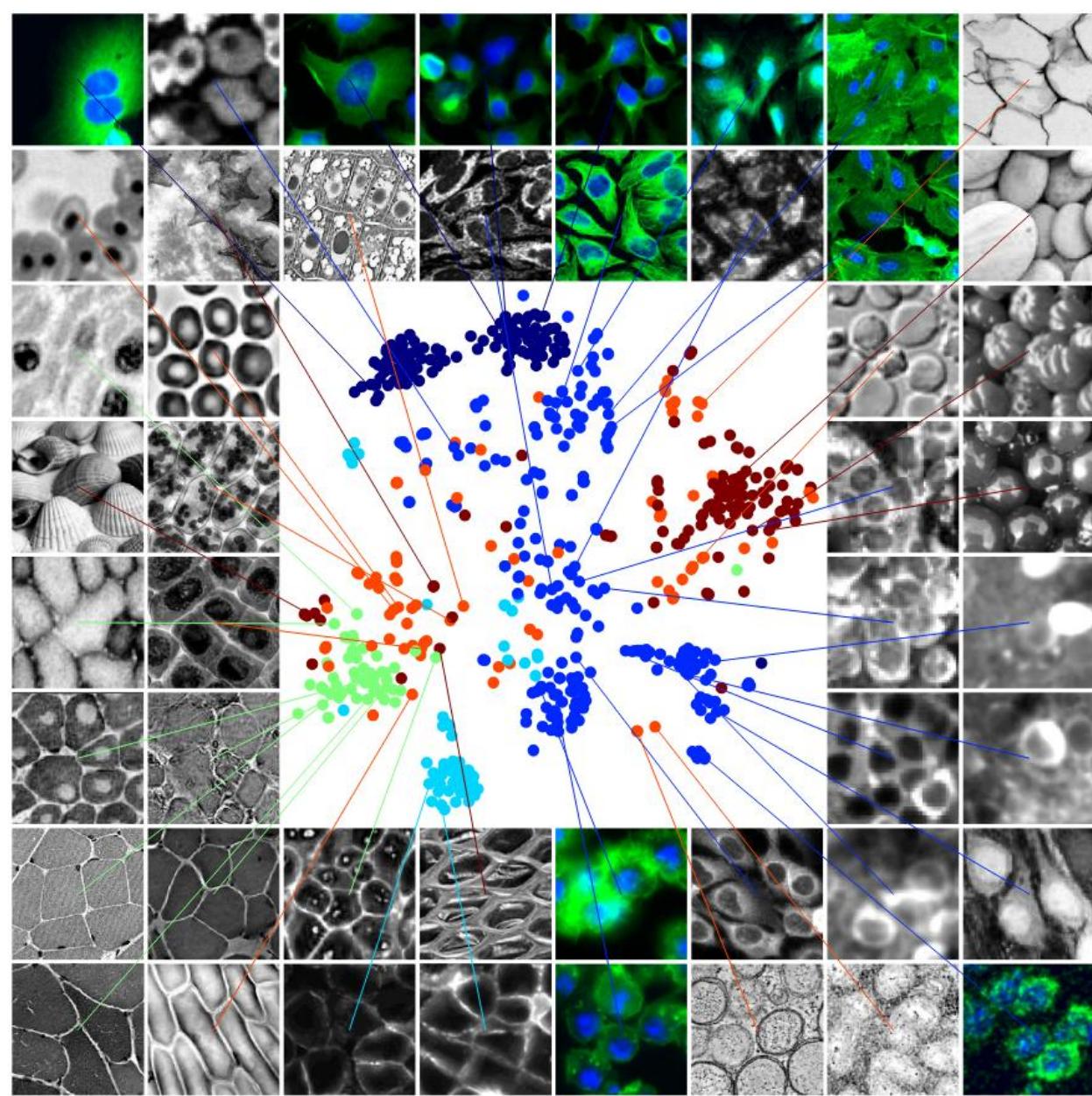
CellPose: Prediction

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

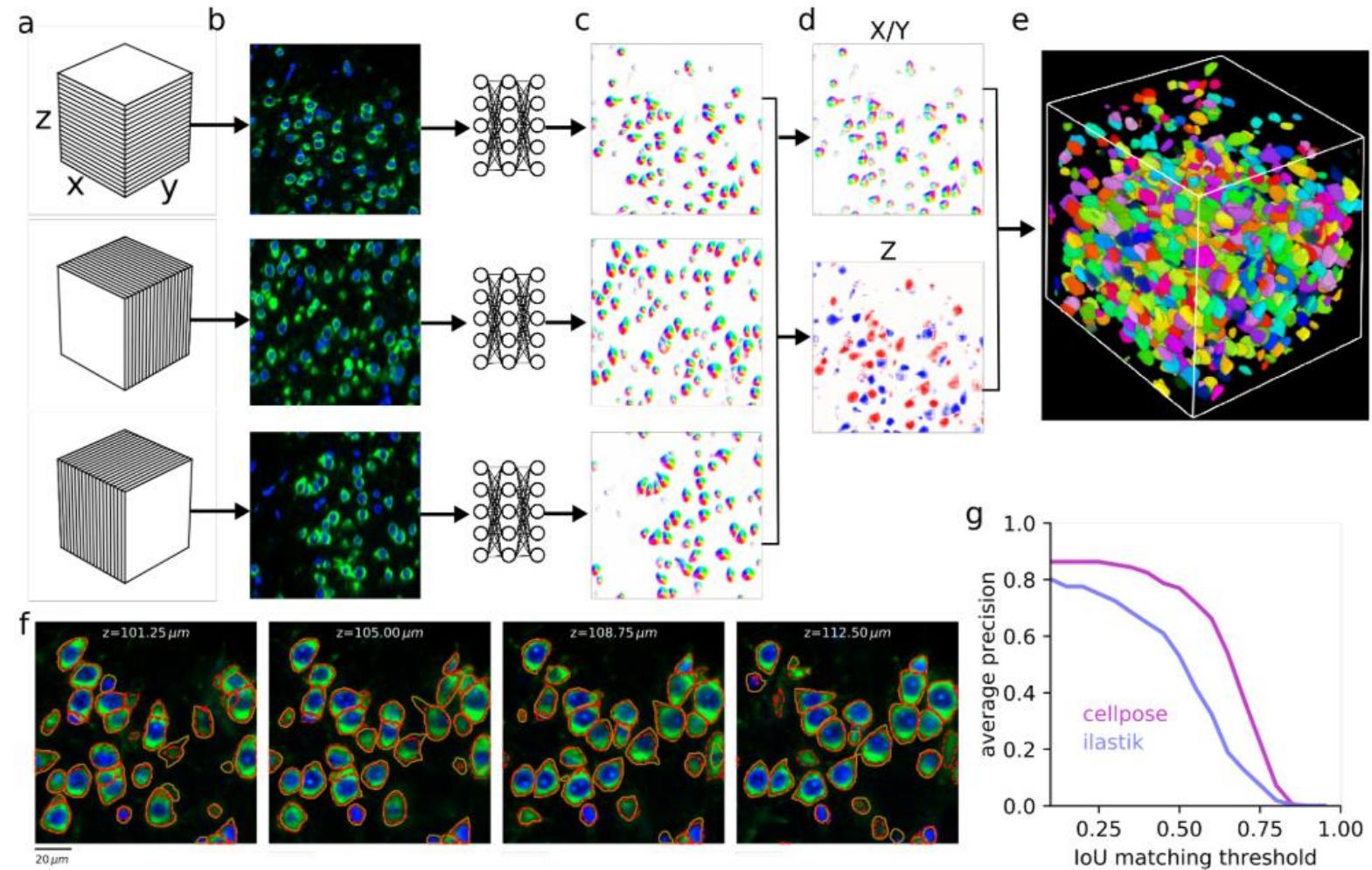


CellPose

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



Cellpose 3D



Cellpose in Python

Initialize a pretrained model

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

List available models

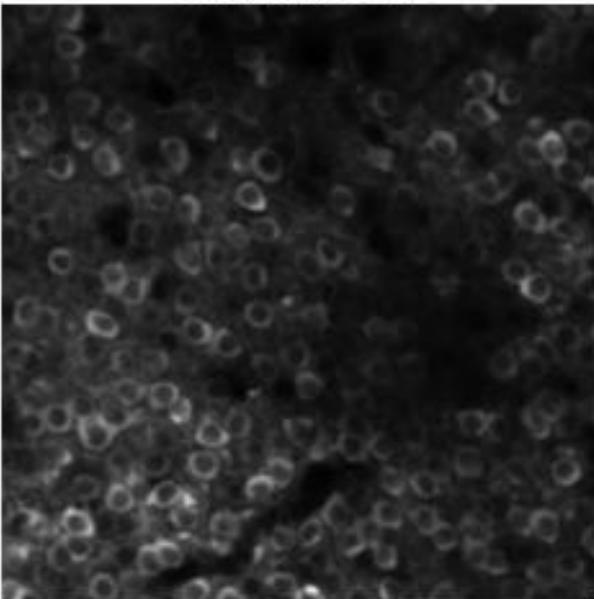
```
[8]: cellpose.models.MODEL_NAMES
```

```
[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']
```

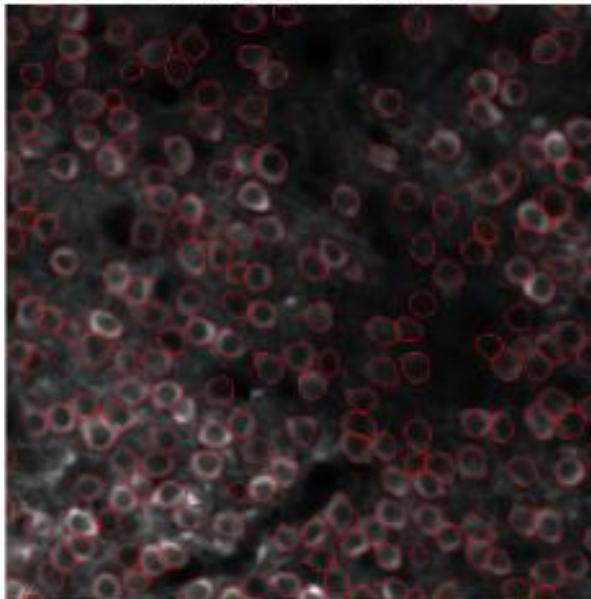
Cellpose in Python

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

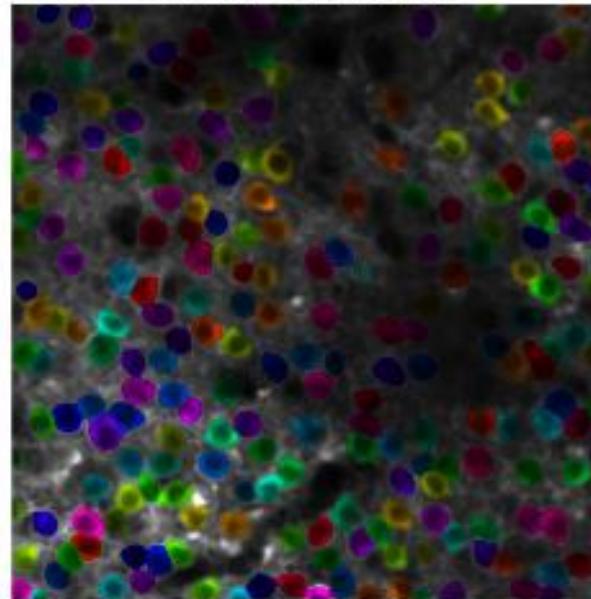
original image



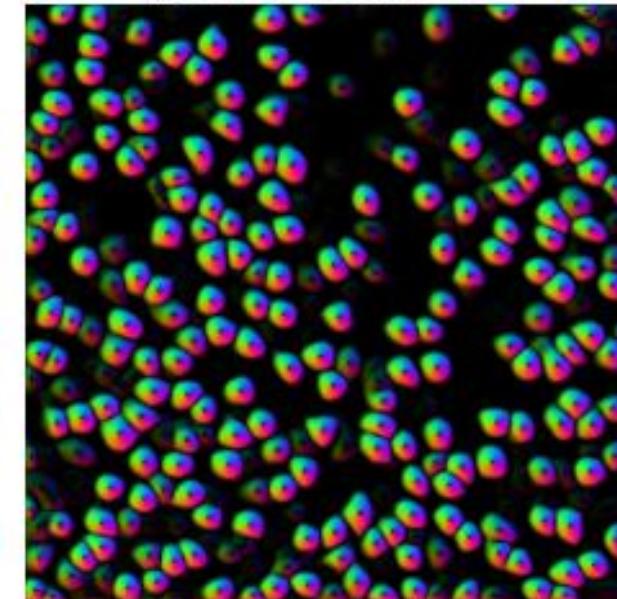
predicted outlines



predicted masks

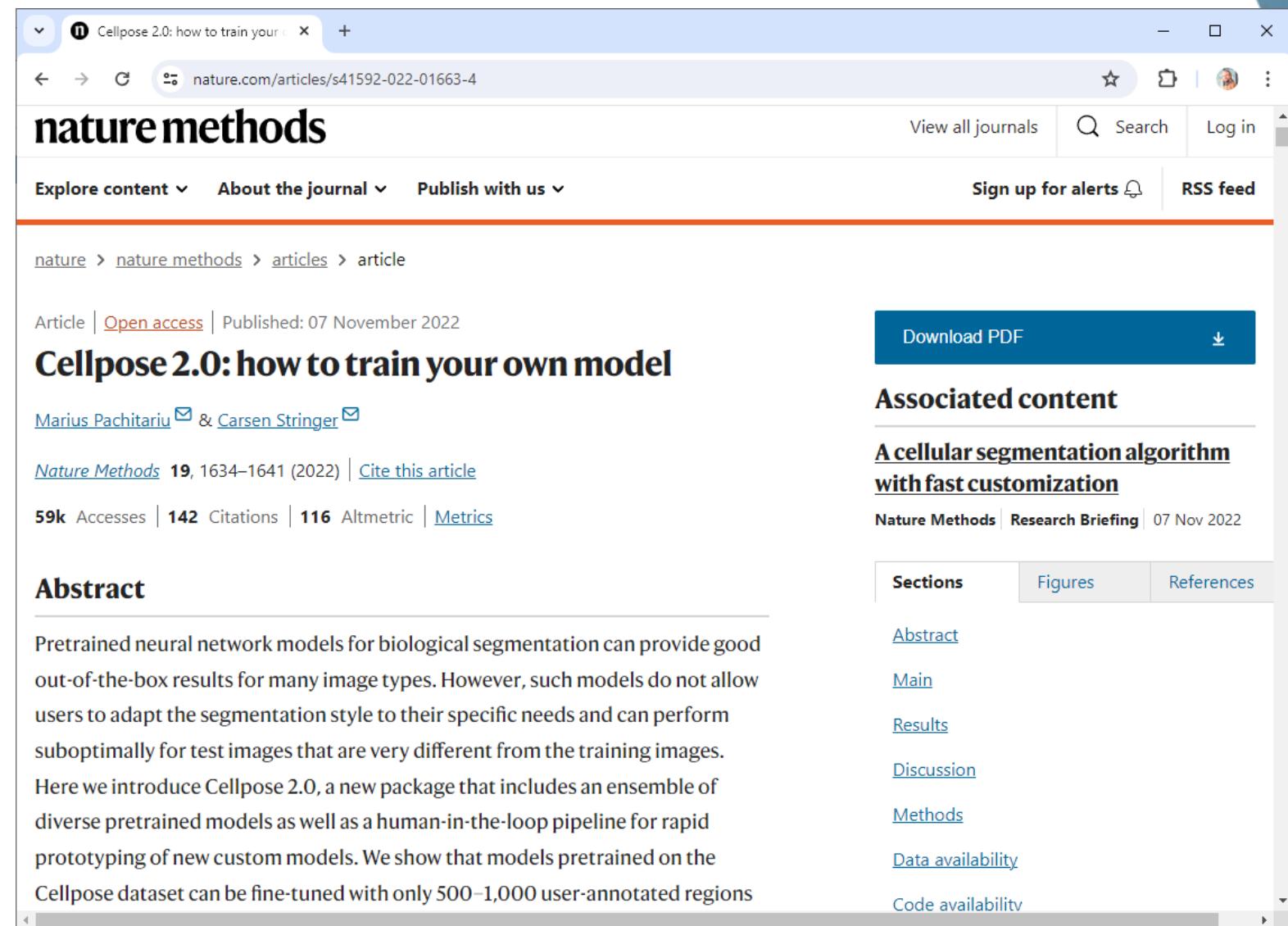


predicted cell pose



Cellpose 2

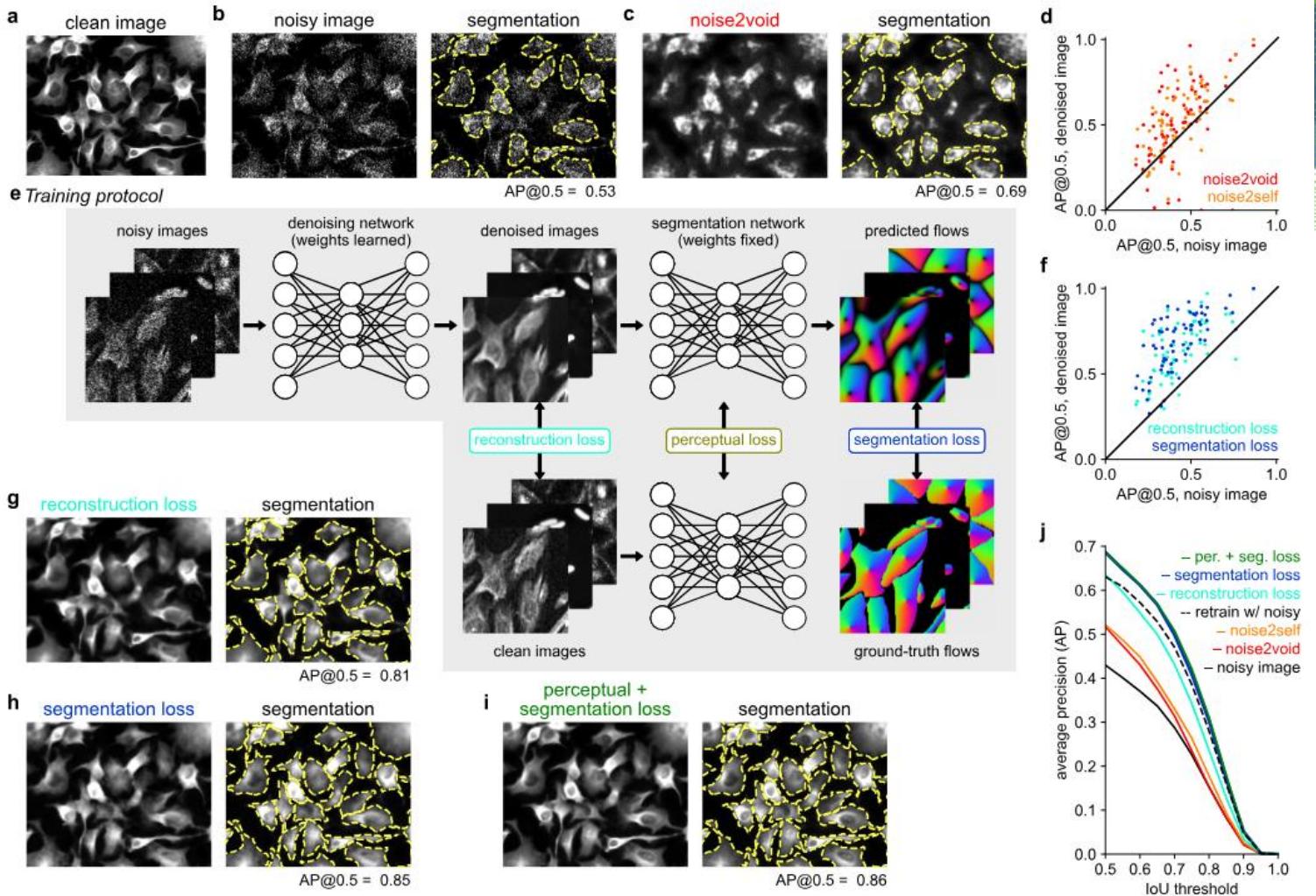
Train your own model



The screenshot shows a web browser displaying the article "Cellpose 2.0: how to train your own model" from the journal *nature methods*. The page includes the header with navigation links like "Explore content", "About the journal", "Publish with us", "Sign up for alerts", and "RSS feed". Below the header, the breadcrumb navigation shows "nature > nature methods > articles > article". The main title is "Cellpose 2.0: how to train your own model" by Marius Pachitariu and Carsen Stringer, published in *Nature Methods* 19, 1634–1641 (2022). The article has 59k accesses, 142 citations, and 116 Altmetric. To the right, there is a "Download PDF" button and a sidebar titled "Associated content" featuring a brief about a cellular segmentation algorithm with fast customization. The abstract section discusses the limitations of pretrained neural network models and introduces Cellpose 2.0, which includes an ensemble of diverse pretrained models and a human-in-the-loop pipeline for rapid prototyping of new custom models. The abstract also notes 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



A short excursion...

Published 1 year ago:

Claim to
outperform current
state of the art.

“The T1 algorithms achieved a median F1 score of 89.7% (IQR 36.7–82.4%), surpassing the KIT-GE, Cellpose-pretrain, Cellpose-scratch, Omnipose-pretrain and Omnipose-scratch by 49.9%, 24.4%, 35.4%, 58.9% and 48.7%, respectively.”

The screenshot shows the 'nature methods' journal website. The article title is 'The multimodality cell segmentation challenge: toward universal solutions'. It lists numerous authors including Jun Ma, Ronald Xie, Shamini Ayyadhury, Cheng Ge, Anubha Gupta, Ritu Gupta, Song Gu, Yao Zhang, Gihun Lee, Joonkee Kim, Wei Lou, Haofeng Li, Eric Upschulte, Timo Dickscheid, José Guilherme de Almeida, Yixin Wang, Lin Han, Xin Yang, Marco Labagnara, Vojislav Gligorovski, Maxime Scheder, Sahand Jamal Rahi, Carly Kempster, Alice Pollitt, Bo Wang. The article was published on 26 March 2024. It has 7645 accesses, 19 Altmetric, and Metrics. There is a link to 'Download PDF' and a section for 'Associated content' titled 'Creating a universal cell segmentation algorithm'.

A short excursion...

“However my initial testing of the winning entry [mediar 8](#) reveals errors (which are typical without tuning) when applied to recent data posted [here 6](#) for which cellpose produced a (subjectively) good result.”

►  Community Partners

Thoughts on the latest SOTA Segmentation algorithms

Blog Posts segmentation, cellpose, deep-learning, paper-review

 Brian Northan  bnorthan Community Forum Team member 2 13d

Curious what people think about the latest paper on DL Segmentation that can be found [here](#) 45

It's behind a paywall but some of the figures are available publicly. In particular I am trying to understand this figure as it does not really reconcile with my practical experience. The F1 scores for cellpose seem really low.

A short excursion...

Published 11 days later:
(by the CellPose authors)

“Impairment One. Cellpose was trained and tested on grayscale images while the other methods were trained on the full multi-channel data...”

The image shows a screenshot of a bioRxiv preprint page. At the top, the bioRxiv logo is displayed with the tagline "THE PREPRINT SERVER FOR BIOLOGY". Below the logo, there is a red button labeled "View current version of this article" and a blue button labeled "Follow this preprint". The main title of the preprint is "Transformers do not outperform Cellpose". Below the title, the authors are listed as "Carsen Stringer" and "Marius Pachitariu". The DOI is provided as "doi: <https://doi.org/10.1101/2024.04.06.587952>". A note states, "This article is a preprint and has not been certified by peer review [what does this mean?].". Below the title, there are navigation links for "Abstract", "Full Text", "Info/History" (which is highlighted in a box), and "Metrics". At the bottom of the page, there is an "ARTICLE INFORMATION" section with the DOI again and a "History" entry dated "April 7, 2024".

A short excursion...

Published 11 days later:
(by the CellPose authors)

“Impairment Two. Cellpose was trained with images rescaled so that cells have an average diameter of 30 pixels (which is the default), but tested with unresized images in which cells have diameters up to 400 pixels.”

The image shows a screenshot of a bioRxiv preprint page. At the top, the bioRxiv logo is displayed with the tagline "THE PREPRINT SERVER FOR BIOLOGY". Below the logo, there is a red button labeled "View current version of this article" and a blue button labeled "Follow this preprint". The main title of the preprint is "Transformers do not outperform Cellpose". Below the title, the authors are listed as "Carsen Stringer" and "Marius Pachitariu". The DOI is provided as "doi: <https://doi.org/10.1101/2024.04.06.587952>". A note states, "This article is a preprint and has not been certified by peer review [what does this mean?].". Below the title, there are tabs for "Abstract", "Full Text", "Info/History" (which is highlighted in a white box), and "Metrics". At the bottom of the page, there is an "ARTICLE INFORMATION" section with the DOI again and the date "History April 7, 2024".

A short excursion...

Published 11 days later:
(by the CellPose authors)

“Impairment Three. The training set for Cellpose included only the challenge images, while the top performing methods used additional datasets.”

The image shows a screenshot of a bioRxiv preprint page. At the top, the bioRxiv logo is displayed with the tagline "THE PREPRINT SERVER FOR BIOLOGY". Below the logo, there is a red button labeled "View current version of this article" and a blue button labeled "Follow this preprint". The main title of the preprint is "Transformers do not outperform Cellpose". Below the title, the authors are listed as "Carsen Stringer" and "Marius Pachitariu". The DOI is provided as "doi: <https://doi.org/10.1101/2024.04.06.587952>". A note states, "This article is a preprint and has not been certified by peer review [what does this mean?].". Below the title, there are navigation links for "Abstract", "Full Text", "Info/History" (which is highlighted in a box), and "Metrics". At the bottom of the page, there is an "ARTICLE INFORMATION" section with the DOI again and a "History" entry dated "April 7, 2024".

A short excursion...

Published 11 days later:
(by the CellPose authors)

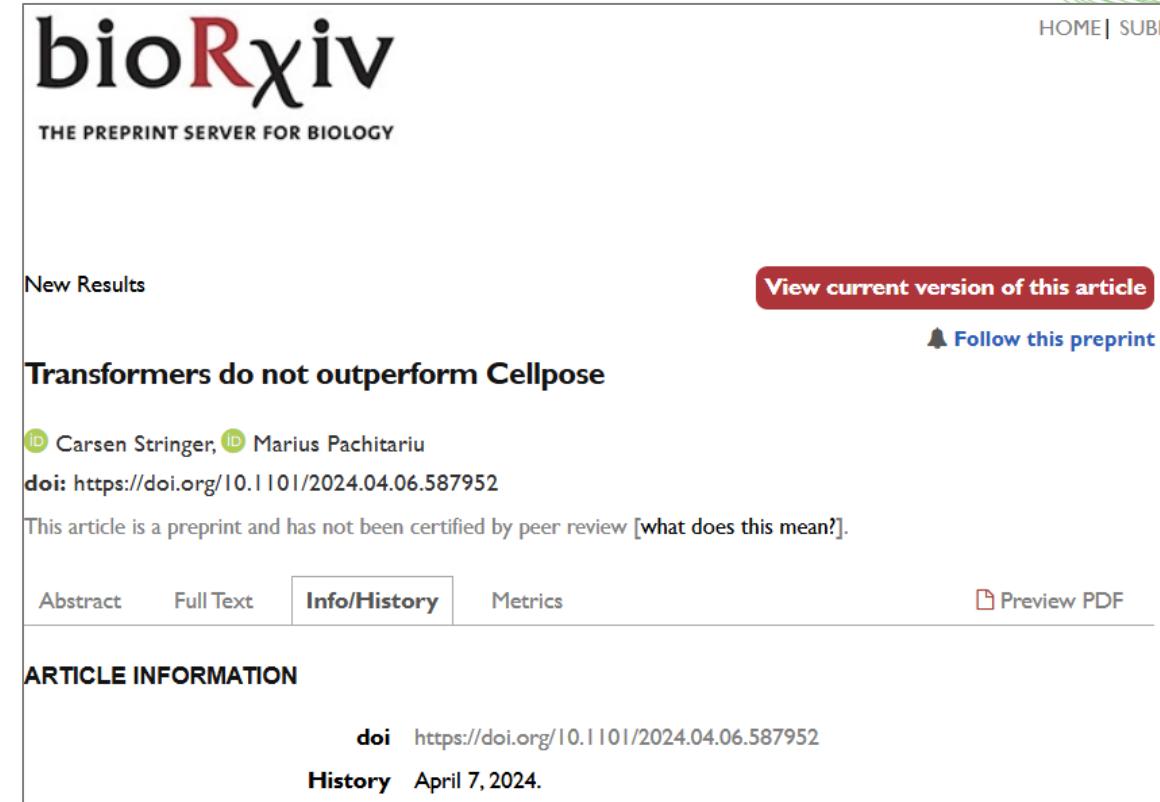
“Impairment Four. Cellpose was run without testtime augmentations (TTA). The Cellpose framework makes it easy to augment images at test time and we described this in the original study [3].”

The image shows a screenshot of a bioRxiv preprint page. At the top, the bioRxiv logo is displayed with the tagline "THE PREPRINT SERVER FOR BIOLOGY". Below the logo, there is a red button labeled "View current version of this article" and a blue button labeled "Follow this preprint". The main title of the preprint is "Transformers do not outperform Cellpose". Below the title, the authors are listed as "Carsen Stringer" and "Marius Pachitariu". The DOI is provided as "doi: <https://doi.org/10.1101/2024.04.06.587952>". A note states, "This article is a preprint and has not been certified by peer review [what does this mean?].". Below the title, there are navigation links for "Abstract", "Full Text", "Info/History" (which is highlighted in a box), and "Metrics". At the bottom of the page, there is an "ARTICLE INFORMATION" section with the DOI again and the date "History April 7, 2024".

A short excursion...

Published 11 days later:
(by the CellPose authors)

“In fact, Mediar directly copied our codebase for implementing the Cellpose framework without modification [...] Ma et al [1] does not dwell on this similarity between Mediar and Cellpose. Instead, they emphasize the novelty of using a transformer architecture”



The image shows a screenshot of a bioRxiv preprint page. At the top, the bioRxiv logo is displayed with the tagline "THE PREPRINT SERVER FOR BIOLOGY". In the top right corner, there are links for "HOME" and "SUBMIT". Below the logo, a red button says "View current version of this article". To the right of that, a blue button says "Follow this preprint". The main title of the preprint is "Transformers do not outperform Cellpose". Below the title, it lists two authors: "Carsen Stringer" and "Marius Pachitariu". The DOI is provided as "doi: <https://doi.org/10.1101/2024.04.06.587952>". A note states, "This article is a preprint and has not been certified by peer review [what does this mean?].". Below the title, there are tabs for "Abstract", "Full Text", "Info/History" (which is highlighted in a white box), and "Metrics". To the right of these tabs, there is a link "Preview PDF". Under the "ARTICLE INFORMATION" section, the DOI is again listed as "doi: <https://doi.org/10.1101/2024.04.06.587952>". Below that, the "History" is noted as "April 7, 2024".

A short excursion...

Professional discussion about Copyright:

The screenshot shows a GitHub issue page for the repository 'Lee-Gihun/MEDIAR'. The title of the issue is 'Please retain the Cellpose copyright as required by the BSD-3 license #17'. The status is 'Closed' and it was opened by 'marius10p' last week with 7 comments. The comment from 'marius10p' states: 'The Cellpose license (BSD-3) is very permissive. However, it requires you to retain the copyright notice [here](#) when you copy portions of the code. All of the contents of [this library](#) in your repository are copied over from the Cellpose dynamics library [here](#) (primarily) or from the Cellpose utils and metrics libraries. The copyright notice must be maintained and distributed with every copy of your codebase (including in the napari plugin and in the docker image). In addition, we ask that you include this copyright statement at the top of your file:
Copyright © 2022 Howard Hughes Medical Institute, Authored by Carsen Stringer and Marius Pachitariu.' The right sidebar shows assignees 'Lee-Gihun' and 'joonkeekim', labels 'documentation', projects 'None yet', milestones 'No milestone', and development details.

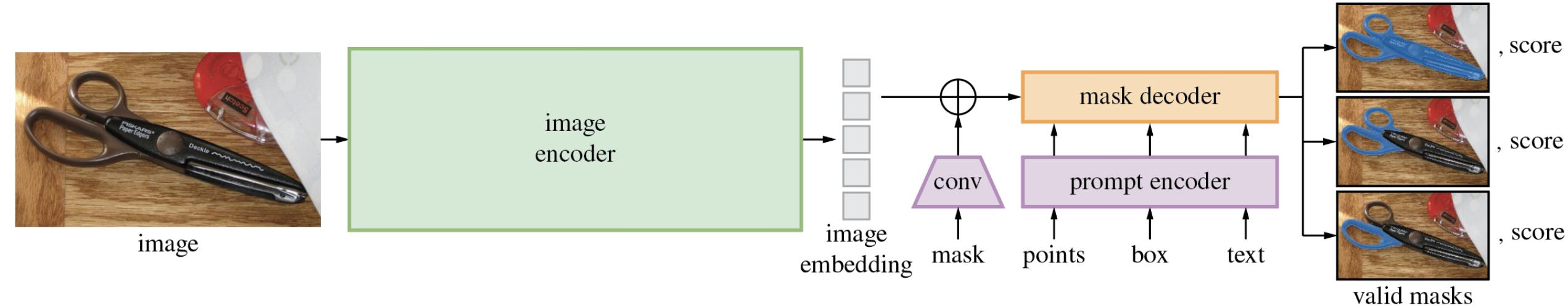
Segment Anything Model

New approach to DL-based image segmentation involving prompts



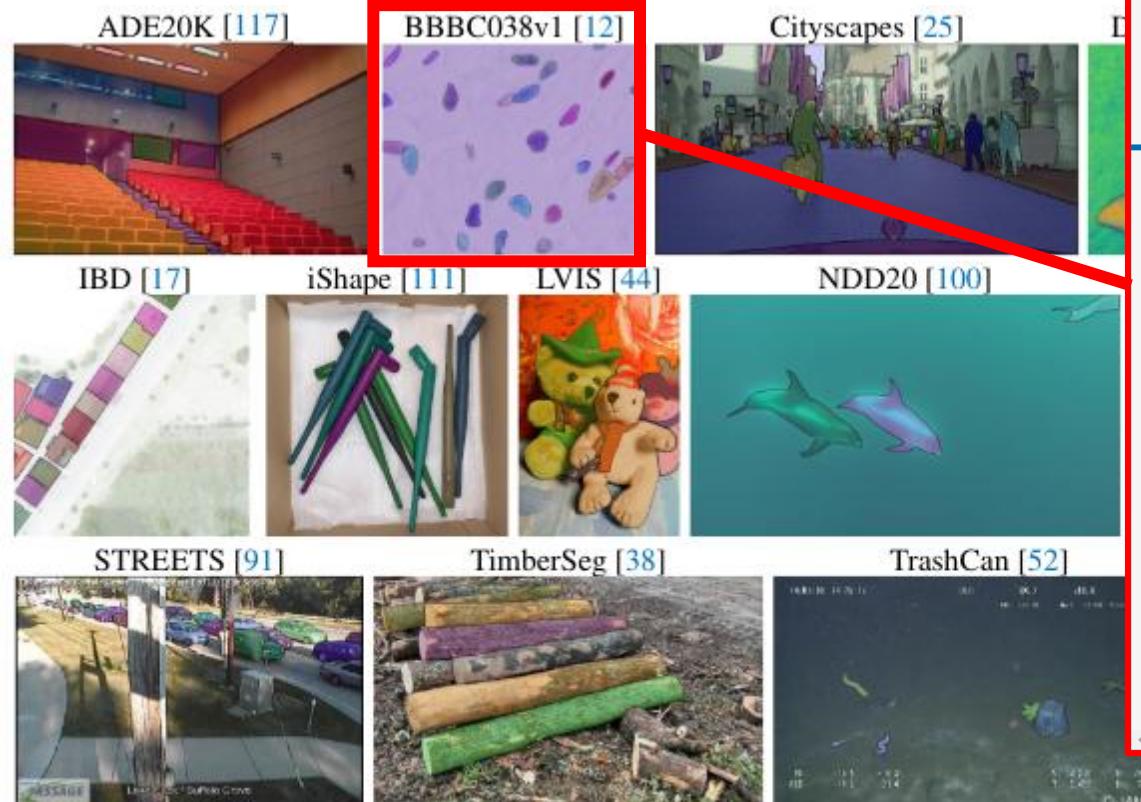
Segment Anything Model

New approach to DL-based image segmentation involving prompts



Segment Anything Model

Trained on mostly natural images



Screenshot of the Kaggle 2018 Data Science Bowl page for BBBC038. The page shows the Broad Bioimage Benchmark Collection logo and navigation links for Introduction, Image Sets, Benchmarking, and Contribute. A legend titled "LEGEND: KINDS OF GROUND TRUTH" lists six categories: COUNTS (C), FOREGROUND / BACKGROUND (F), OUTLINES OF OBJECTS (O), BIOLOGICAL LABELS (B), LOCATION (L), and BOUNDING BOXES (U). Below the legend are three example images showing biological tissue segmentation.

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

Segment Anything Model

Downsides:

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

Segment Anything Model for Microscopy

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

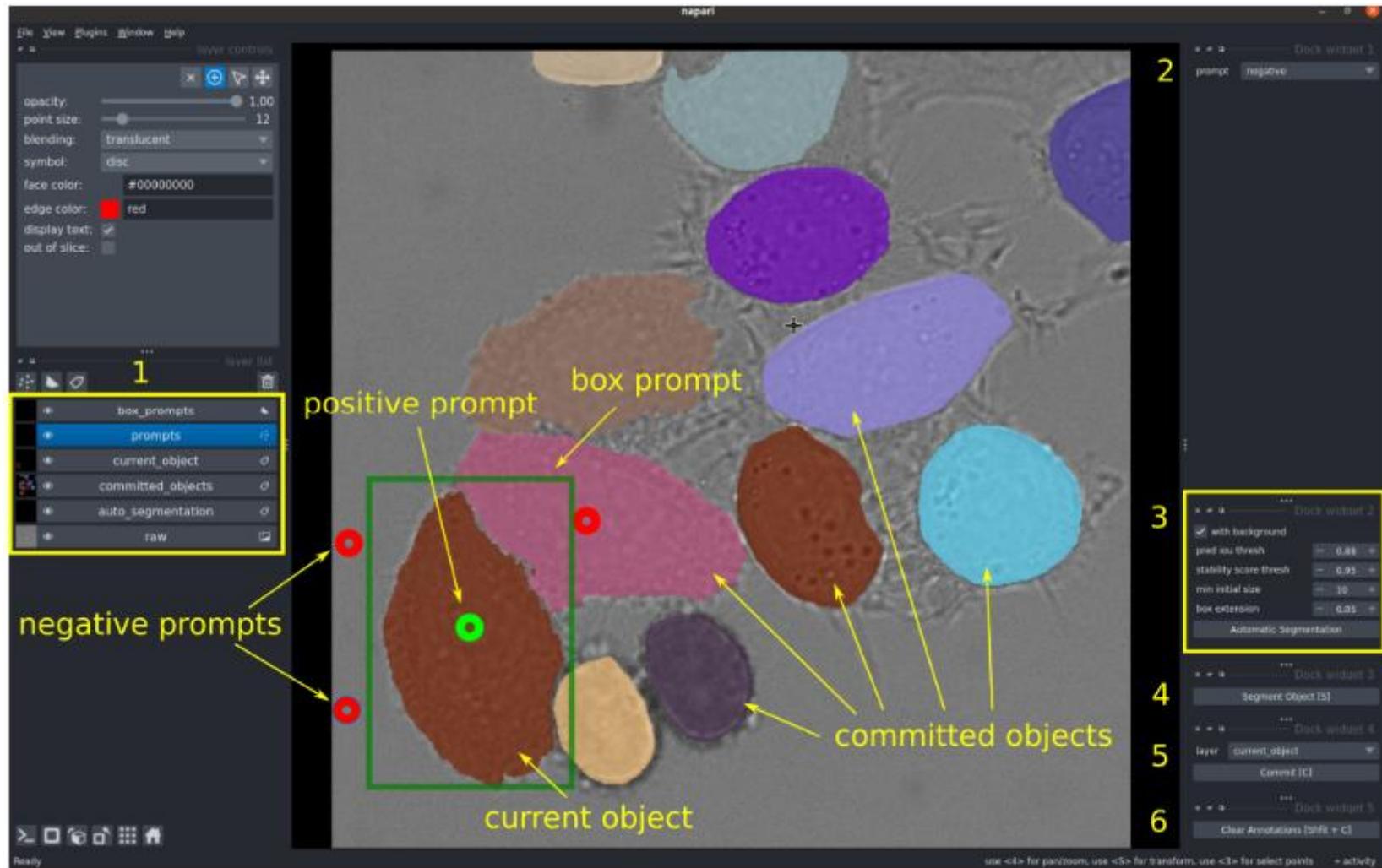
The image shows four browser tabs side-by-side, each displaying a GitHub repository page for a napari plugin:

- napari-SAM4IS**: A plugin for instance and semantic segmentation using the Segment Anything Model (SAM). It includes a screenshot of the napari interface showing a heatmap over a microscopy image.
- napari-segmentanything**: A napari plugin of Segment Anything Model (SAM), featuring a video thumbnail of the plugin in action.
- Segment Anything Model (SAM) in Napari**: A plugin that integrates the Segment Anything Model into Napari, supporting both 2D and 3D images.
- Segment Anything for Microscopy**: Tools for segmentation and tracking in microscopy build on top of Segment Anything, including a microscopy image with tracked objects.

Each repository page includes standard GitHub metrics like license (Apache-2.0), version (e.g., v0.0.6, v0.1.4, v0.4.13), Python requirements (e.g., 3.8 | 3.9 | 3.10), and test status (e.g., failing, unknown).

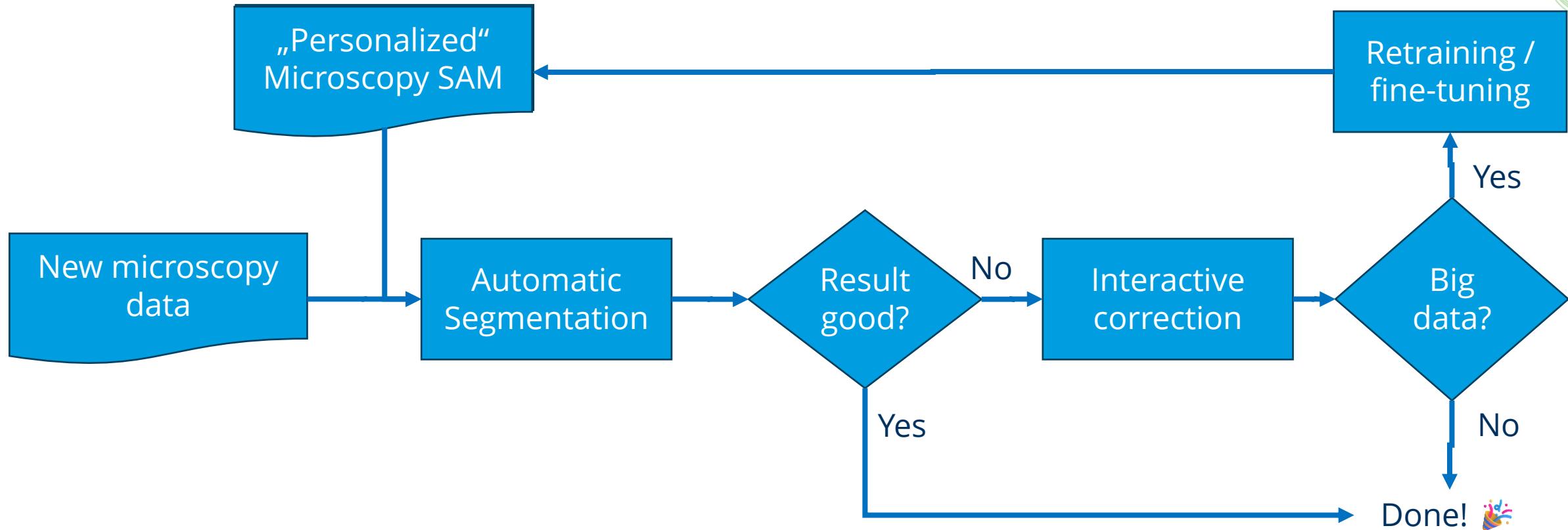
Segment Anything Model for Microscopy

Micro-sam

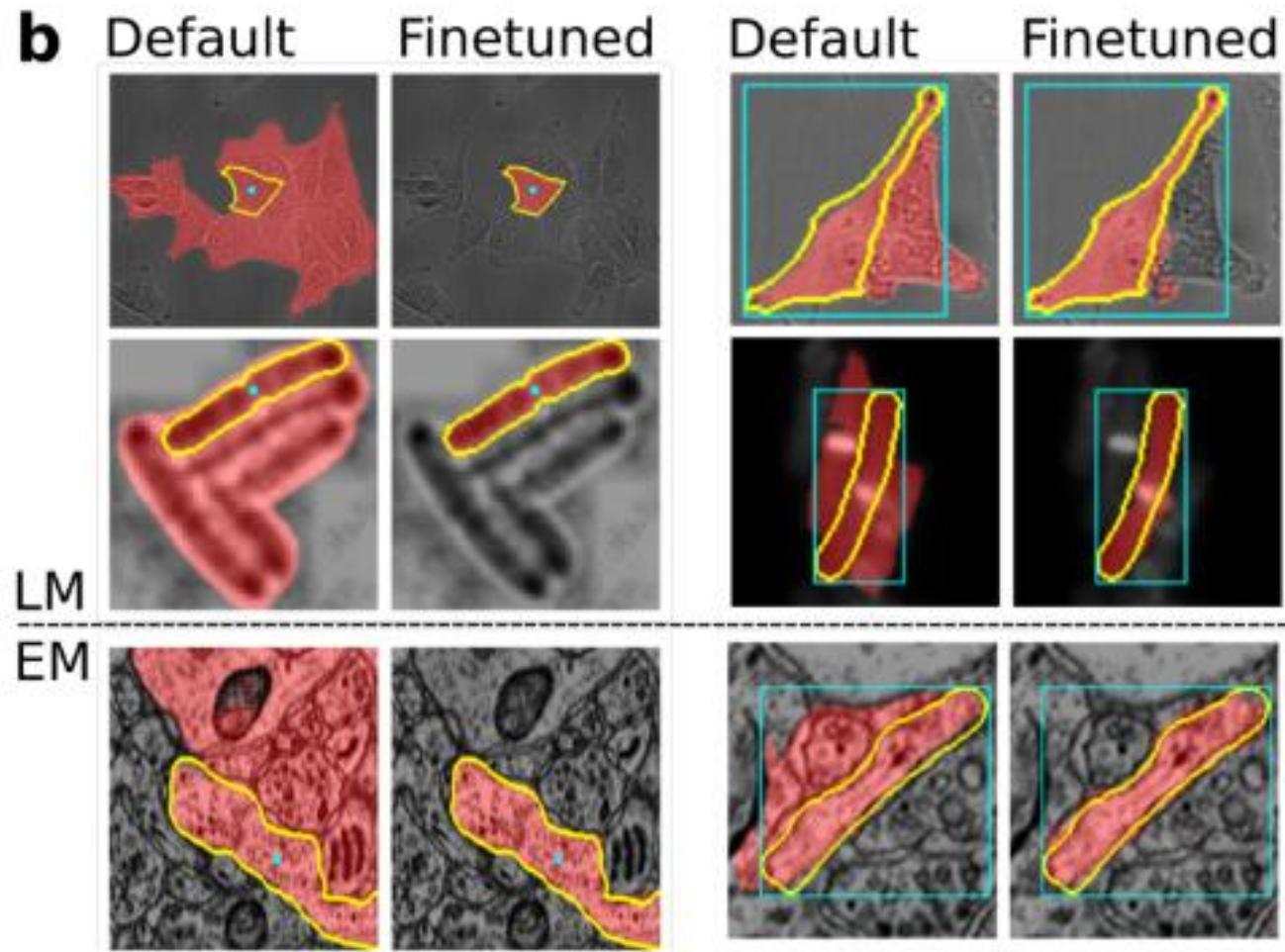


Segment Anything Model for Microscopy

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



Segment Anything Model for Microscopy



Segment Anything Model for Microscopy

In Python

v2 Models

- vit_t_lm (Zenodo): the ViT-Tiny model for segmenting cells and nuclei in LM.
- vit_b_lm (Zenodo): the ViT-Base model for segmenting cells and nuclei in LM.
- vit_l_lm (Zenodo): the ViT-Large model for segmenting cells and nuclei in LM.

v1 Models

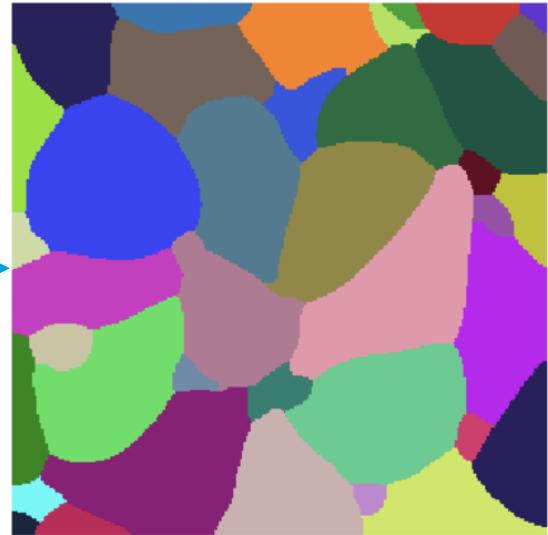
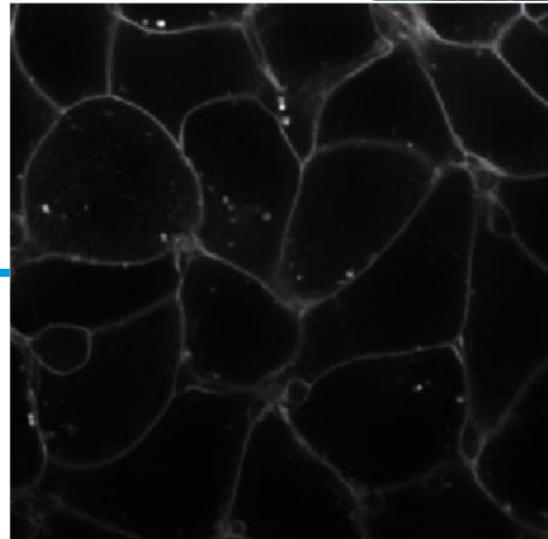
- vit_b_em_boundaries: for segmenting compartments delineated by boundaries such as cells or neurites in EM.
- vit_b_em_organelles: for segmenting mitochondria, nuclei or other organelles in EM.
- vit_b_lm: for segmenting cells and nuclei in LM.
- vit_h_em: for general EM segmentation.
- vit_h_lm: for general LM segmentation.

```
# Load model
predictor, segmenter = get_predictor_and_segmenter(model_type="vit_b_lm")

# Apply model
label_image = automatic_instance_segmentation(predictor=predictor, segmenter=segmenter, input_path=image)

# Visualize result
stackview.insight(label_image)
```

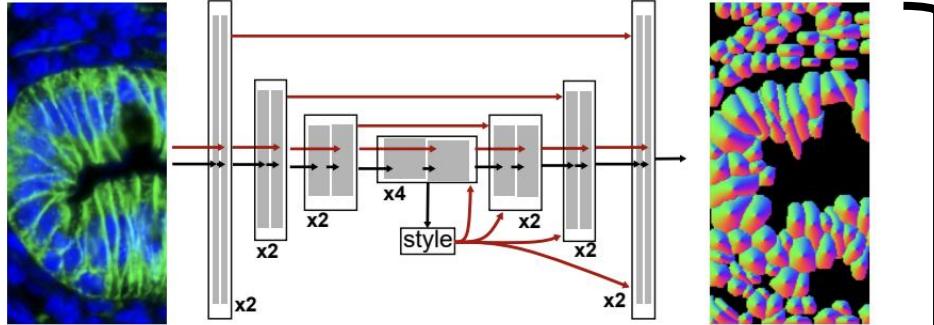
Compute Image Embeddings 2D: 100% | 1/1 [00:01<00:00, 1.08s/it]
Initialize instance segmentation with decoder: 100% | 1/1 [00:00<00:00, 3.37it/s]



CellPose-SAM

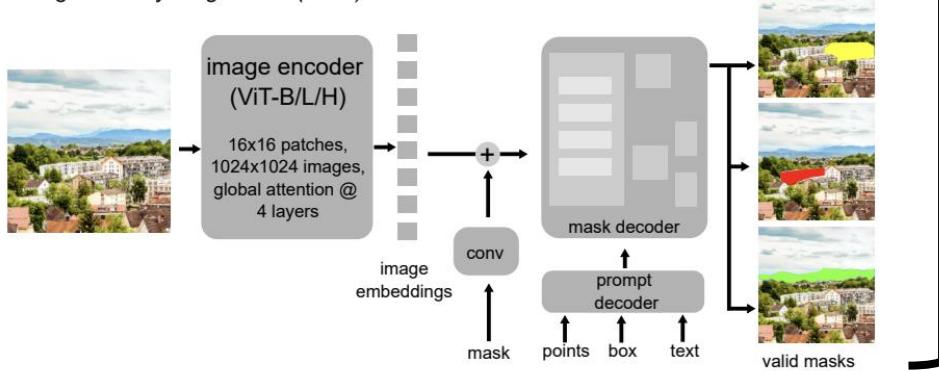
Recent combination of CellPose + Segment Anything Model (May 1st 2025)

a



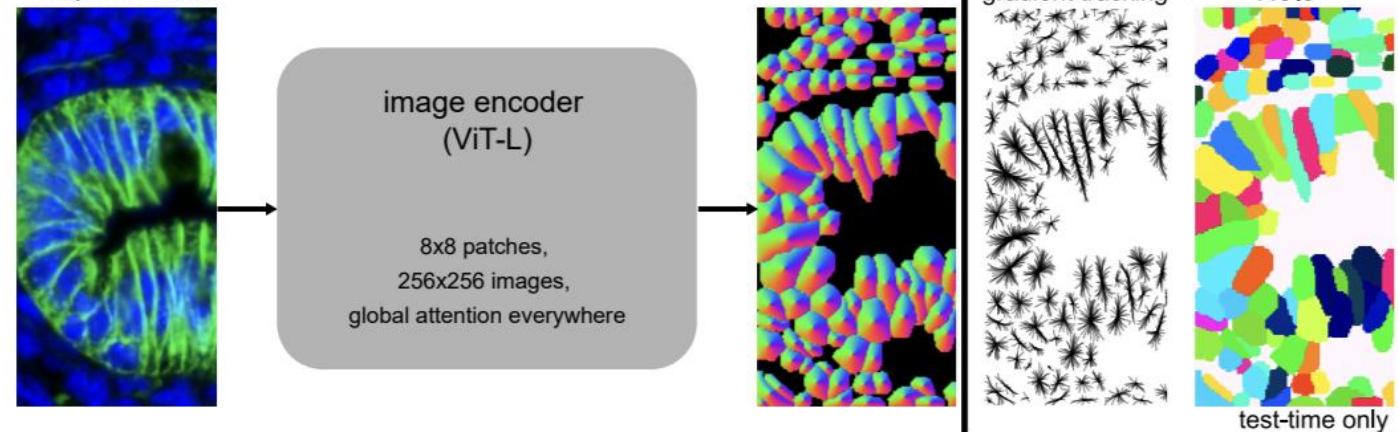
b

Segment Anything Model (SAM)



c

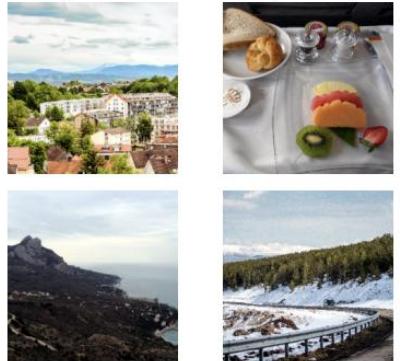
Cellpose-SAM



CellPose-SAM: Model fine-tuning

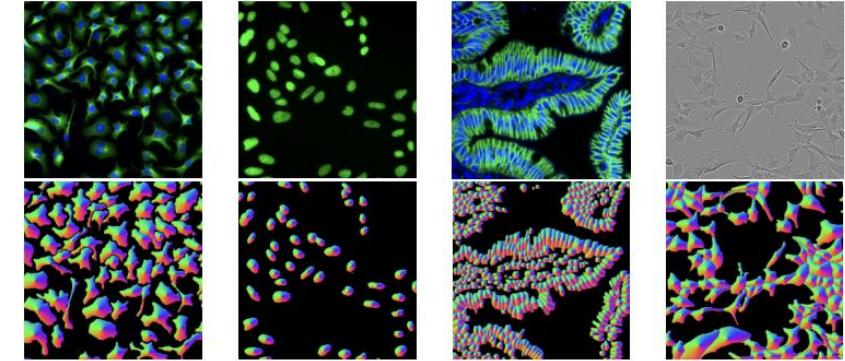
Model fine-tuning to make it a microscopy-specific image processing model

Training dataset for SAM (SA-1B)

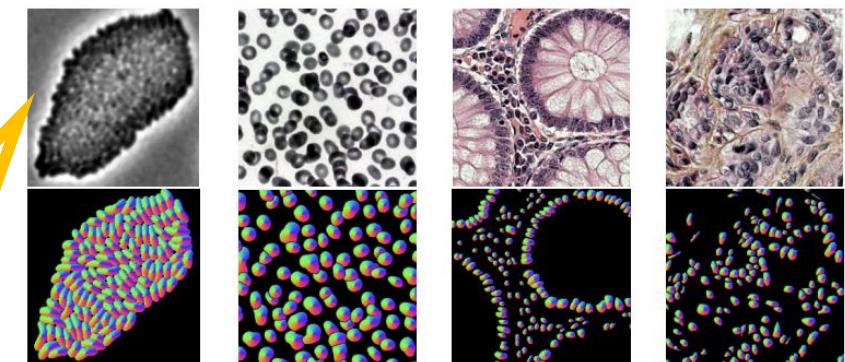


300.000 images, 10.2 million manual ROIs

Finetuning Cellpose-SAM on cellular datasets

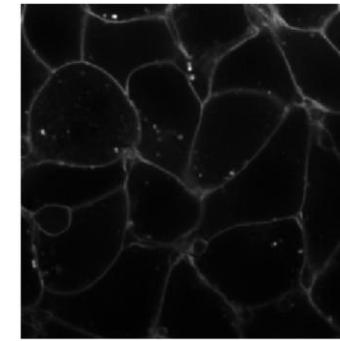
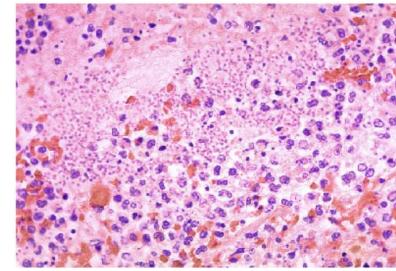
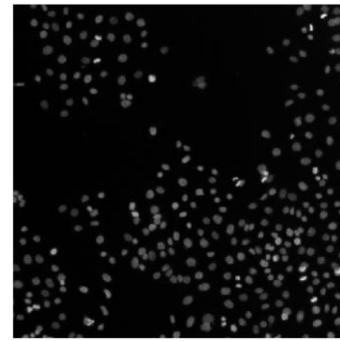
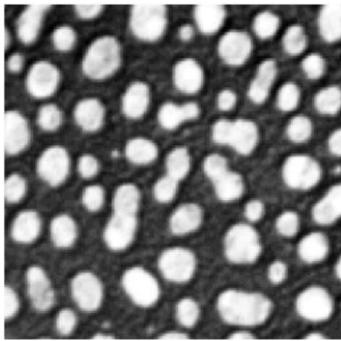


Based on publicly available benchmark data (Grand Challenges etc.)

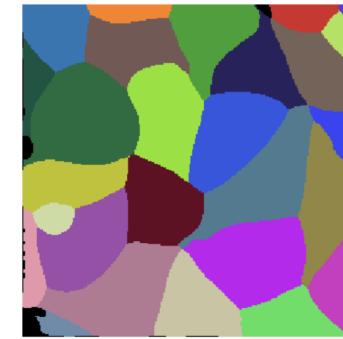
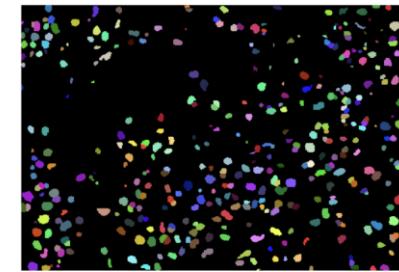
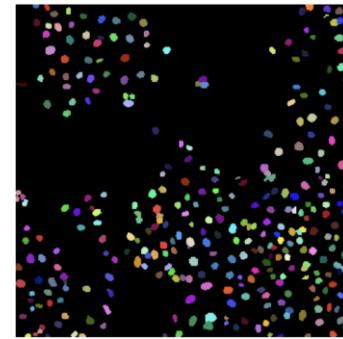
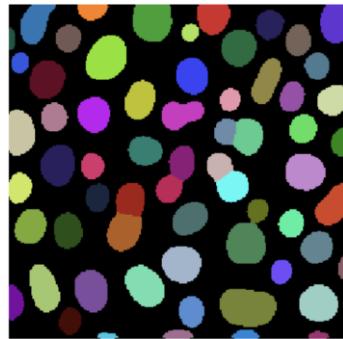


22, 826 images, 3.34 million manual ROIs

CellPose SAM in Python

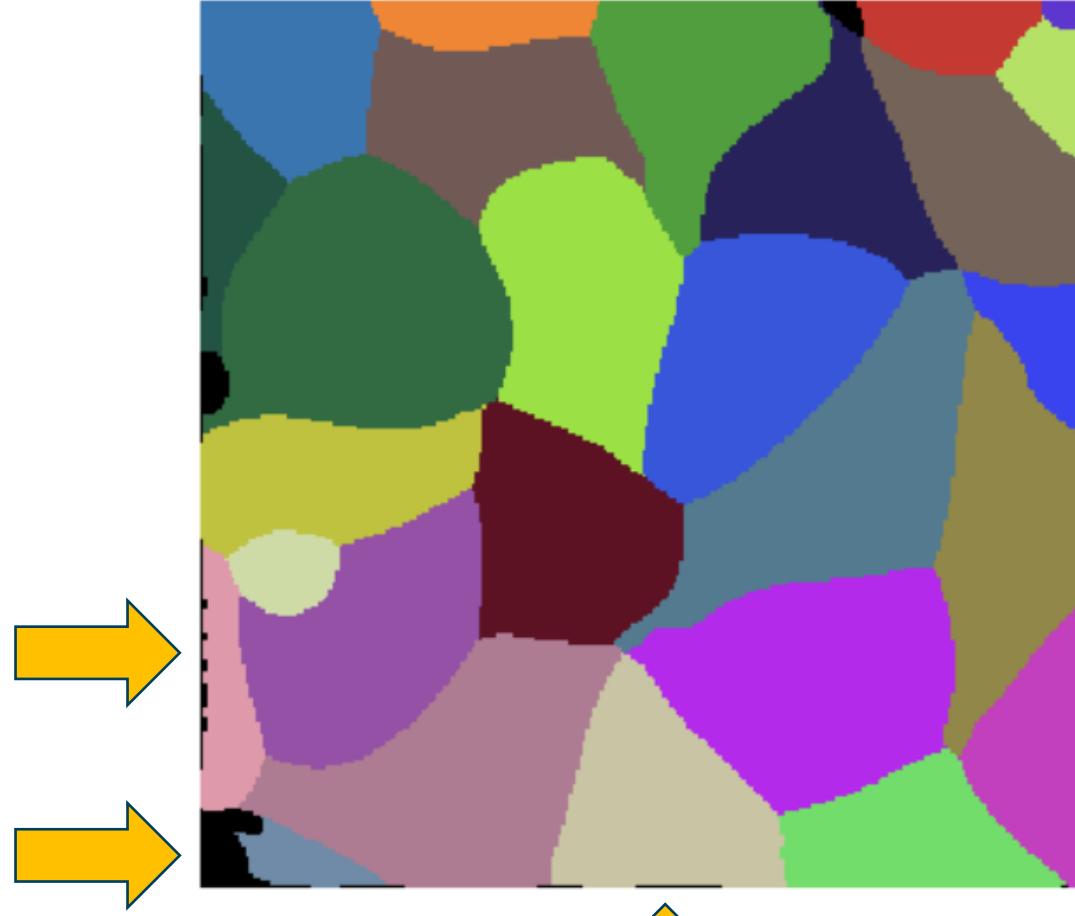


```
masks, flows, styles = model.eval(image,  
batch_size=32,  
flow_threshold=0.4,  
cellprob_threshold=0.0,  
normalize={"tile_norm_blocksize": 0})  
  
stackview.insight(masks.astype(np.uint32))
```



“Foundation
model”

Quiz: What could we do to deal with such artifacts at the image border?



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 images / full annotations
~~(single images, sparse annotations)~~,
 - 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



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, micro-sam and cellpose-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 conda.

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

The screenshot shows two command prompts and a Jupyter Notebook cell. The left command prompt shows the successful uninstallation of several packages. The right command prompt shows the successful uninstallation of matplotlib. In the Jupyter Notebook cell, a user attempts to import n2v and skimage.io.imread. This triggers a ValueError because the notebook's environment has incompatible numpy versions. The error message indicates that numpy 1.19.5 is incompatible with the required version. The code in the cell is as follows:

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

ValueError
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