

Introduction to napari

Robert Haase

With material from

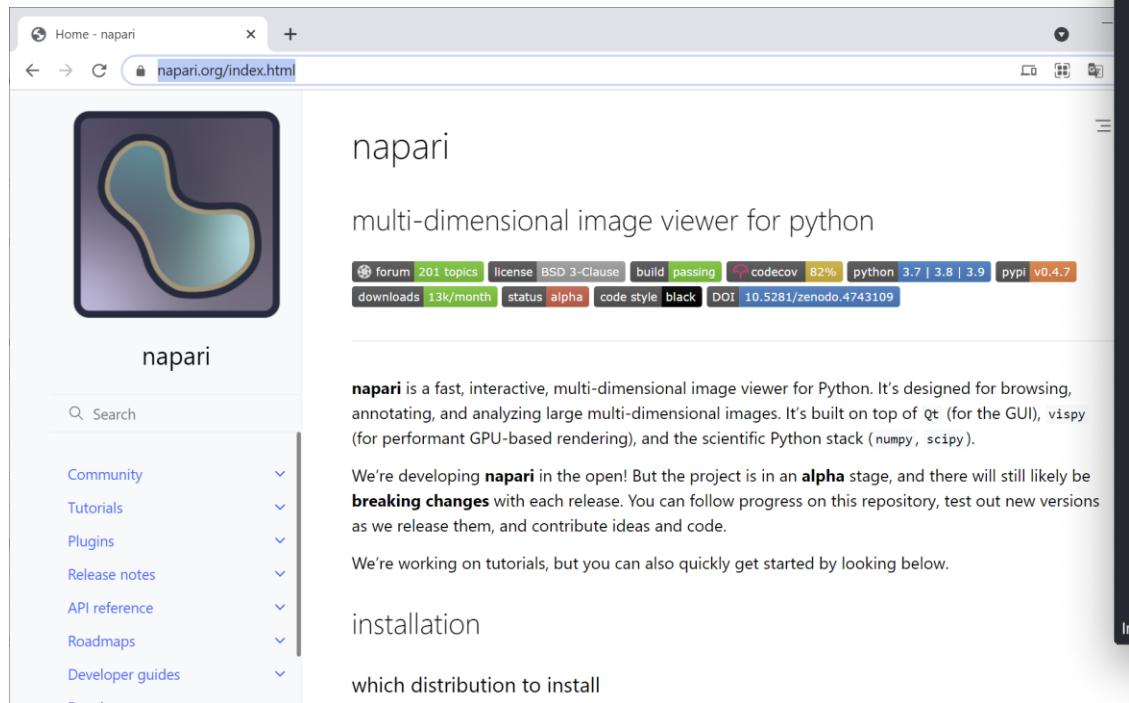
Marcelo Leomil Zoccoler, Physics of Life, TU Dresden

December 2023

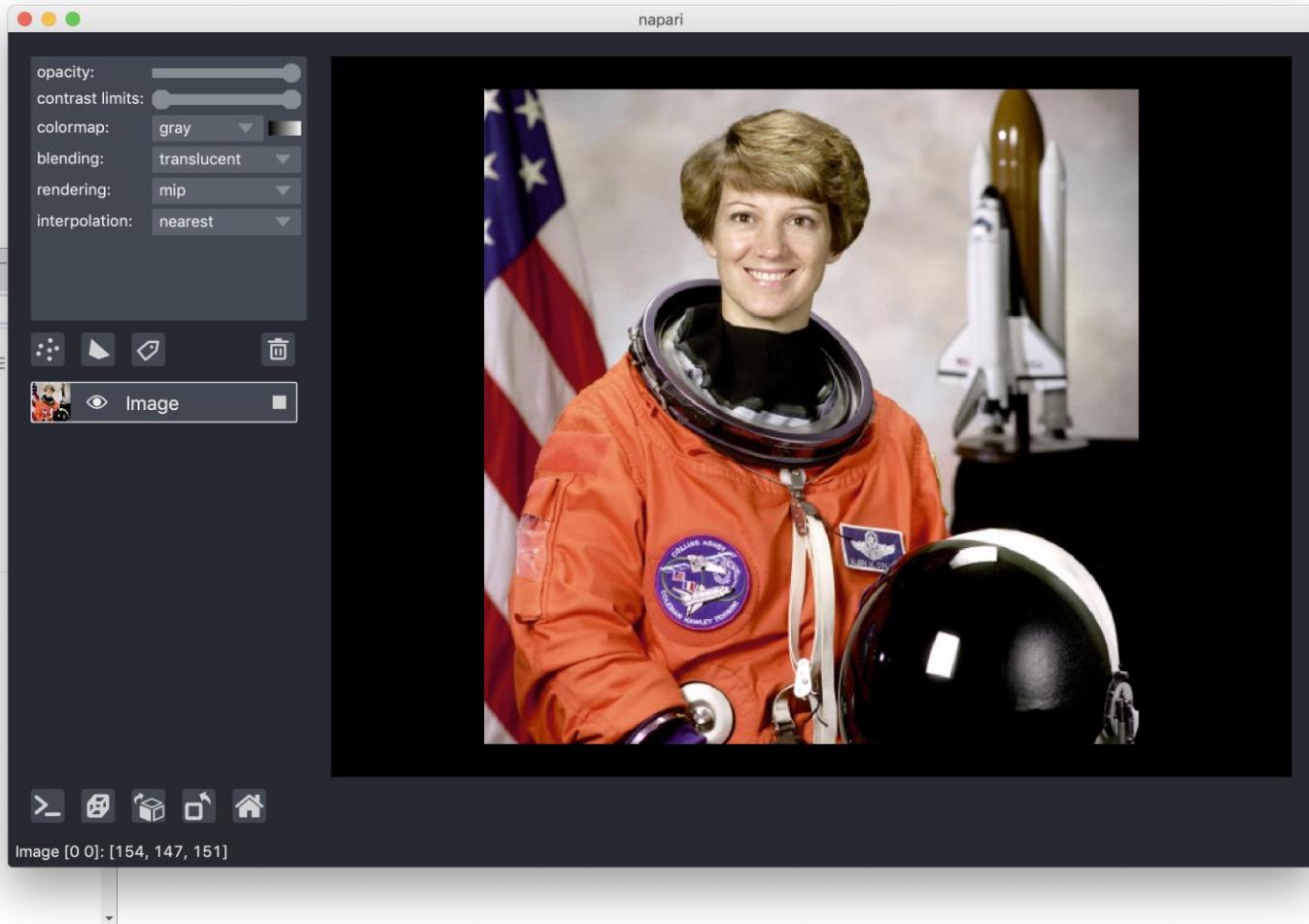
These slides are licensed CC-BY 4.0 unless mentioned otherwise

Napari: 3D viewer for Python

- Multi-dimensional image viewer in python



The screenshot shows the official napari website at napari.org/index.html. It features a sidebar with links for Home, Search, Community, Tutorials, Plugins, Release notes, API reference, Roadmaps, and Developer guides. The main content area displays the napari logo, a brief description of the software as a fast, interactive, multi-dimensional image viewer for Python, and sections for installation and distribution selection.

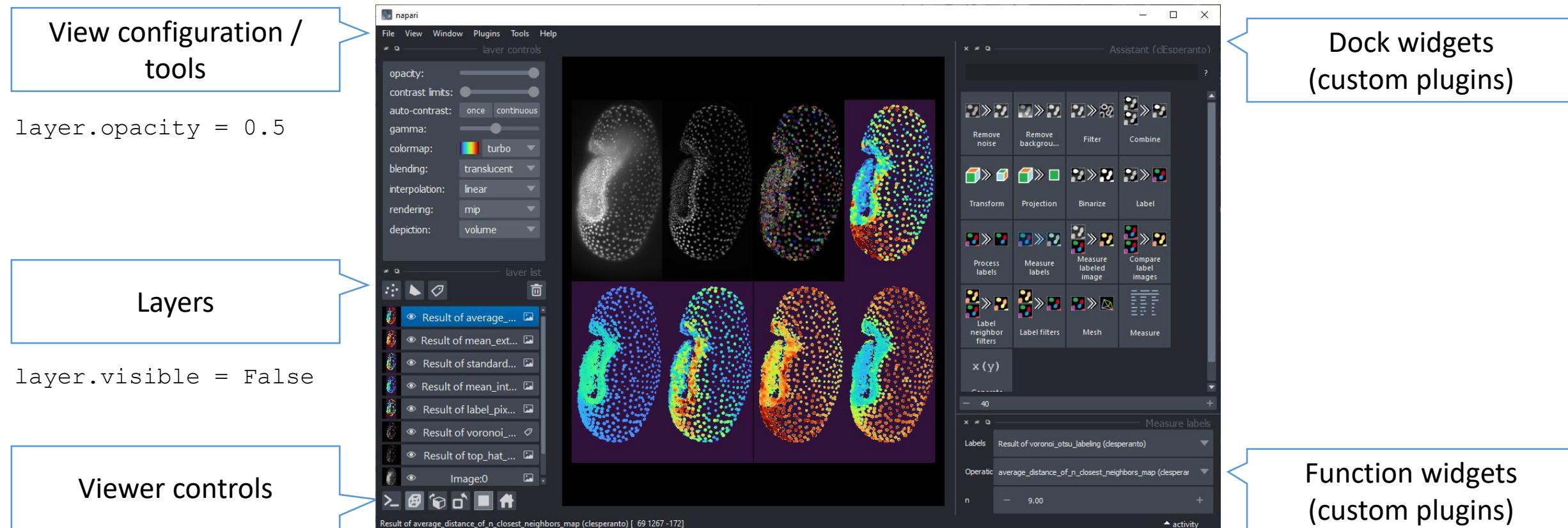


Napari: 3D viewer for Python



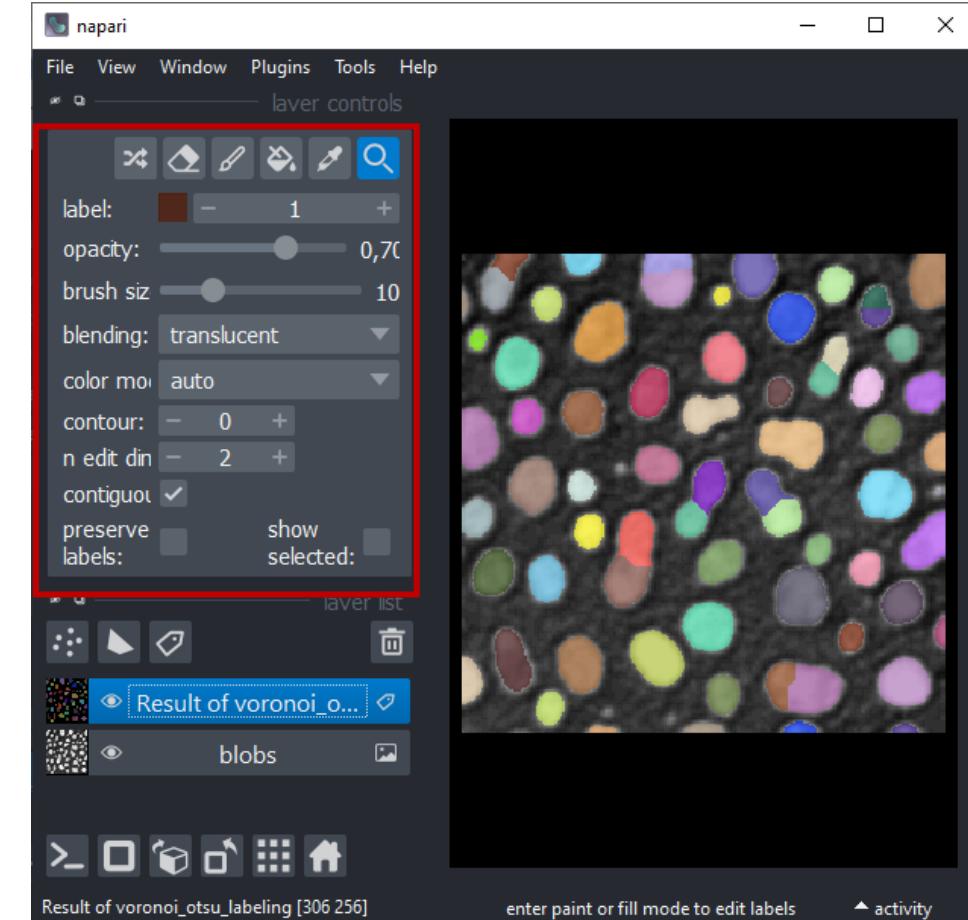
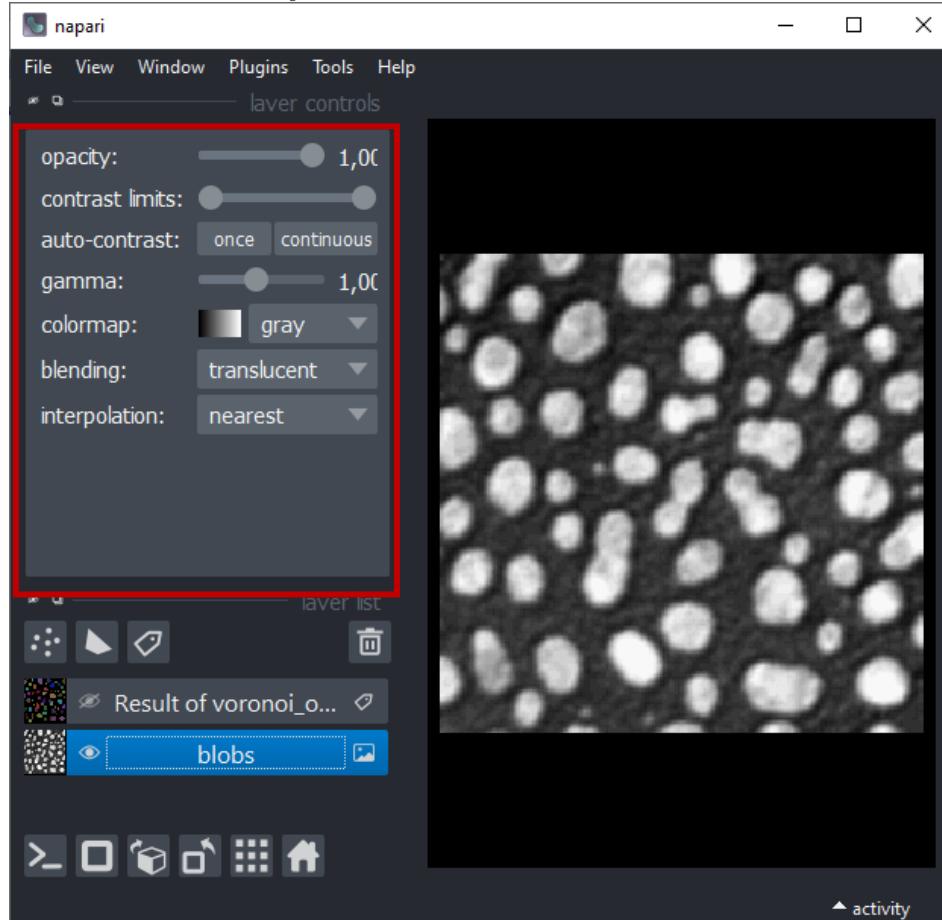
<https://napari.org/>

Napari user interface



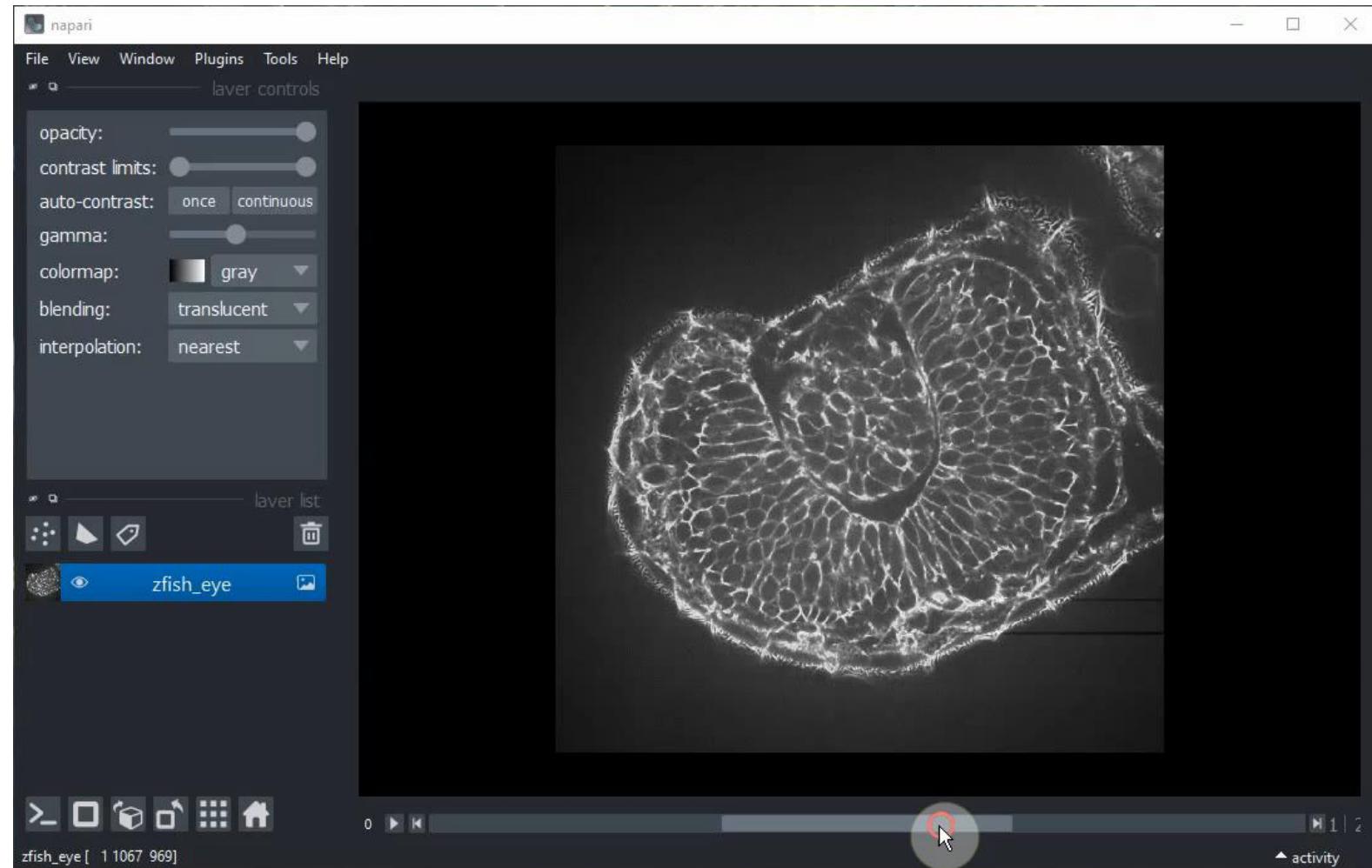
Layer types

- Different layers have different tools and options



Napari core

- Splitting channels & data visualization



Napari core

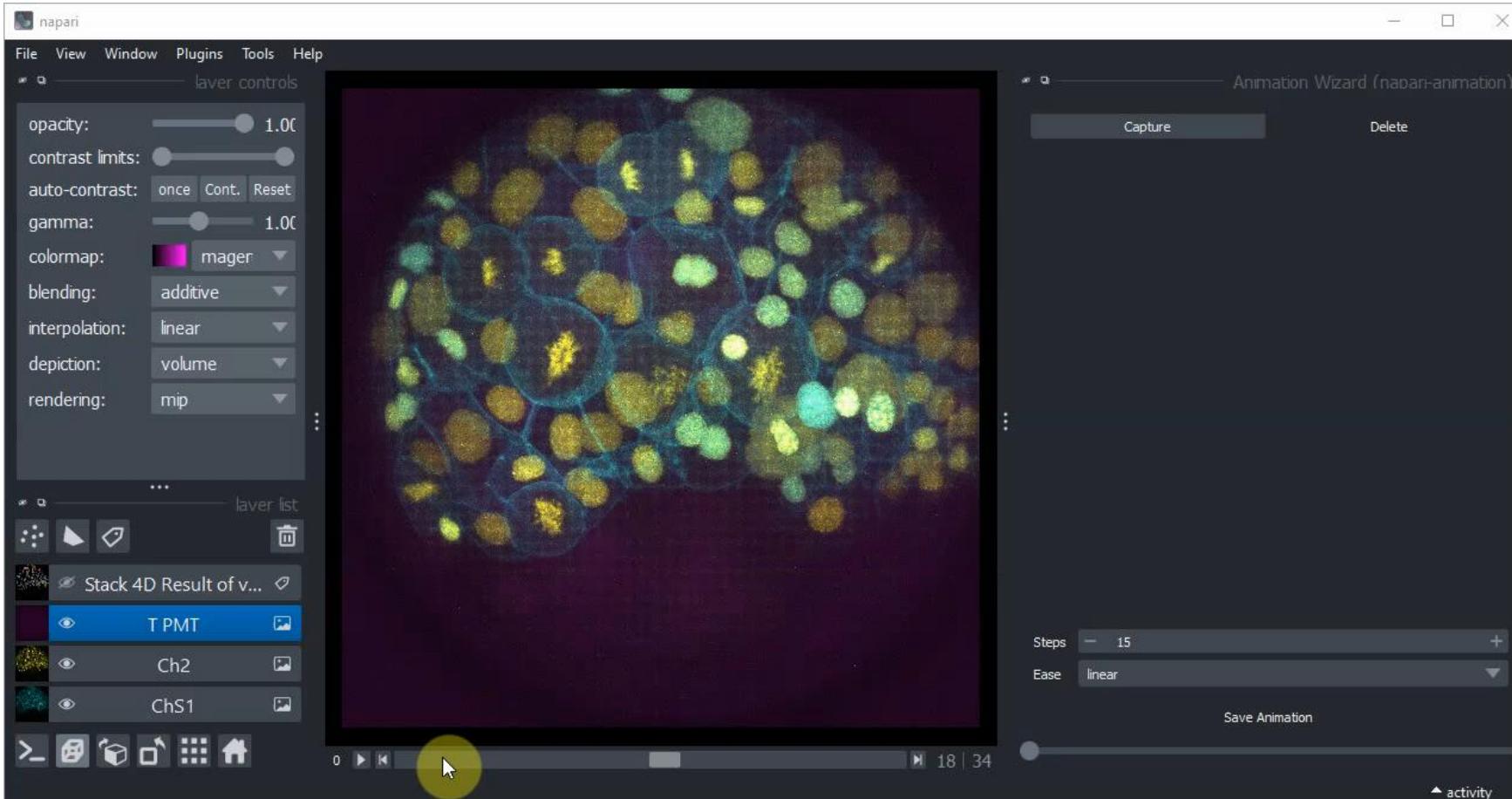
- Brightness & contrast

Napari-brightness-contrast

- Brightness & contrast

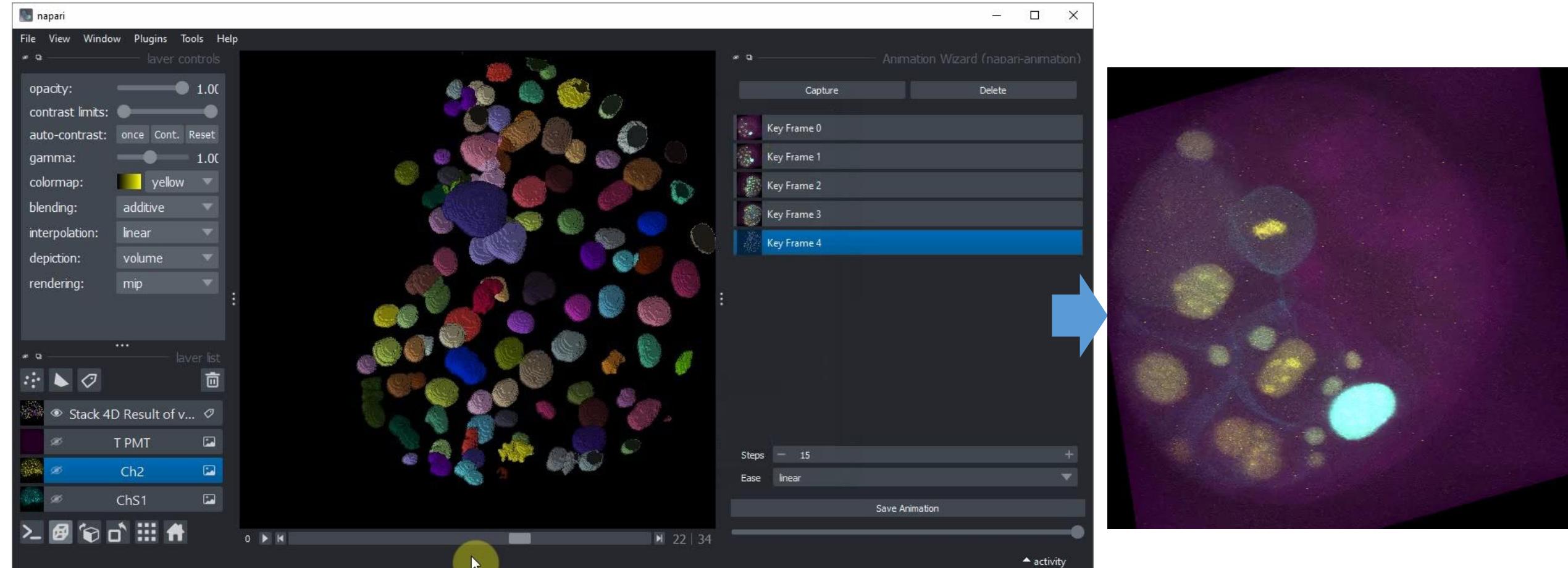
Napari-animation

- Making animations – as easy as it gets

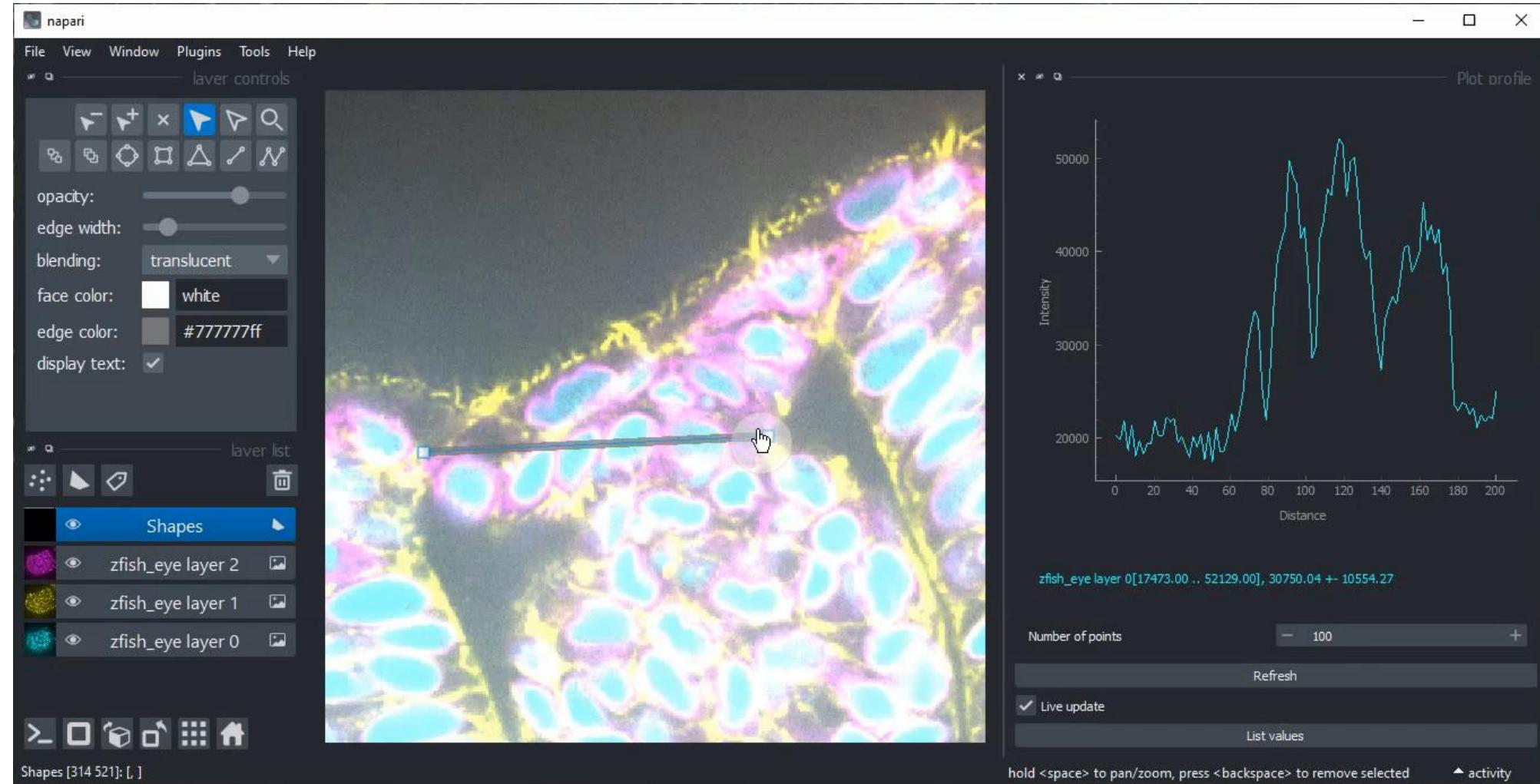


Napari-animation

- Making animations – as easy as it gets

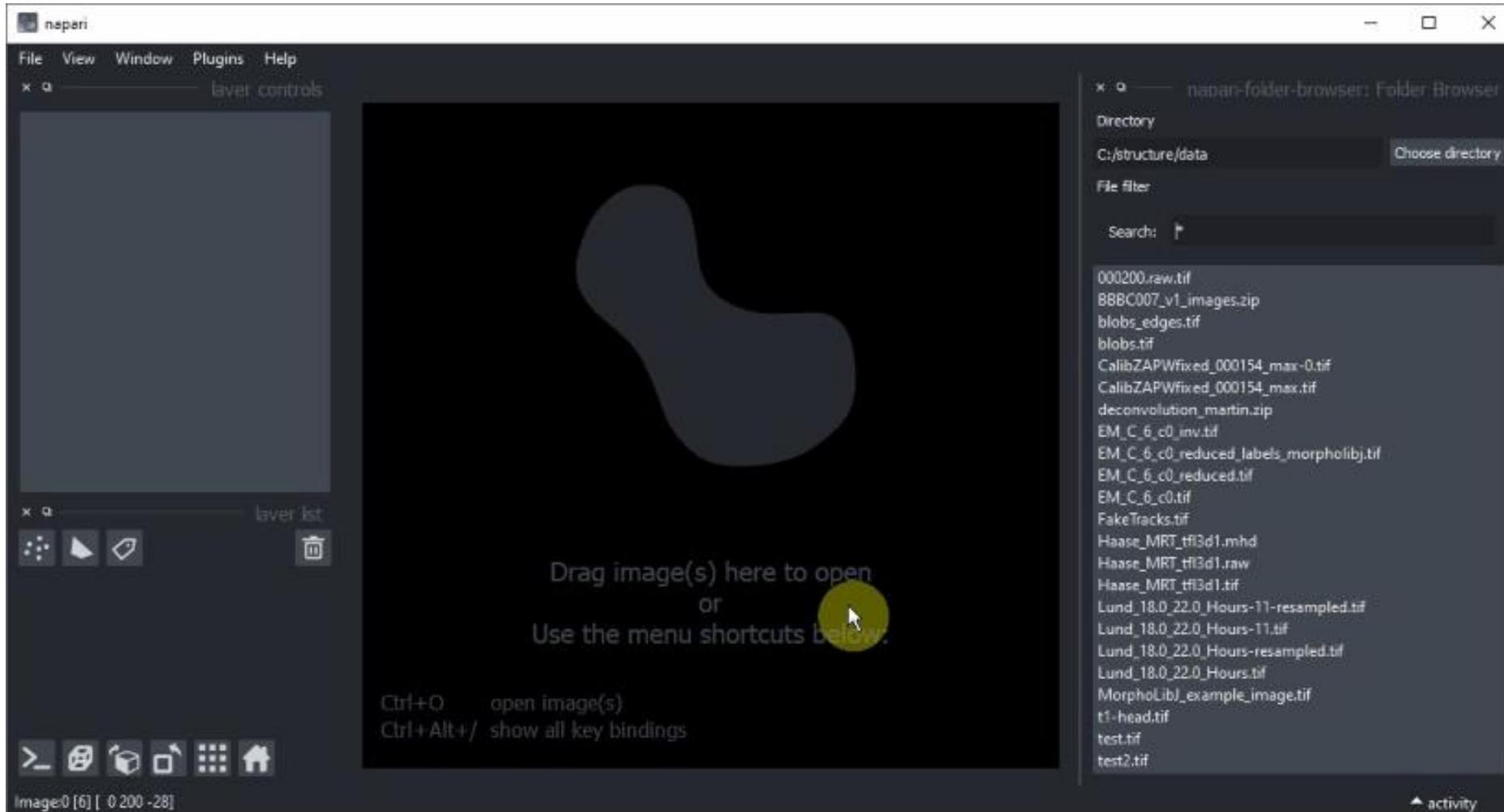


napari-plot-profile

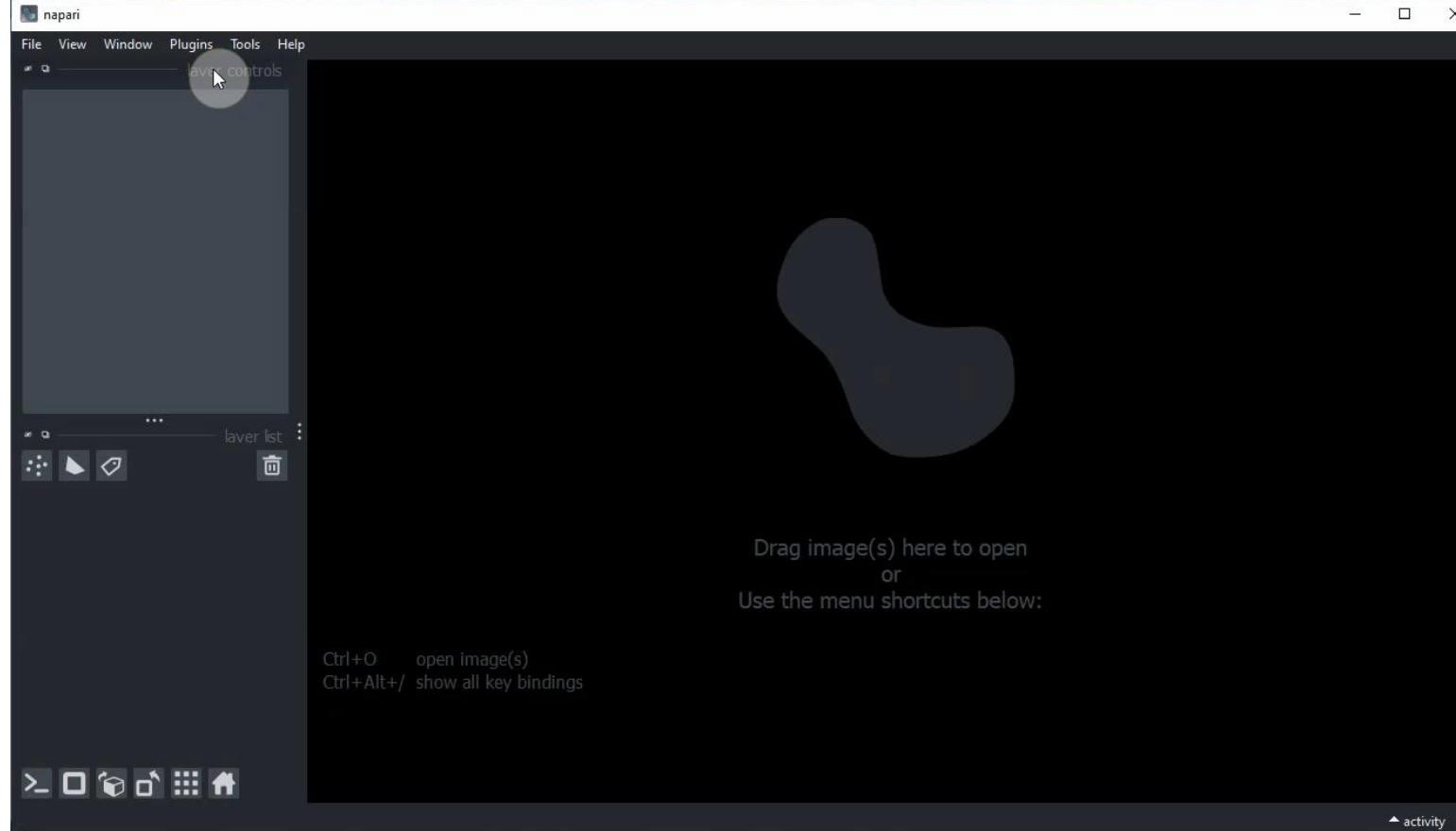


Demo: Visualizing image data in Napari

Napari folder browser



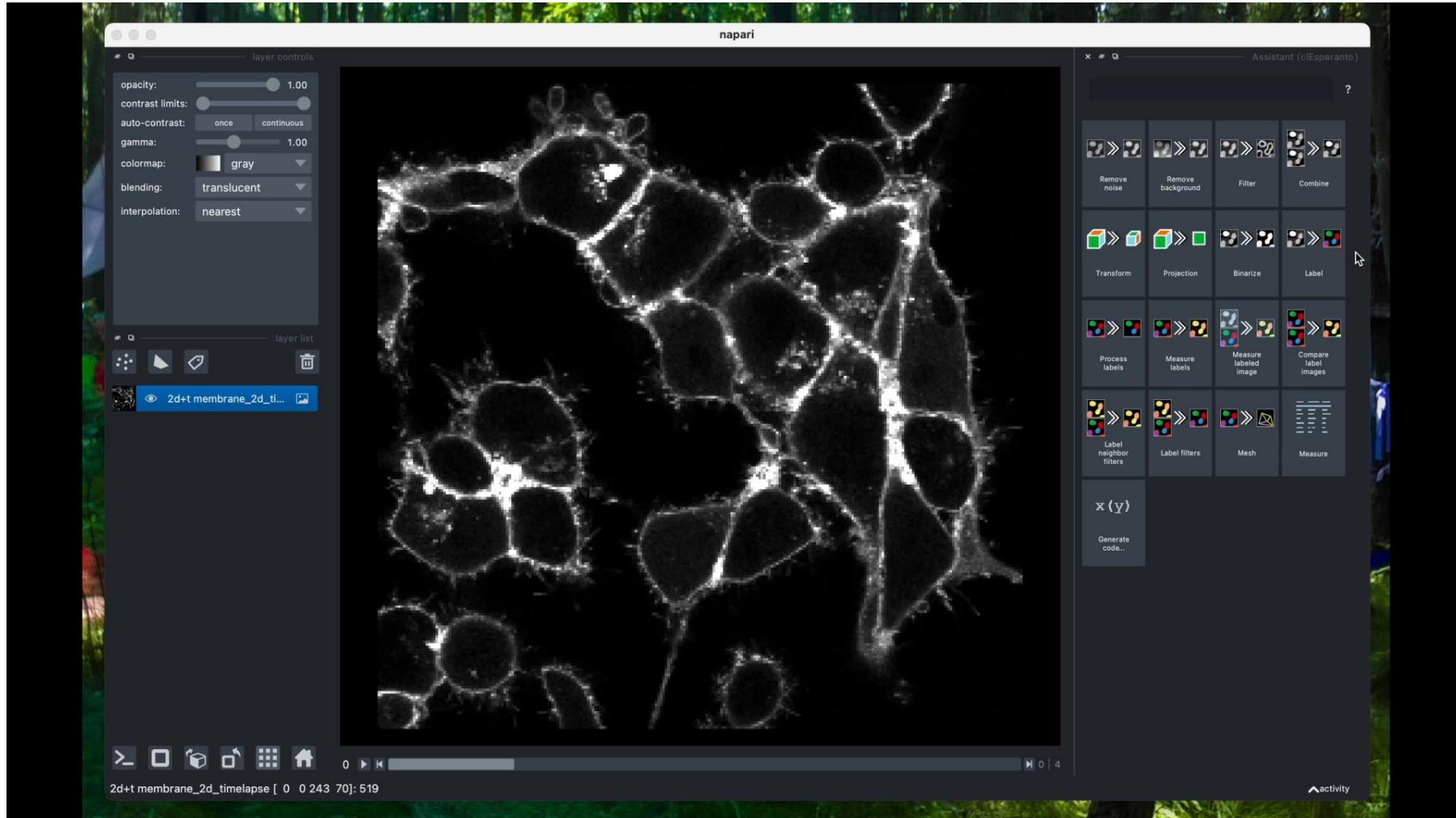
Napari-owncloud



Napari-omero

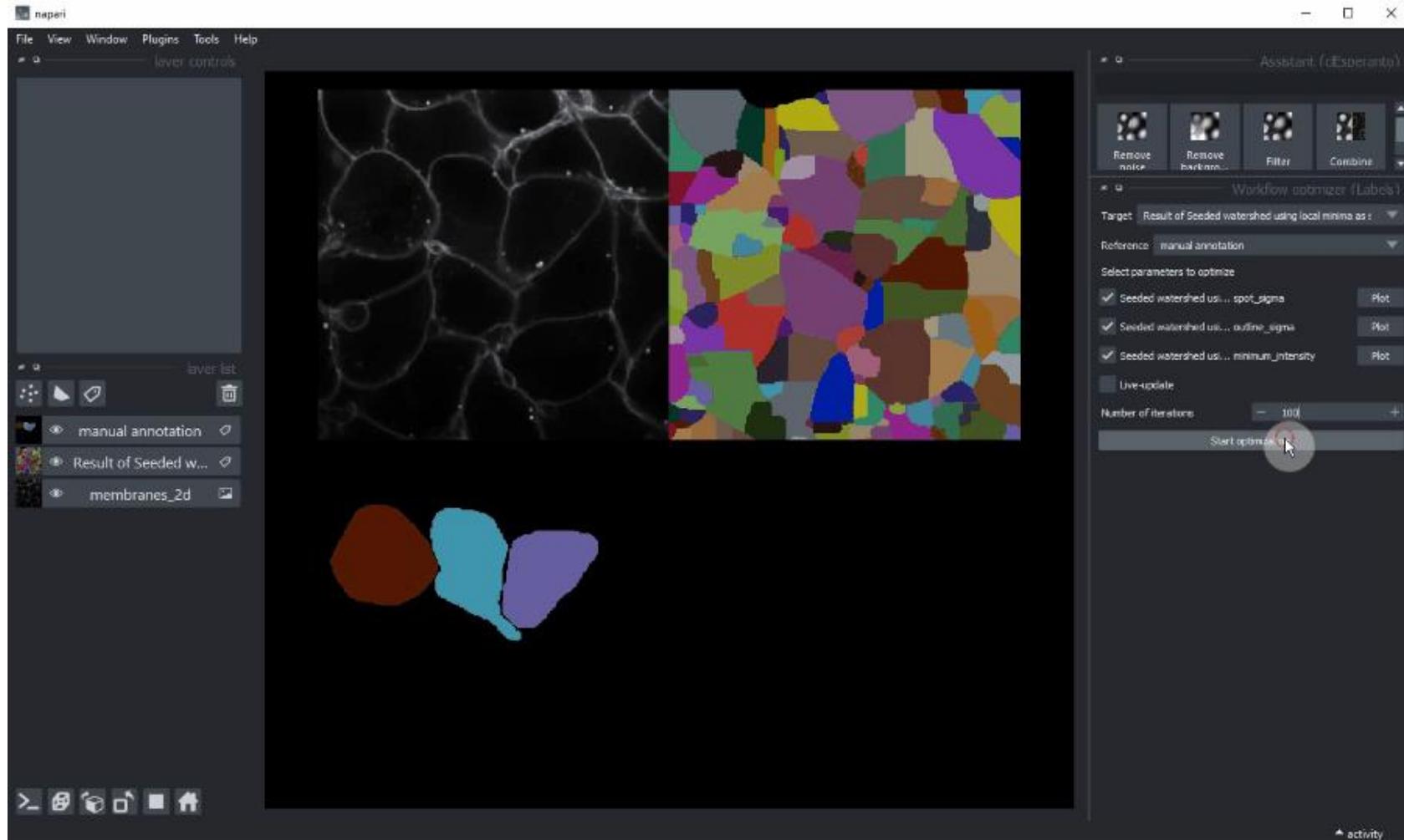
Napari, segment blobs and things with membranes!

- Filtering,
- thresholding,
- spot detection,
- seeded watershed



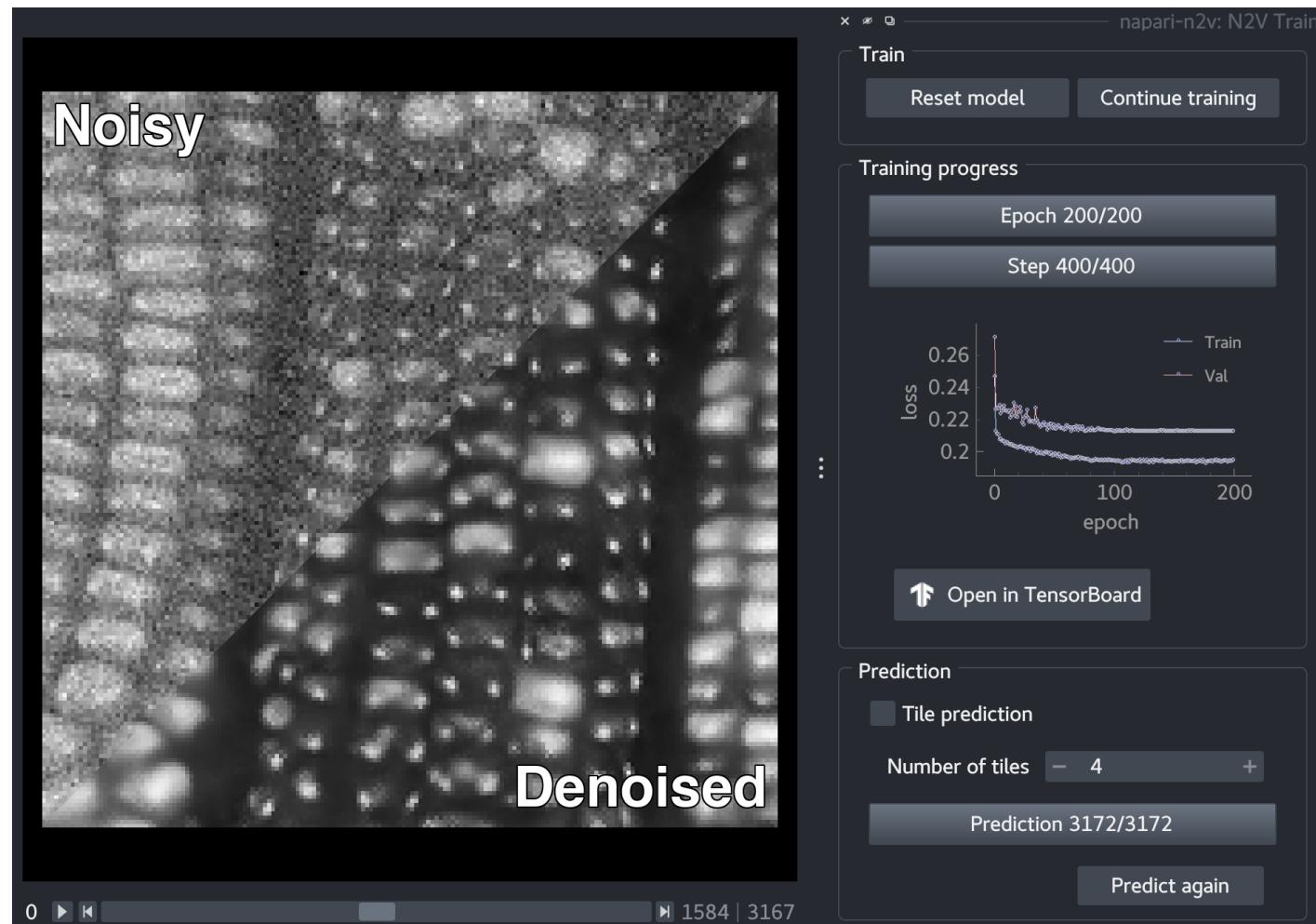
Napari-workflow-optimizer

- Automatic optimization of workflow parameters



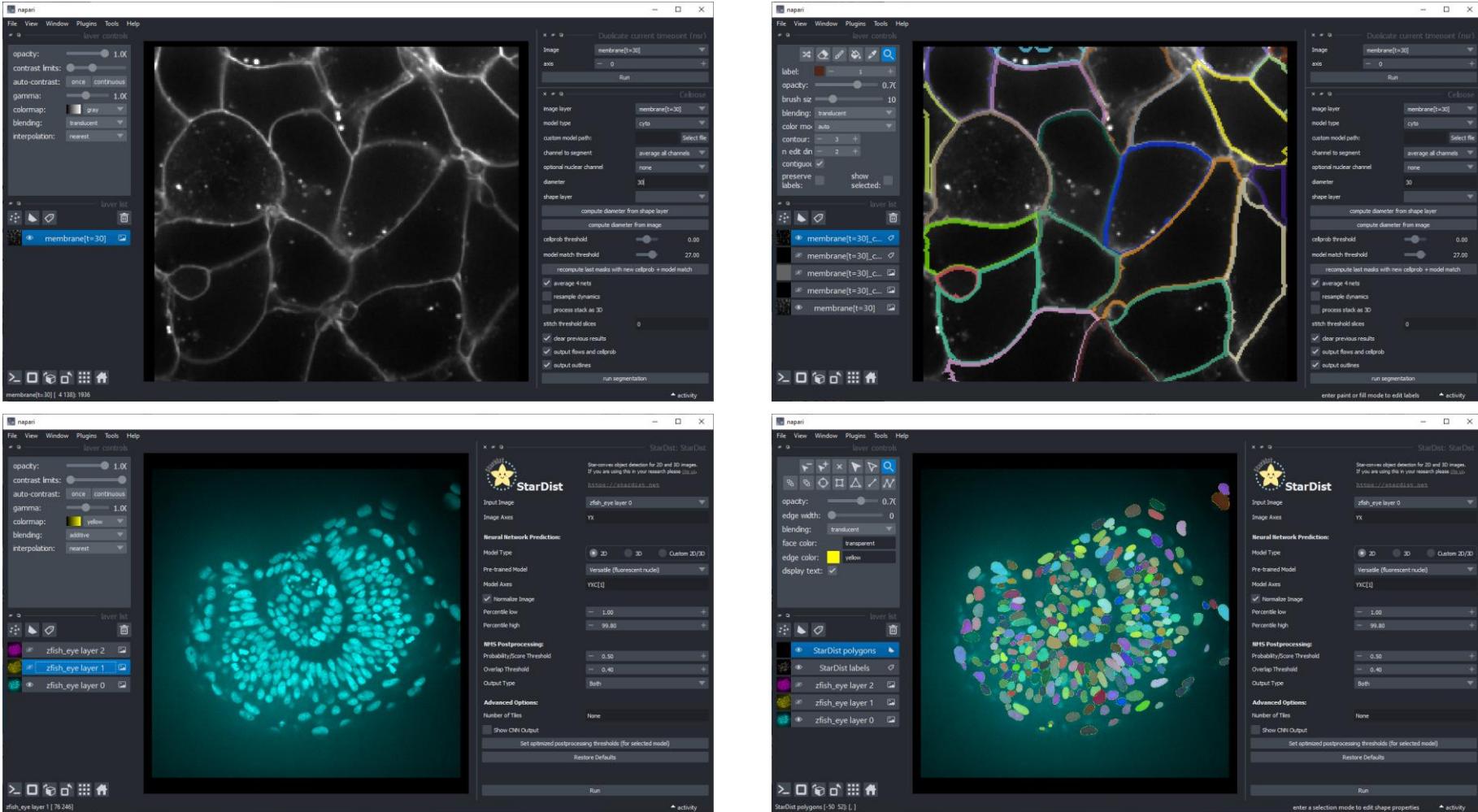
Deep-Learning based denoising

- Train a DL/ML model with a custom user interface
- Noise2void (n2v)



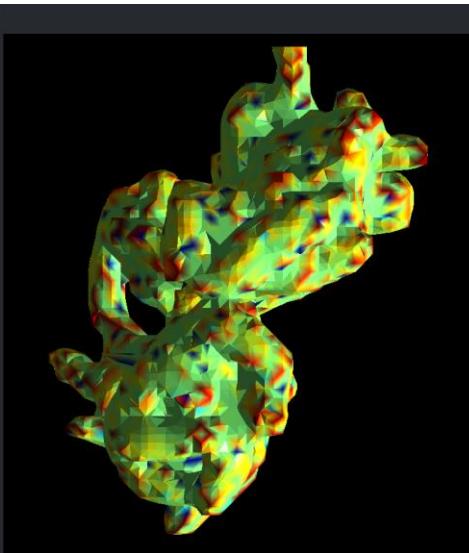
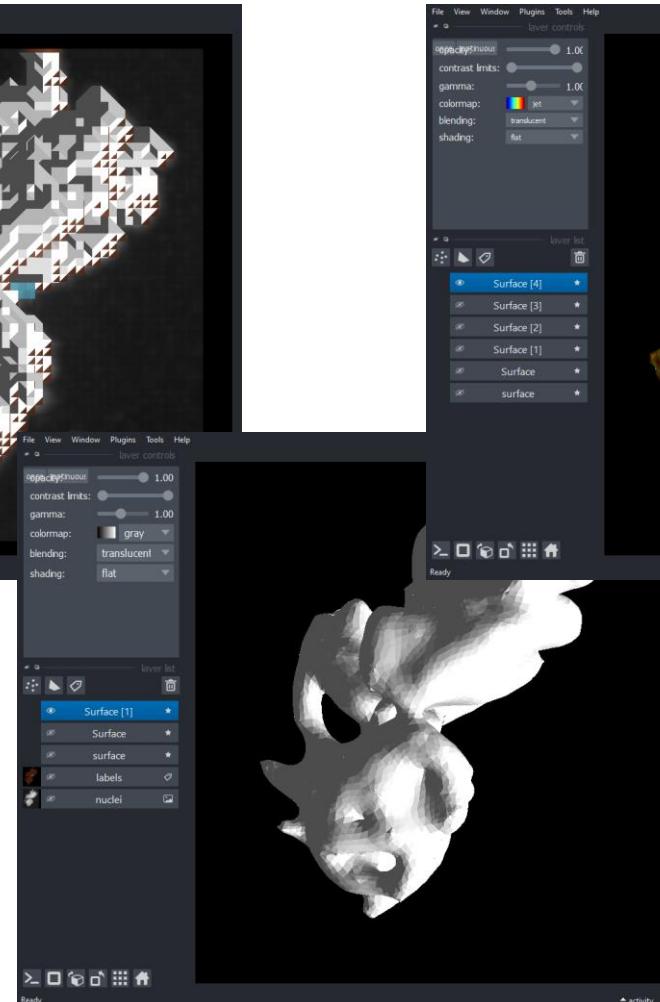
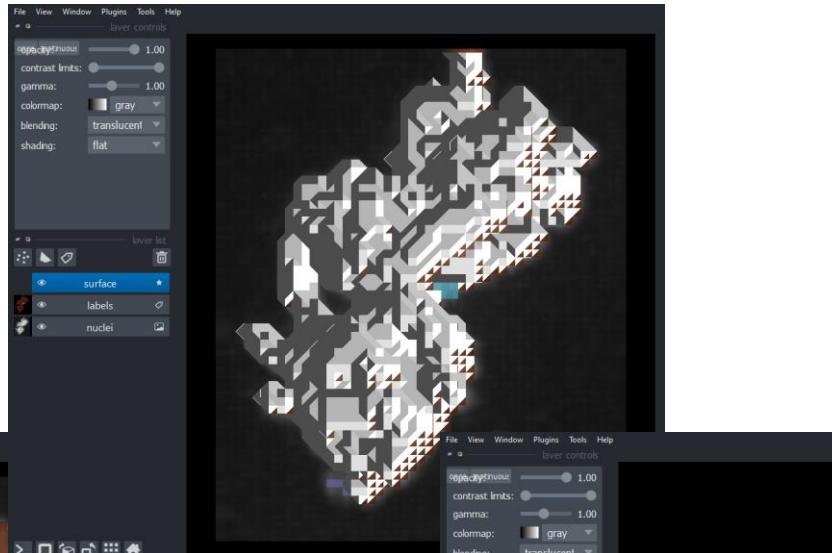
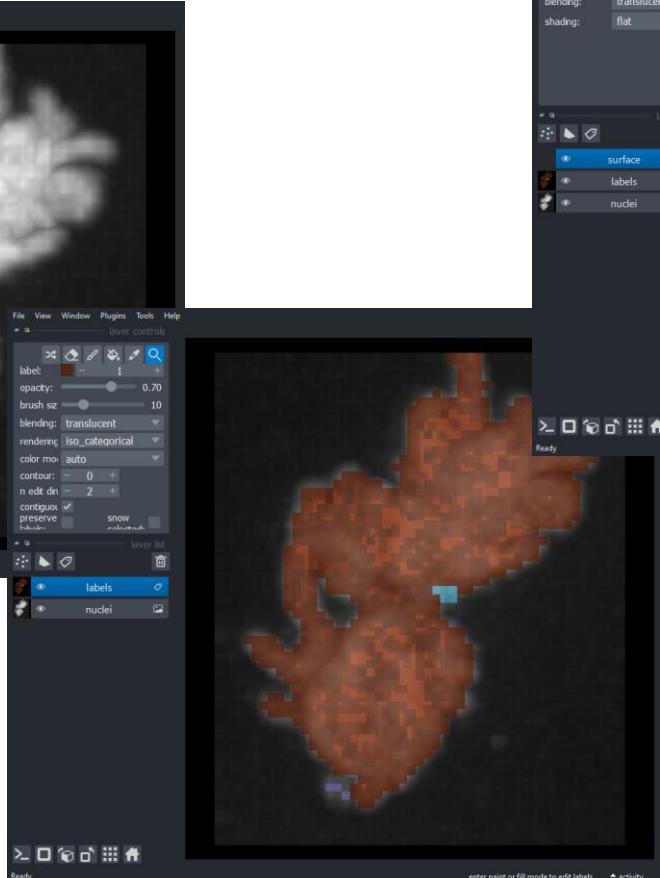
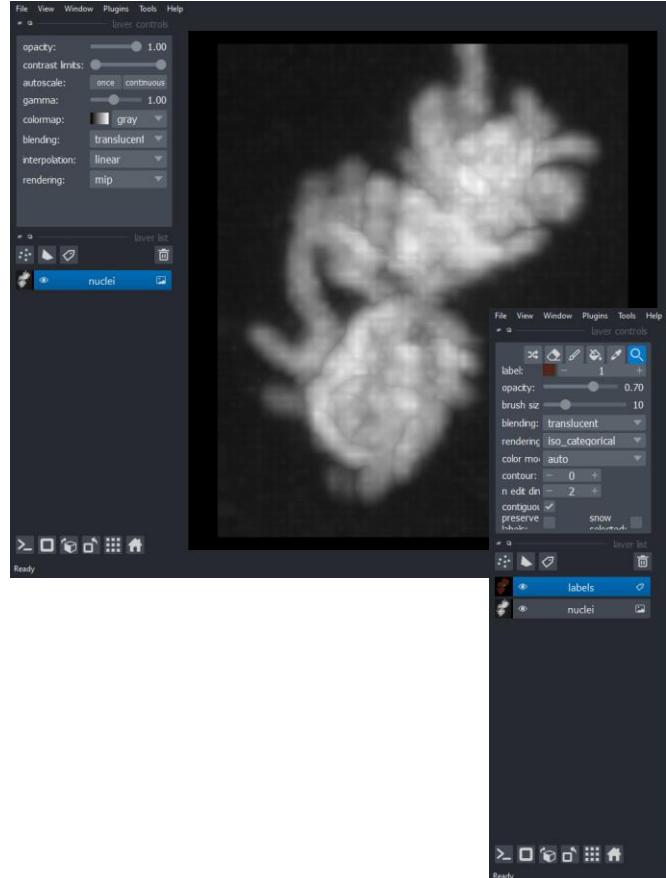
Deep learning based cell + nuclei/cell segmentation

- CellPose
- StarDist

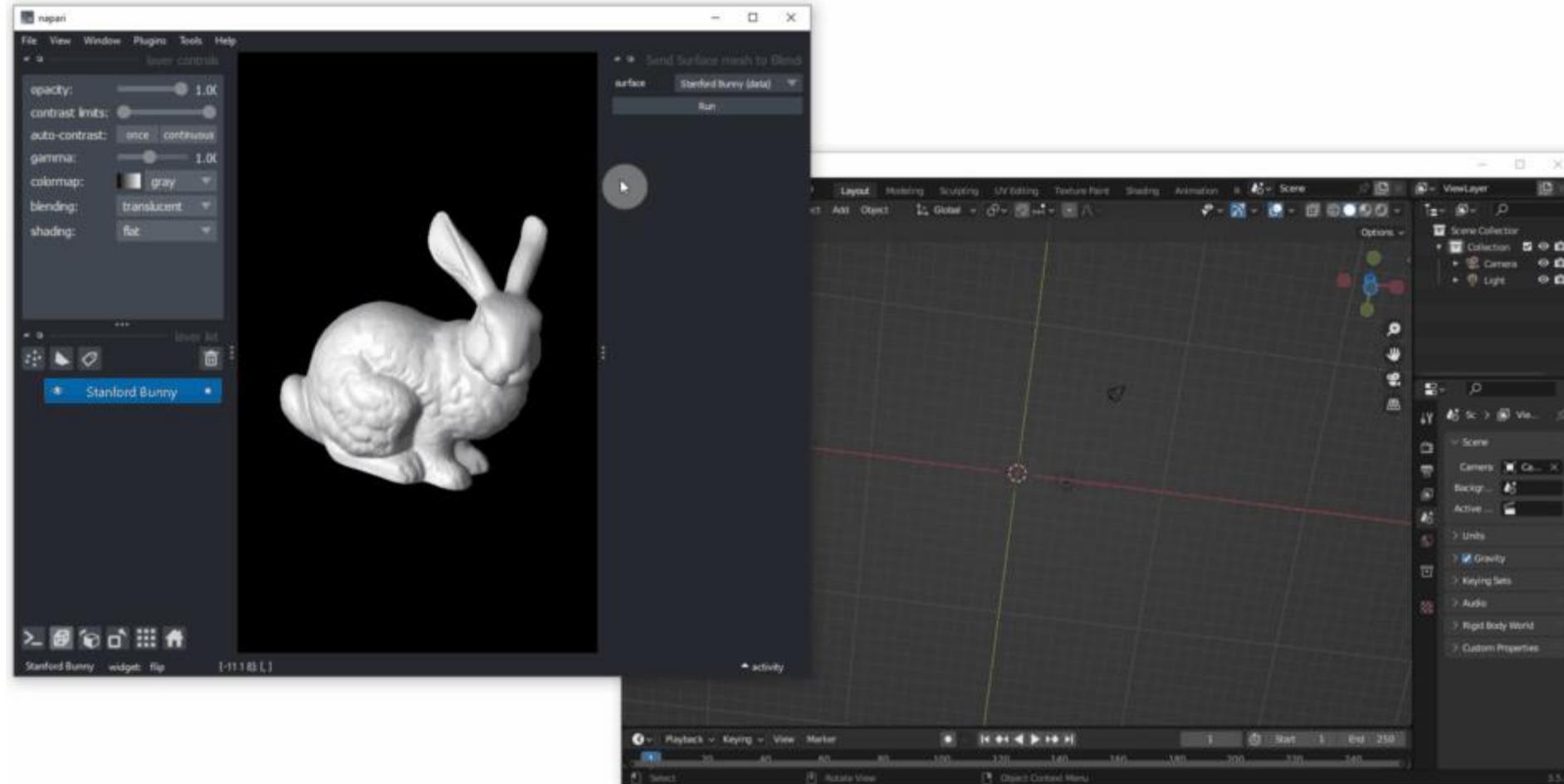


Napari process points and surfaces

- Surface extraction & analysis



Napari-blender-bridge



Napari-clusters-plotter

- ... and unsupervised machine learning

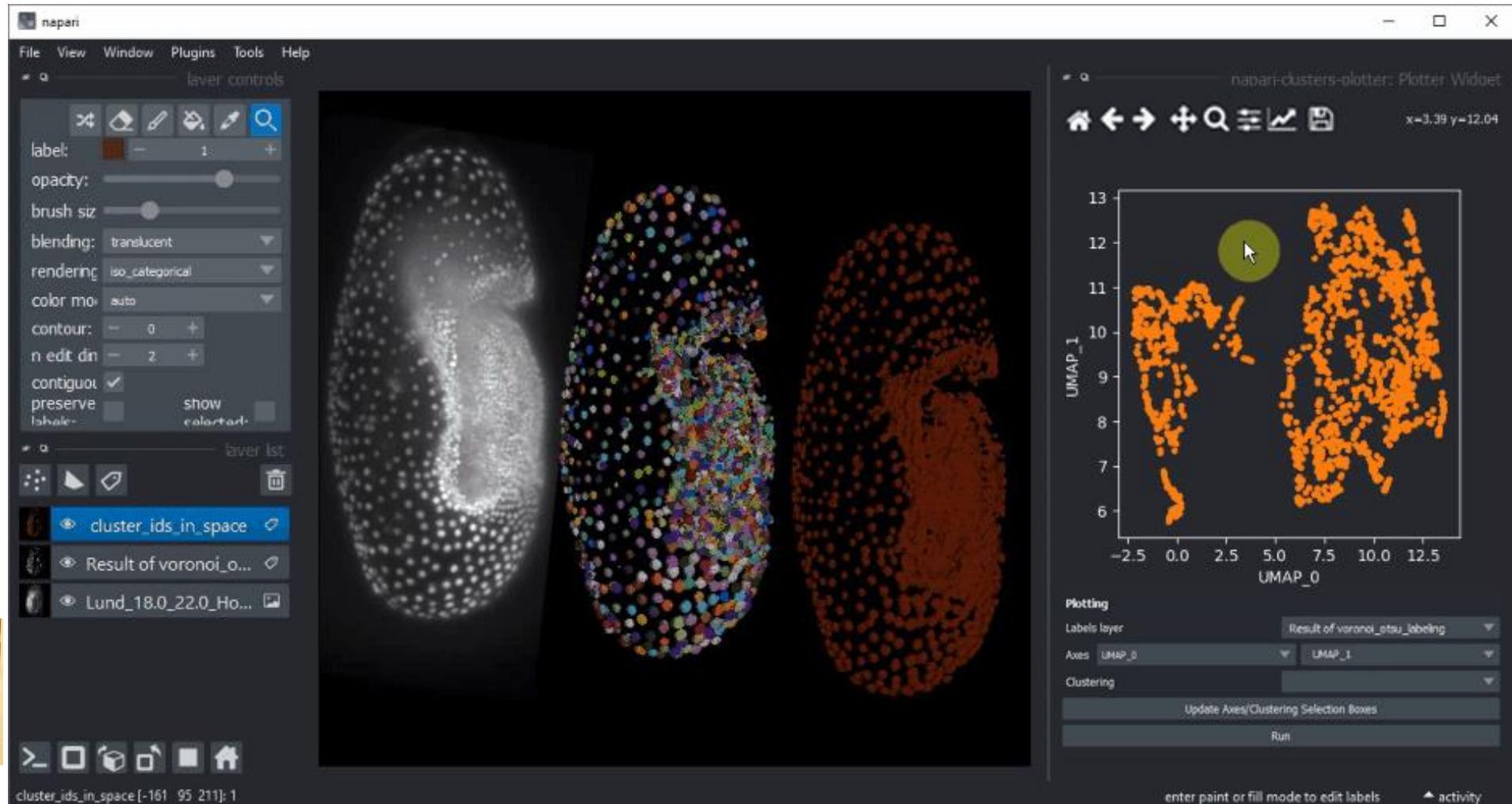


Image data source: Daniela Vorkel, Myers lab, MPI-CBG/CSBD

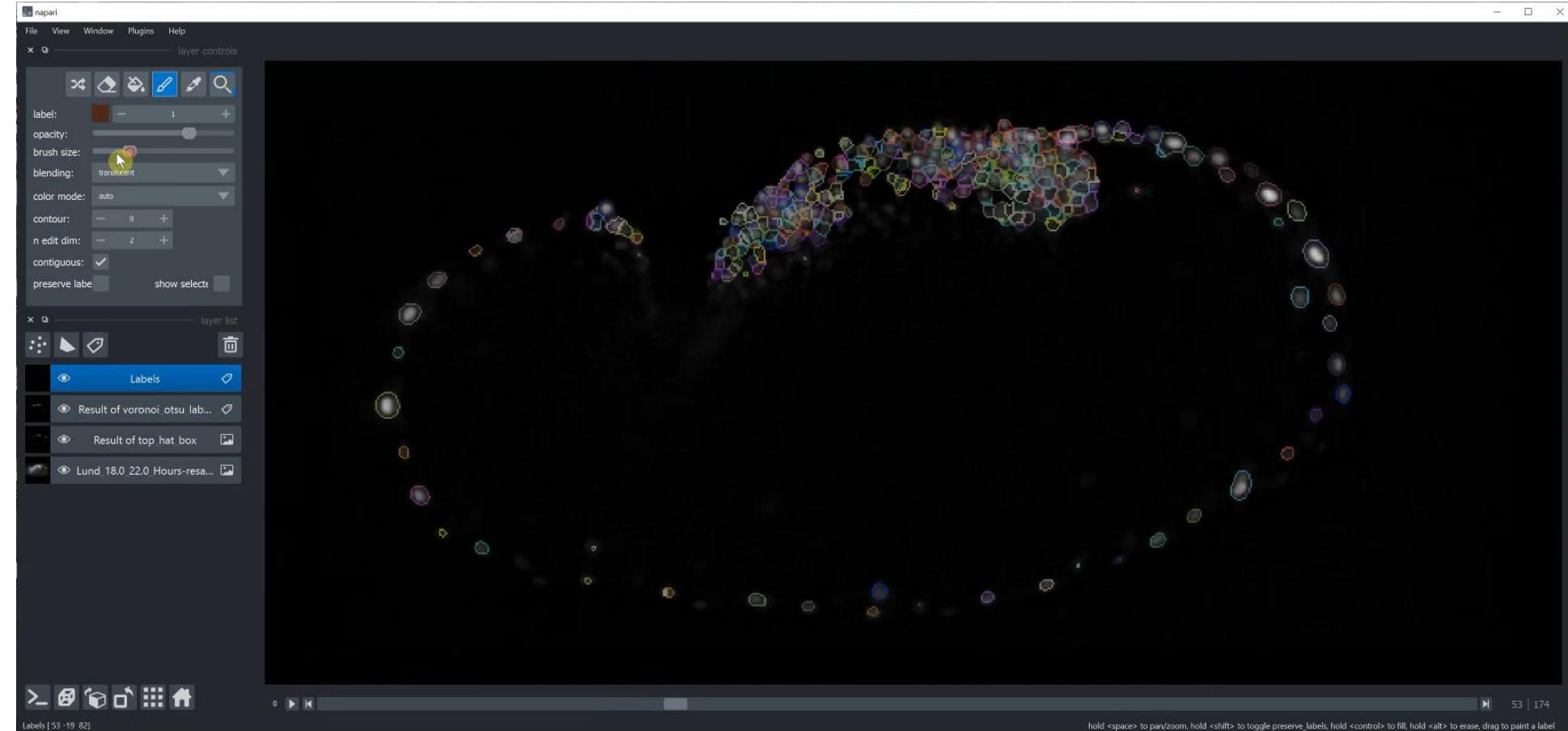


Laura Žigutytė @RyanSavill4 Marcelo Zoccoler
@zigutyte @RyanSavill4 @zoccolermarcelo

Napari-accelerated-pixel-and-object-classification

- Random Forest Classifiers based on
- scikit-learn and
- clesperanto

5x



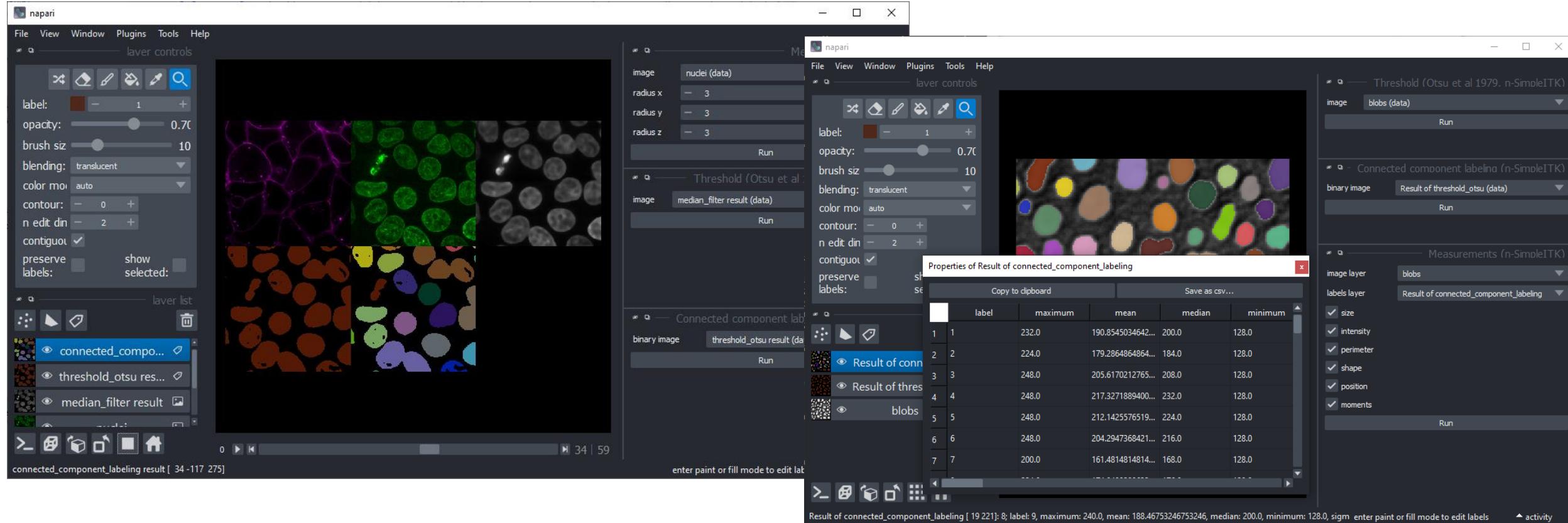
@haesleinhuepf <https://github.com/haesleinhuepf/napari-accelerated-pixel-and-object-classification>

Feature extraction

- napari-skimage-regionprops

Feature extraction

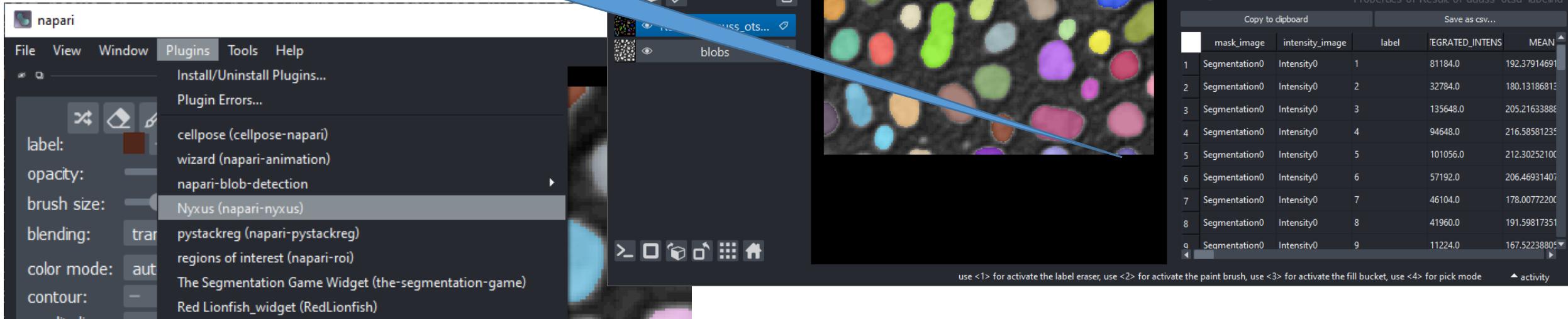
- napari-simpleitk-image-processing
- Recommended for 3D-measurements, based on the SimpleITK-project



Feature extraction

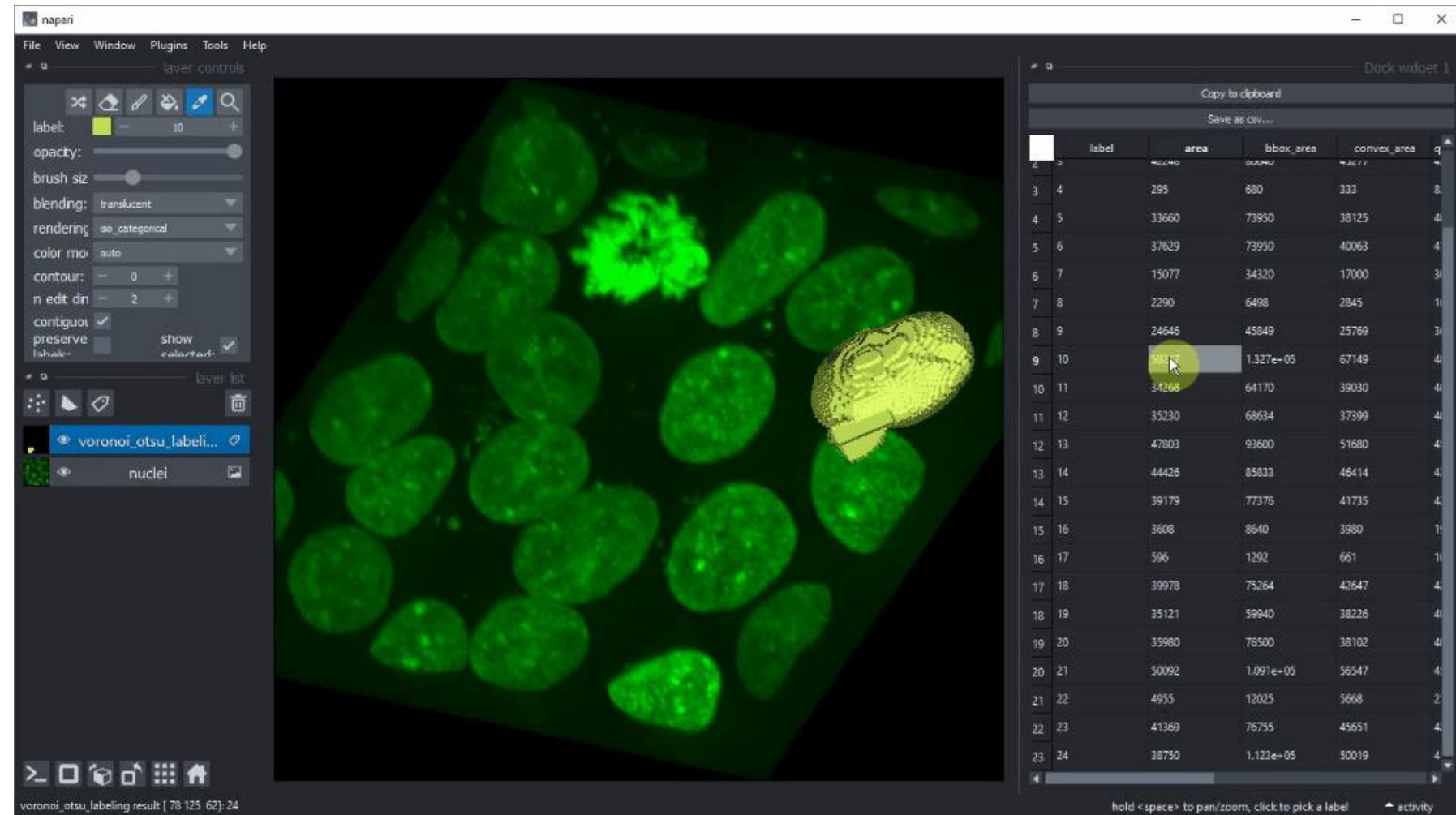
- napari-nyxus
- < 3 Month old plugin!

Measures about 400 features
known from scikit-image,
ImageJ, CellProfiler etc.



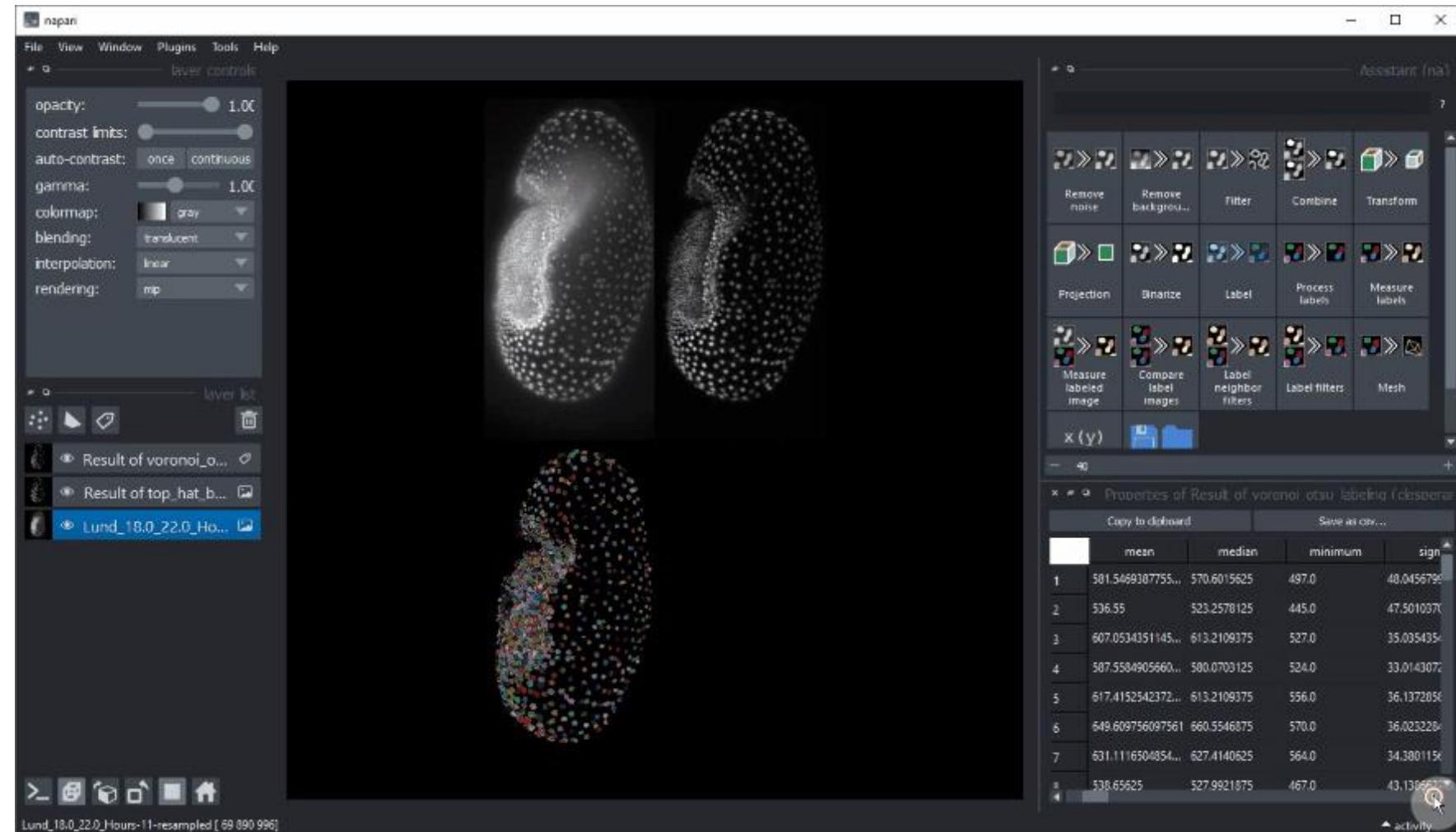
Data exploration

- Click on a cell to view the object the measurement belongs to



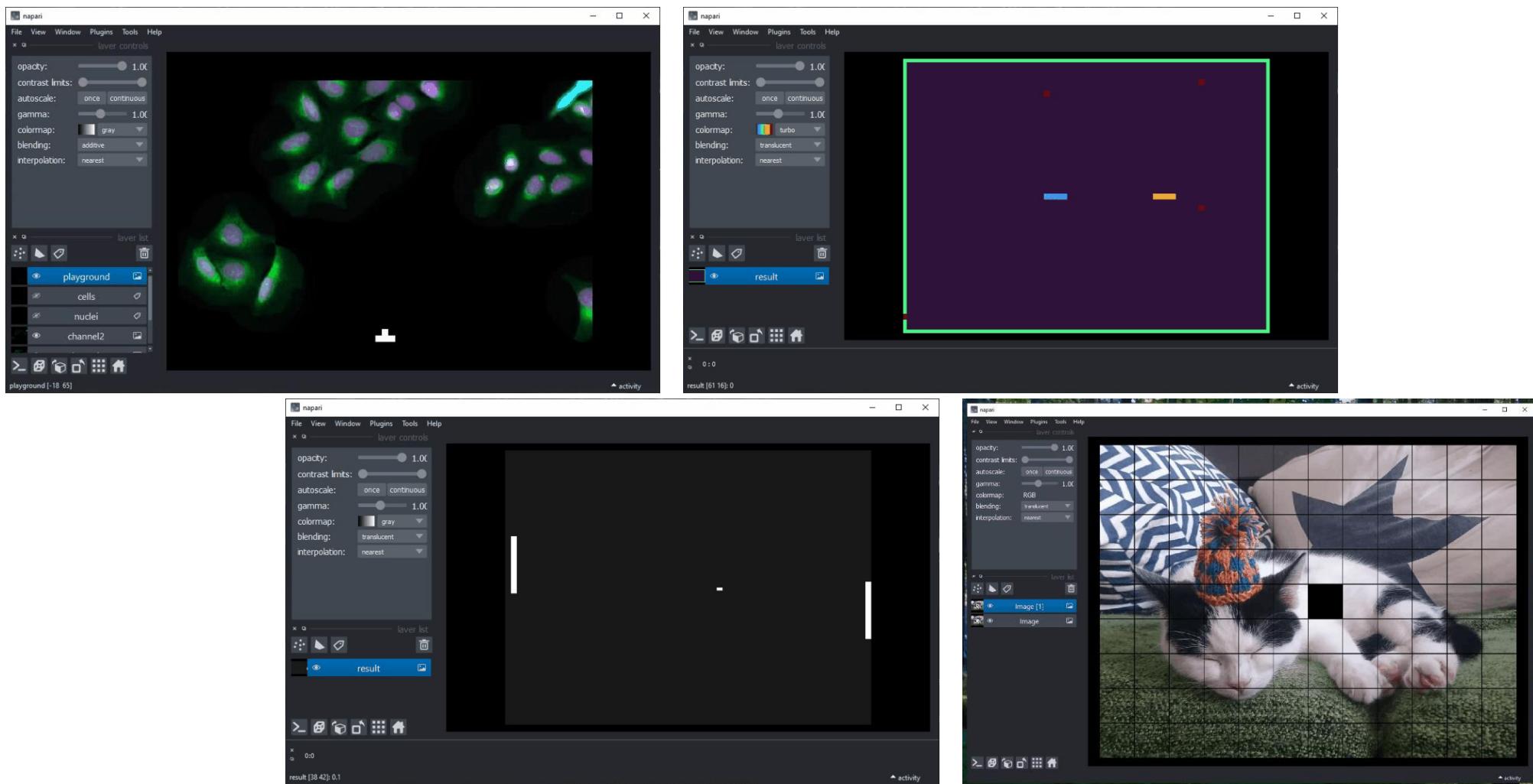
Data exploration

- Double-click on a column of measurements to view a parametric image

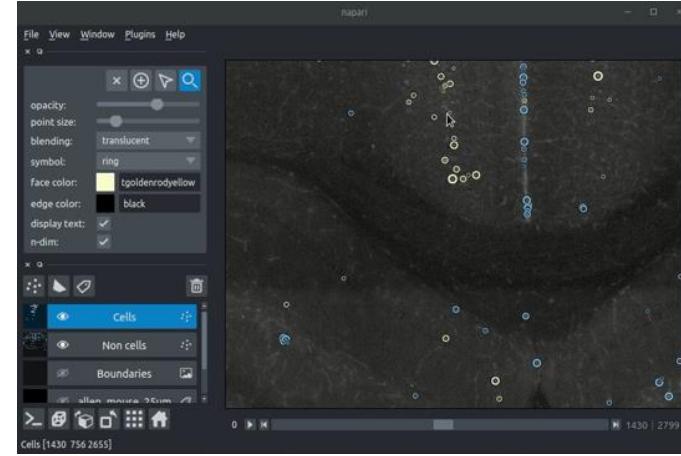


Natari

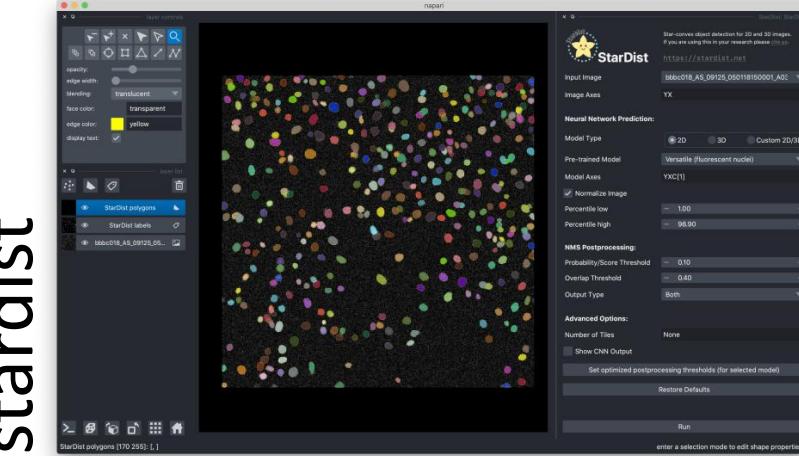
- Interactivity
- Event-handling
- State-handling



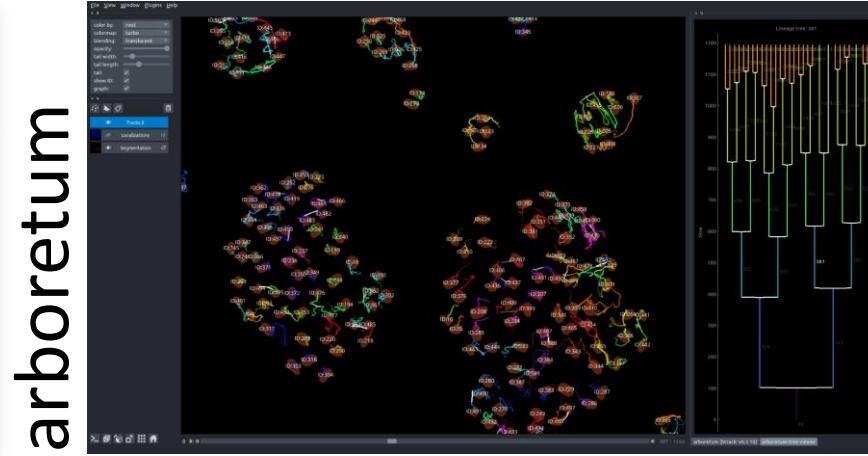
The era of napari plugins has just begun



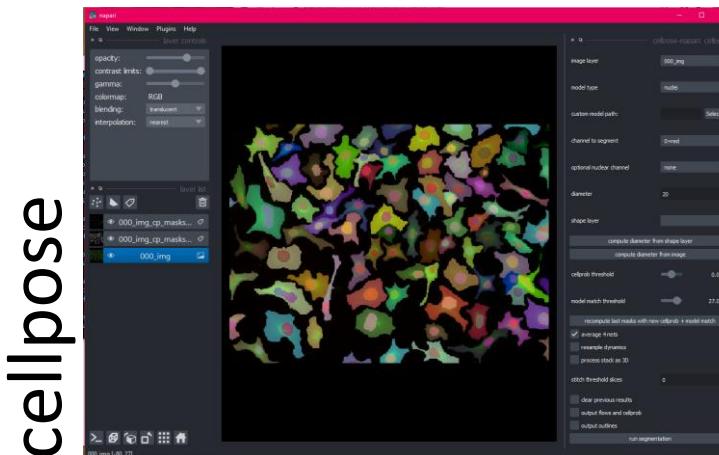
<https://github.com/brainglobe/napari-cellfinder>



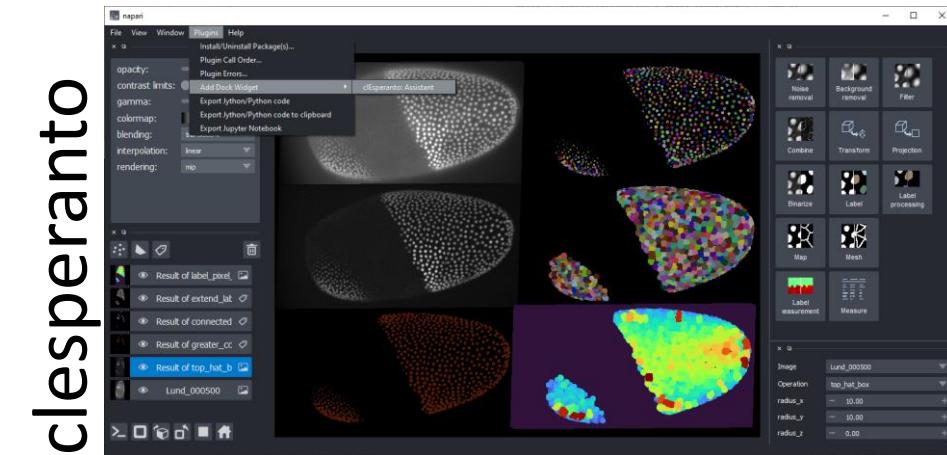
<https://github.com/stardist/stardist-napari>



<https://github.com/quantumjot/arboretum>



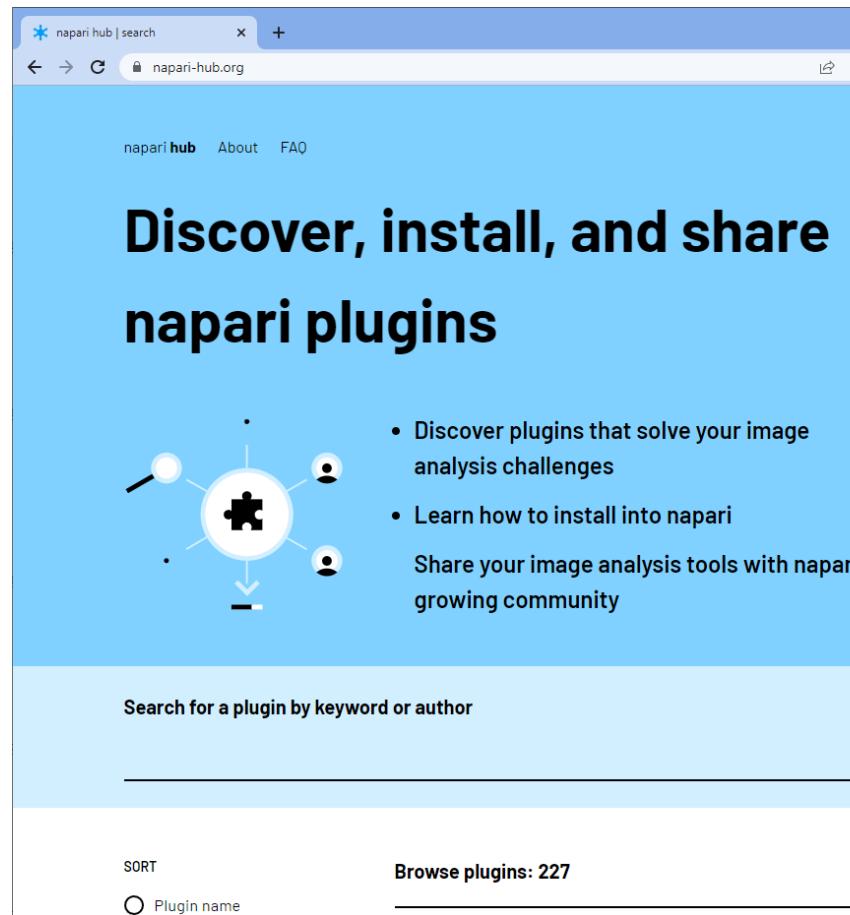
<https://cellpose-napari.readthedocs.io/en/latest/>



https://github.com/clEsperanto/napari_pyclesperanto_assistant

The Napari Hub

- Search engine for napari plugins



napari hub | search napari-hub.org

Discover, install, and share napari plugins

Discover plugins that solve your image analysis challenges

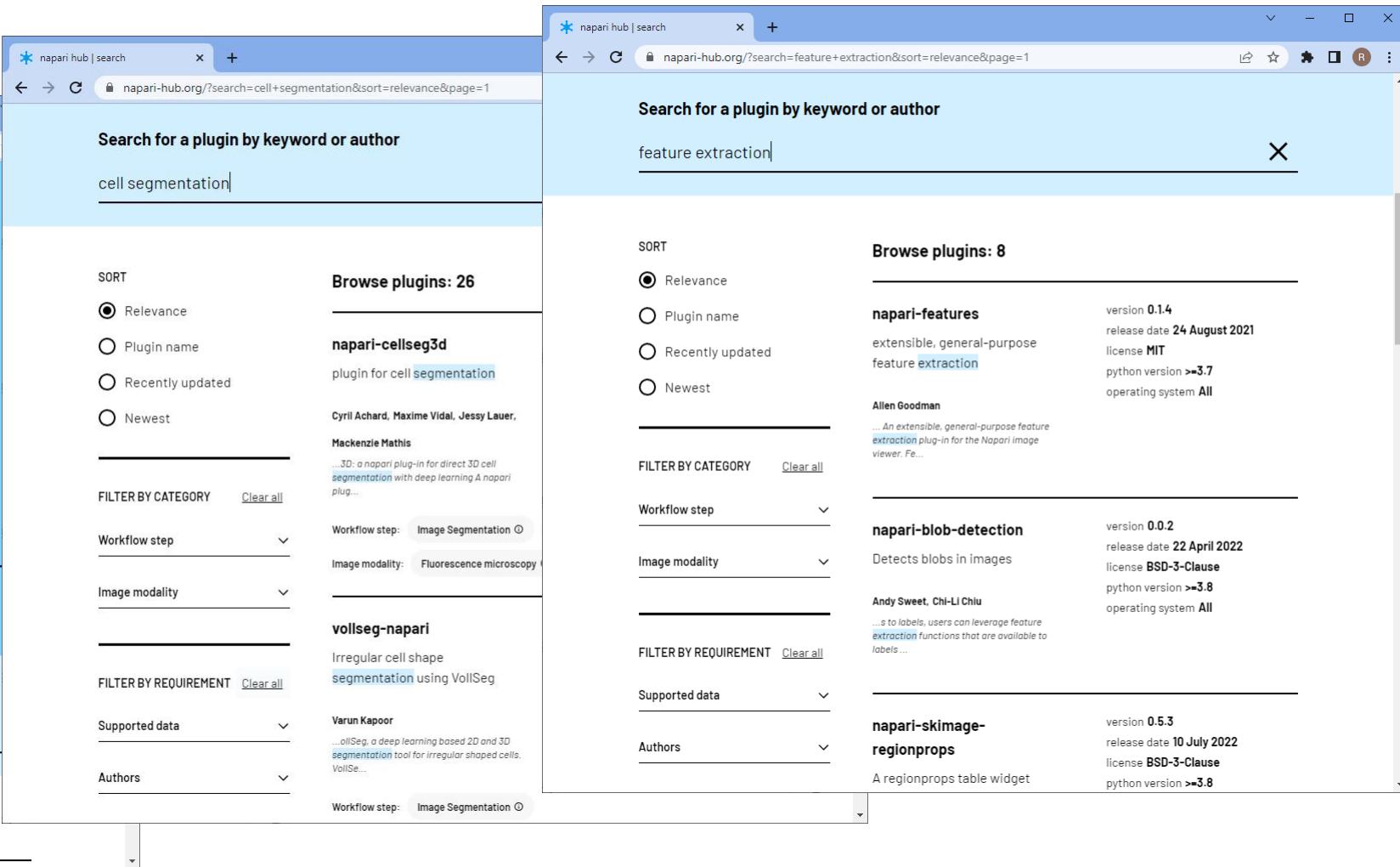
Learn how to install into napari

Share your image analysis tools with napari's growing community

Search for a plugin by keyword or author

SORT Plugin name

Browse plugins: 227



napari hub | search napari-hub.org/?search=cell+segmentation&sort=relevance&page=1

Search for a plugin by keyword or author

cell segmentation

SORT Relevance Plugin name Recently updated Newest

Browse plugins: 26

napari-cellseg3d
plugin for cell segmentation
Cyril Achard, Maxime Vidal, Jessy Lauer, Mackenzie Mathis
...3D: a napari plug-in for direct 3D cell segmentation with deep learning A napari plug...

vollseg-napari
Irregular cell shape segmentation using VollSeg
Varun Kapoor
...llSeg: a deep learning based 2D and 3D segmentation tool for irregular shaped cells. VollSe...

FILTER BY CATEGORY [Clear all](#)

Workflow step: Image Segmentation

Image modality: Fluorescence microscopy

FILTER BY REQUIREMENT [Clear all](#)

Supported data

Authors

Workflow step: Image Segmentation

napari-hub | search napari-hub.org/?search=feature+extraction&sort=relevance&page=1

Search for a plugin by keyword or author

feature extraction

SORT Relevance Plugin name Recently updated Newest

Browse plugins: 8

napari-features
version 0.1.4
release date 24 August 2021
license MIT
python version >=3.7
operating system All
extensible, general-purpose feature extraction

Allen Goodman
An extensible, general-purpose feature extraction plug-in for the Napari image viewer. Fe...

napari-blob-detection
version 0.0.2
release date 22 April 2022
license BSD-3-Clause
python version >=3.8
operating system All
Detects blobs in images

Andy Sweet, Chi-Li Chiu
...s to labels, users can leverage feature extraction functions that are available to labels ...

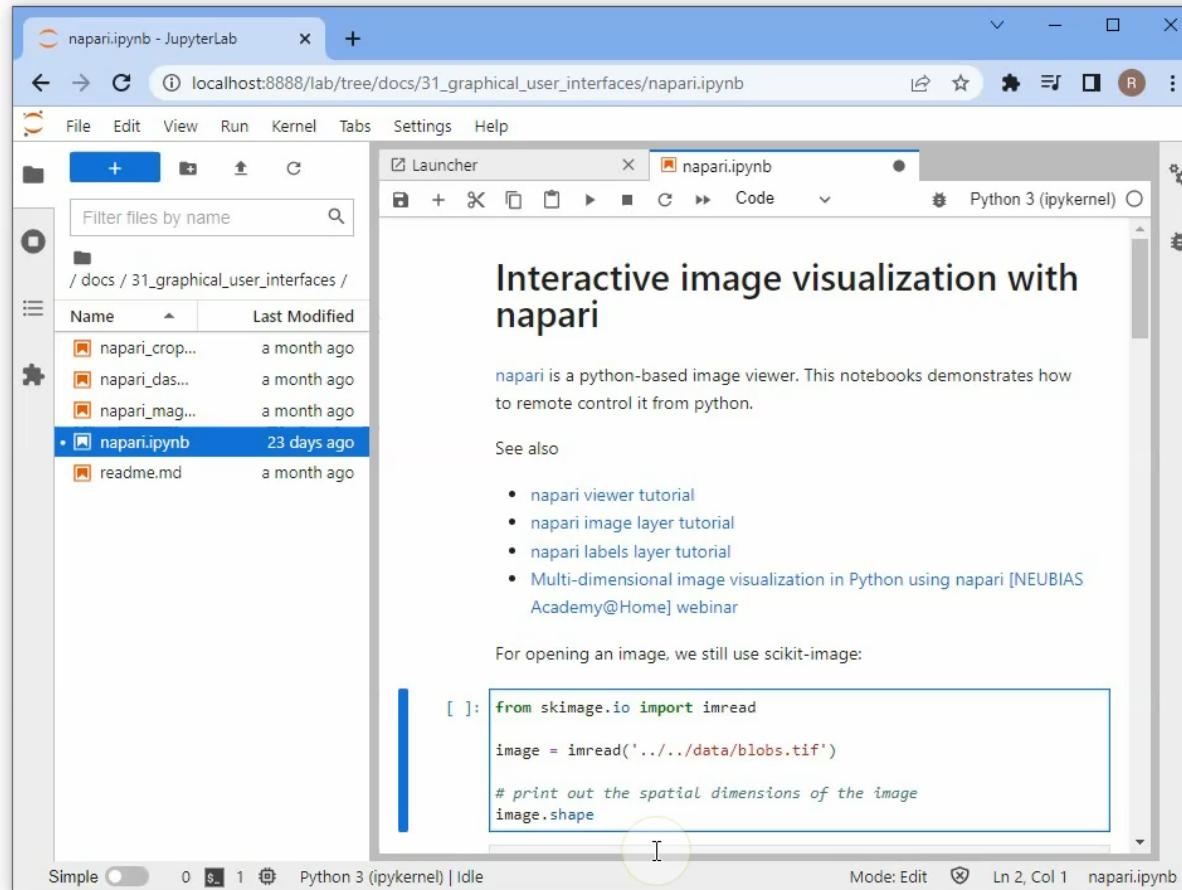
napari-skimage-regionprops
version 0.5.3
release date 10 July 2022
license BSD-3-Clause
python version >=3.8
A regionprops table widget

<https://www.napari-hub.org/>

- Nuclei segmentation + measurement + visualization

Using Napari from Python Code

- A great mix of interactivity and reproducibility



Scripting napari

- Initialization

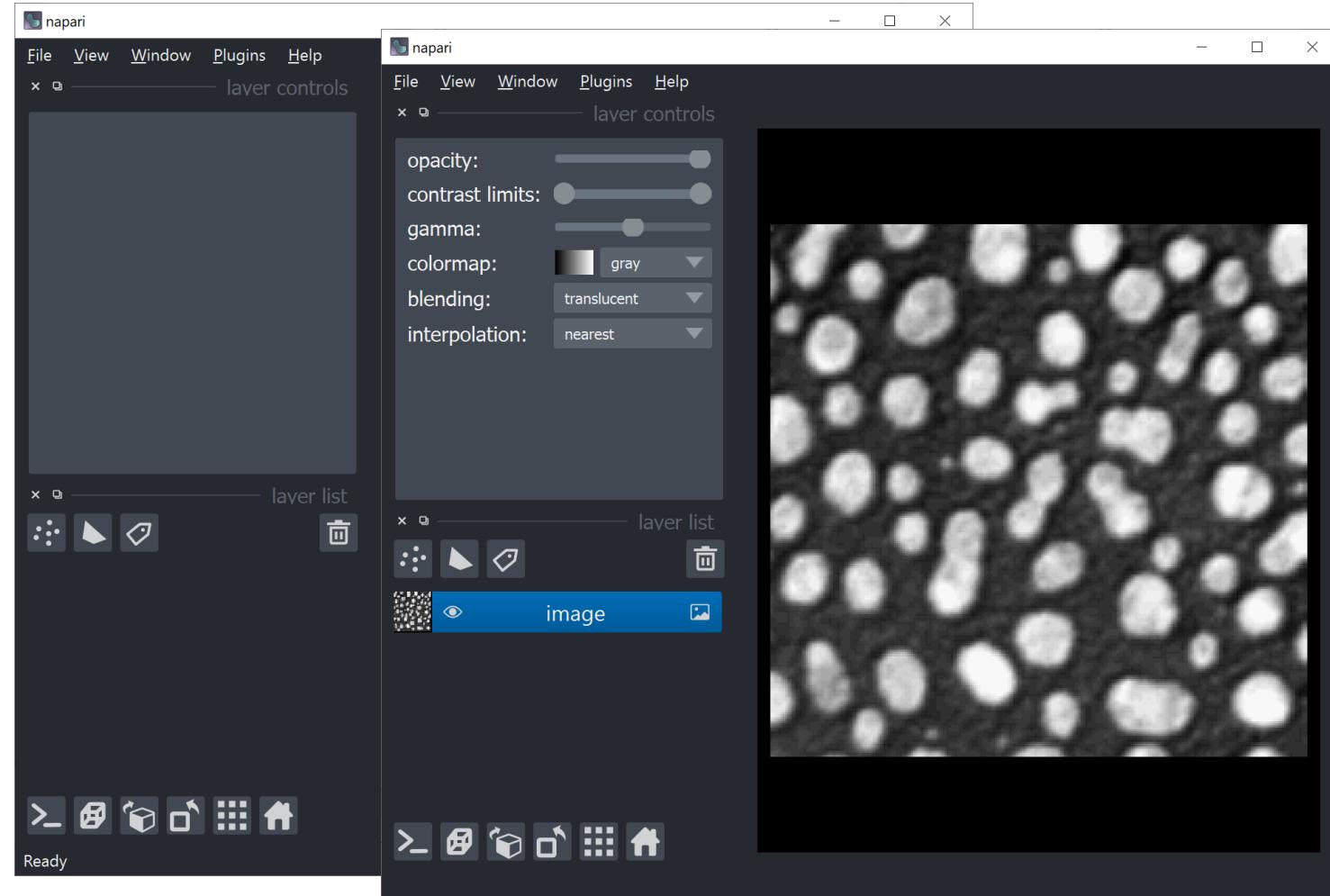
```
import napari
```

```
# Create an empty viewer
viewer = napari.Viewer()
```

```
# Start it
napari.run()
```

- Adding images

```
viewer.add_image(image)
```

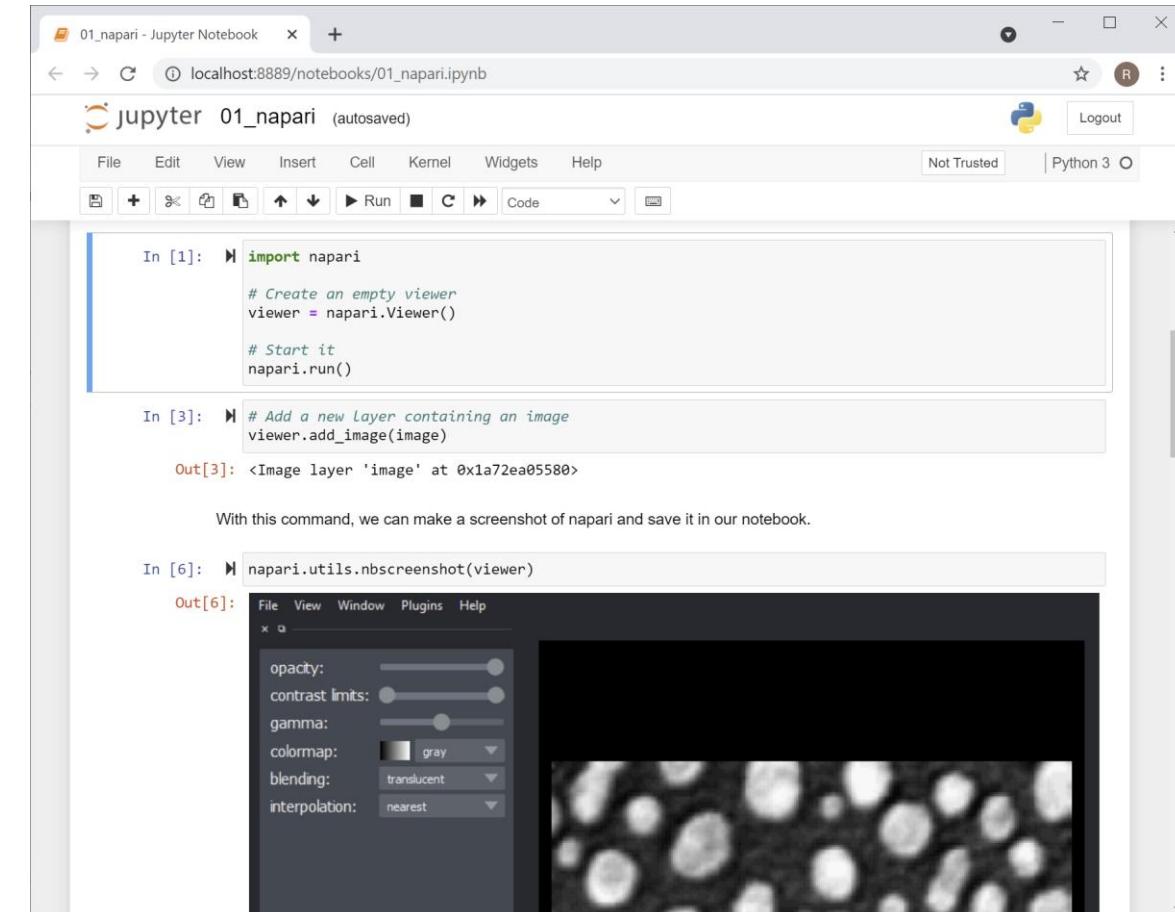


Scripting napari in notebooks

- Make screenshots from napari and put them in your jupyter notebook

```
napari.utils.nbscreenshot(viewer)
```

Place your viewer here



Working with layers

- Removing layers

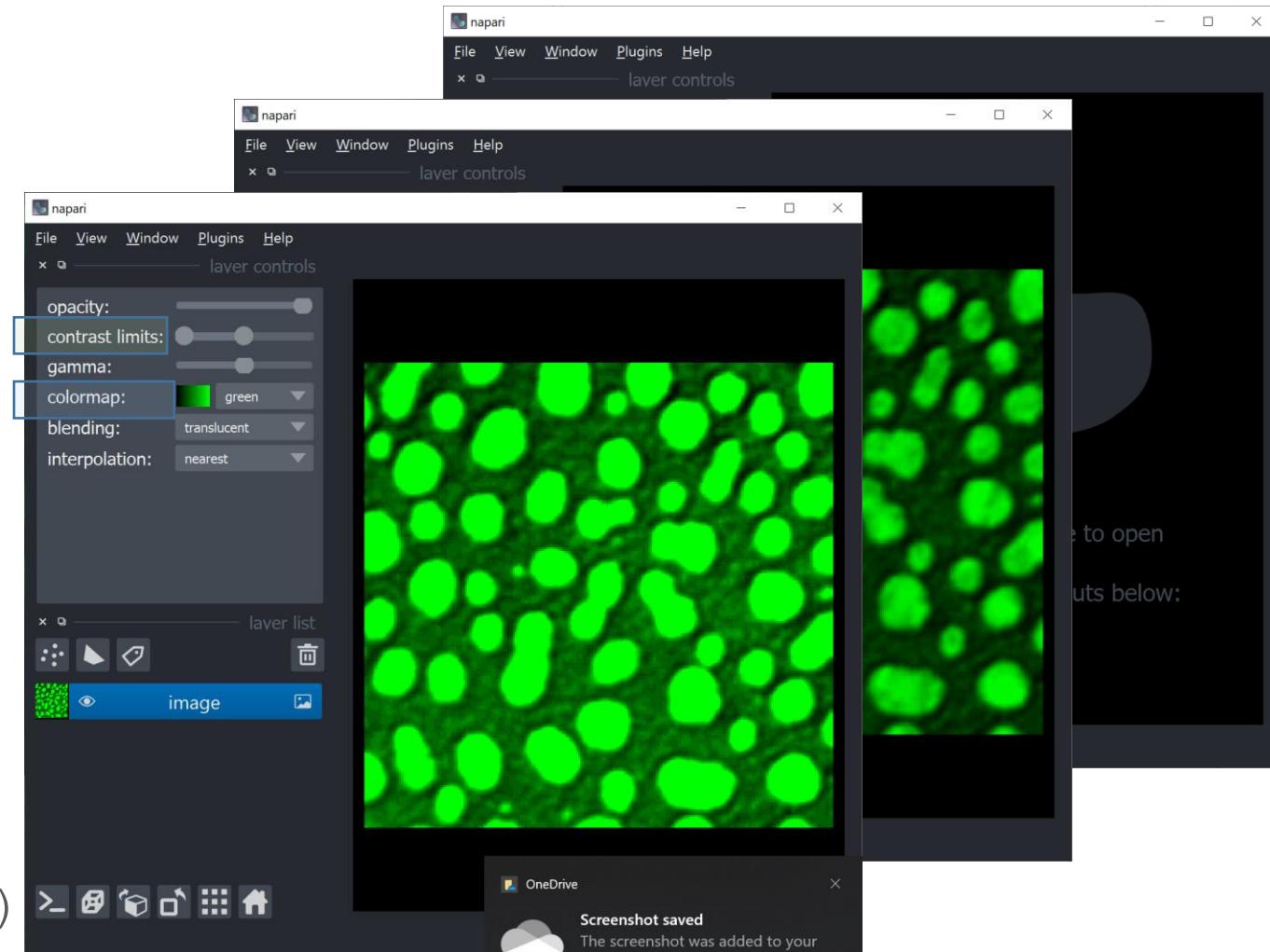
```
for l in viewer.layers:  
    viewer.layers.remove(l)
```

- Modify visualization while adding layers

```
viewer.add_image(image,  
                 colormap='green')
```

- Modify layers after adding

```
layer = viewer.add_image(image)  
layer.colormap = 'green'  
layer.contrast_limits = (0, 128)
```



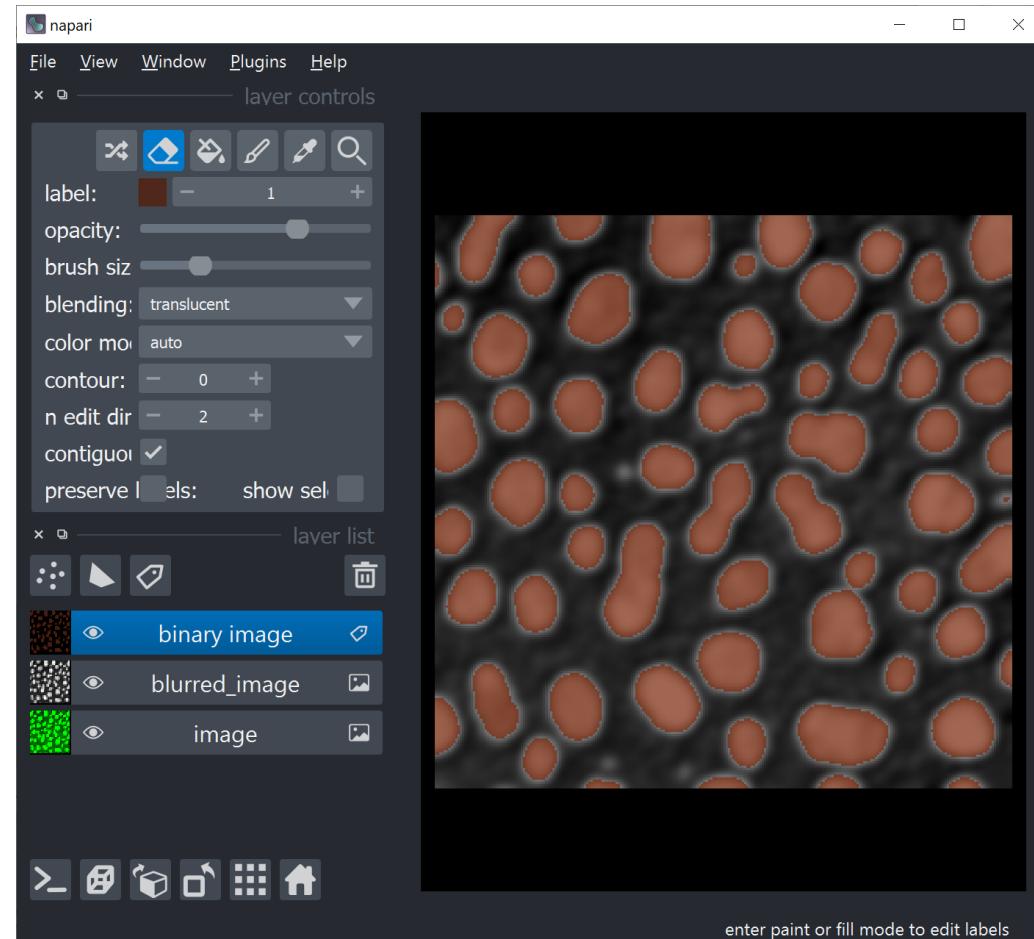
Visualizing image segmentation

- Binary images and **label** images visualized as label layers

```
from skimage.filters import threshold_otsu
threshold = threshold_otsu(blurred_image)
binary_image = blurred_image > threshold

# Add a new labels layer containing an image
viewer.add_labels(binary_image,
                  name="binary image")
```

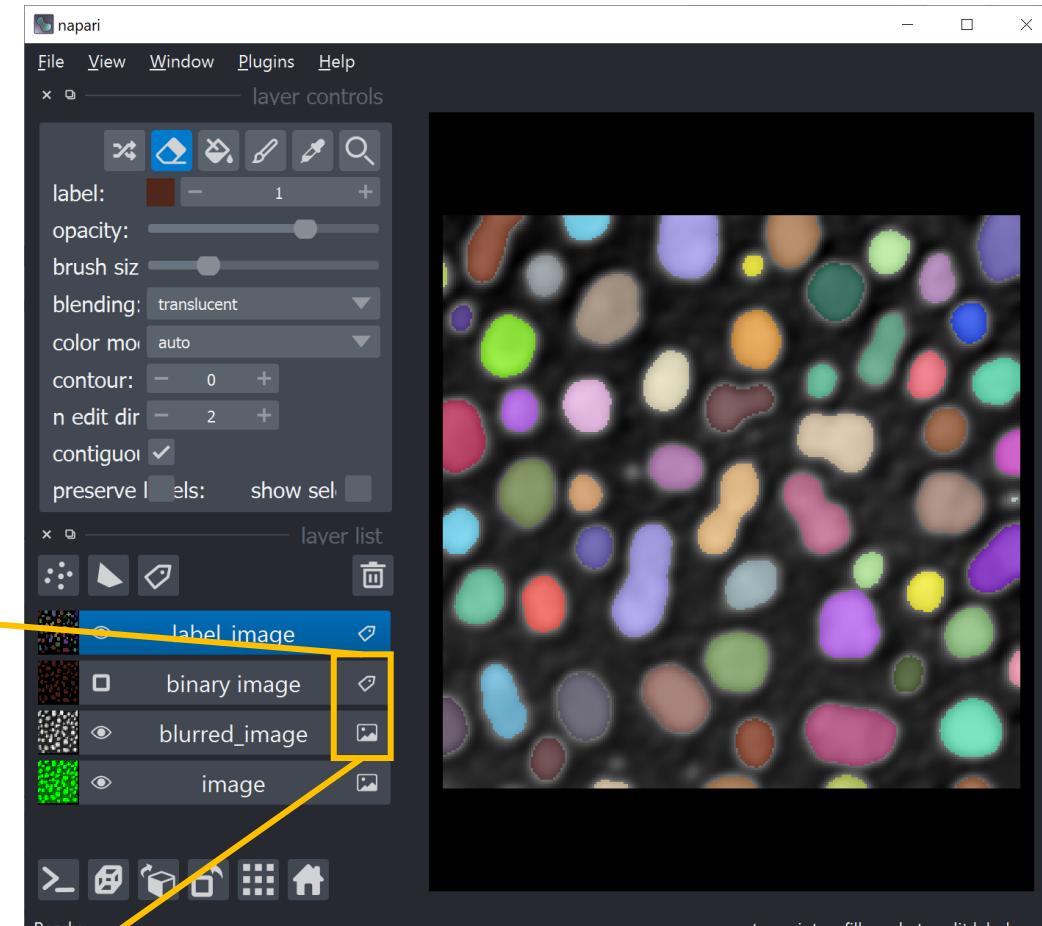
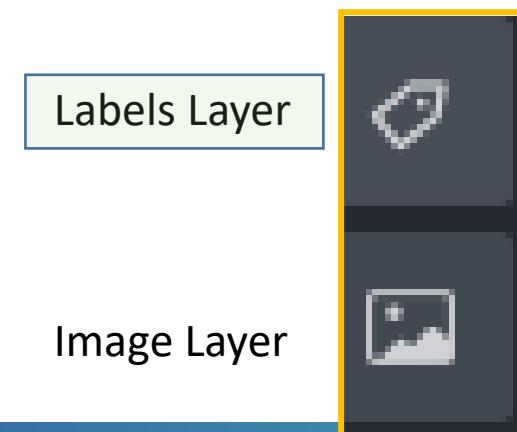
Name your layers to keep track
of what they contain



Visualizing image segmentation

- Binary images and label images visualized as label layers

```
from skimage.measure import label  
  
label_image = label(binary_image)
```



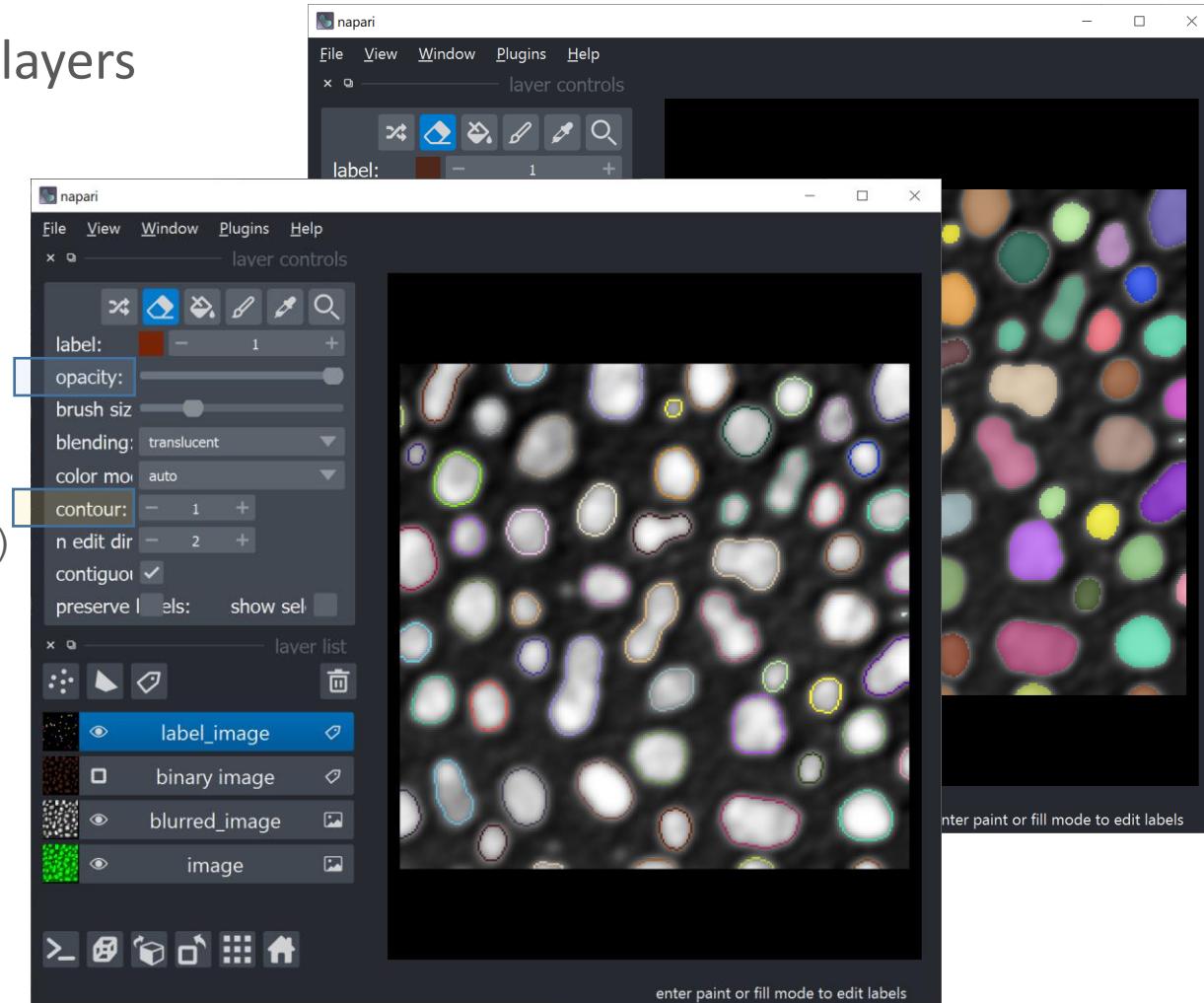
Visualizing image segmentation

- Binary images and label images visualized as label layers

```
from skimage.measure import label  
  
label_image = label(binary_image)  
  
# add labels to viewer  
  
label_layer = viewer.add_labels(label_image)
```

- Visualize contours instead of the overlay

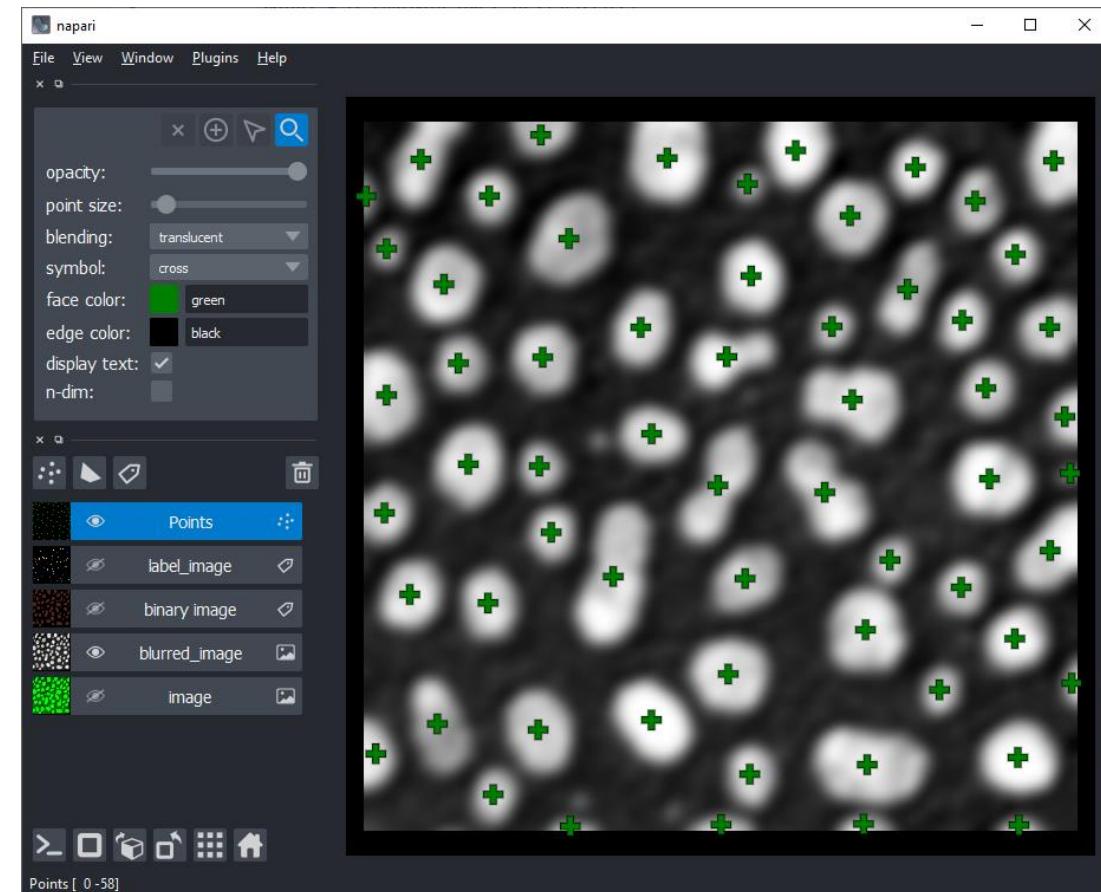
```
label_layer.contour = 1  
  
label_layer.opacity = 1
```



Points layers

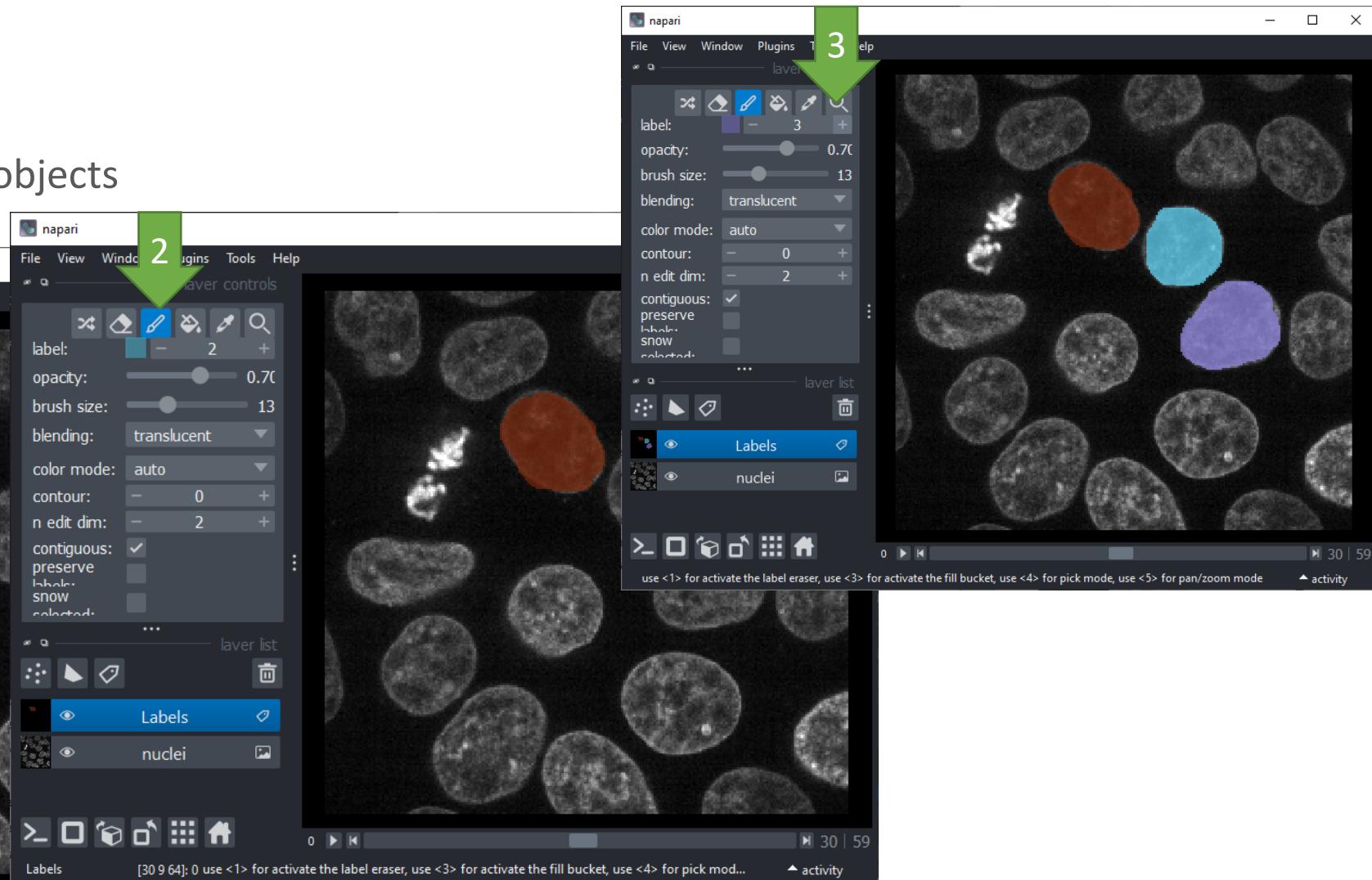
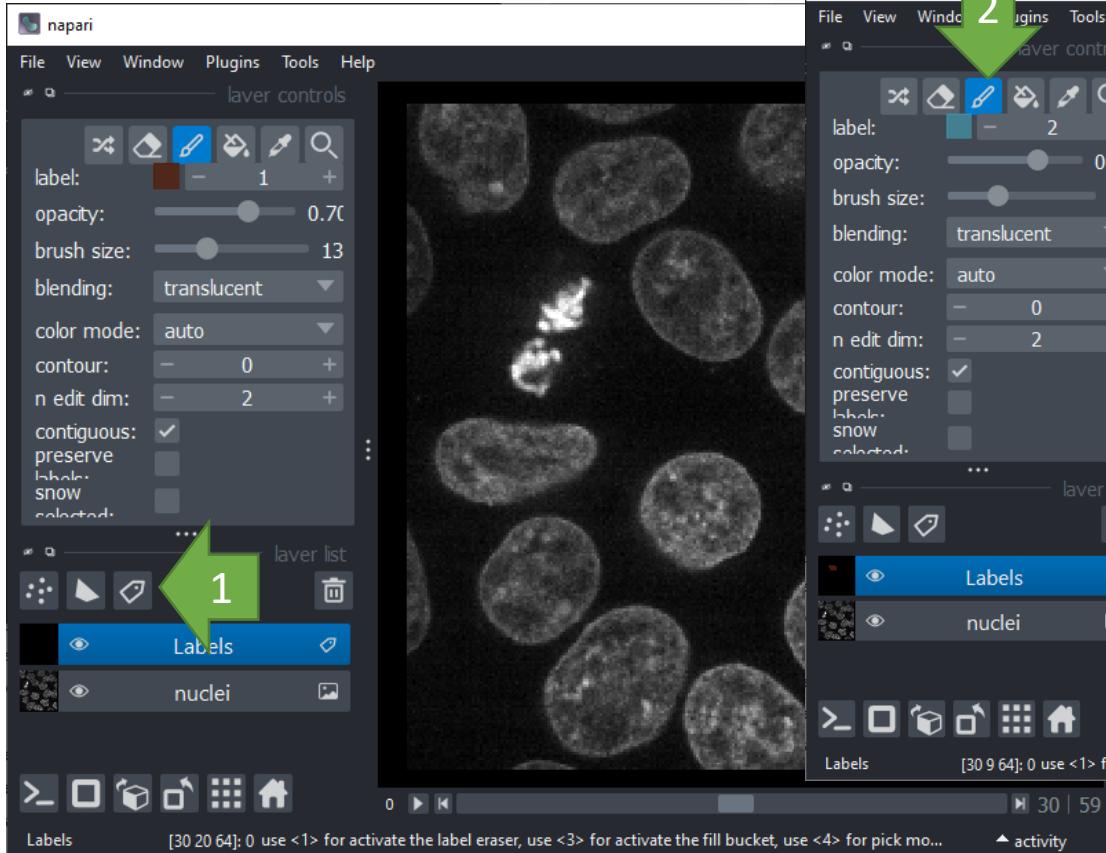
- There is also other layer types
 - Shapes
 - Points
 - Surfaces
 - Tracks
 - Vectors

```
from skimage.measure import regionprops
statistics = regionprops(label_image)
points = [s.centroid for s in statistics]
# add points to viewer
label_layer = viewer.add_points(points, face_color='green', symbol='cross', size=5)
```



Manual annotation

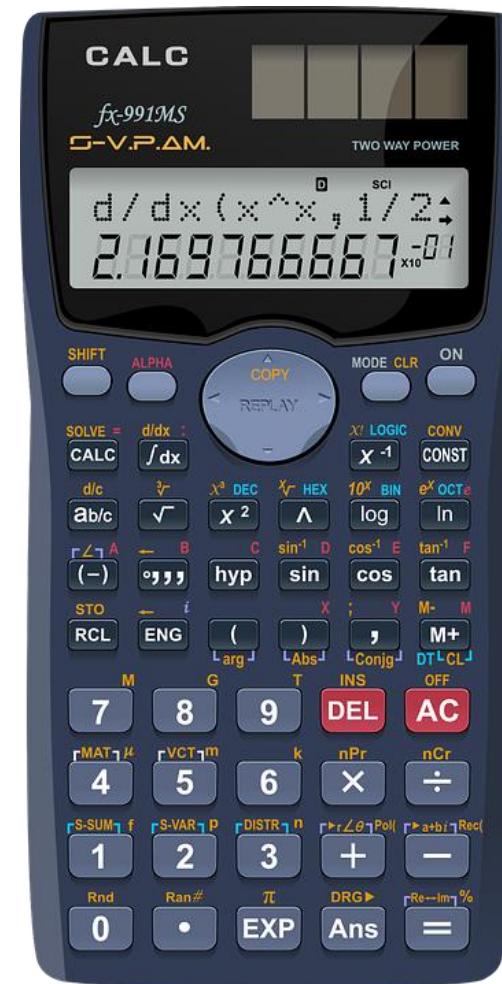
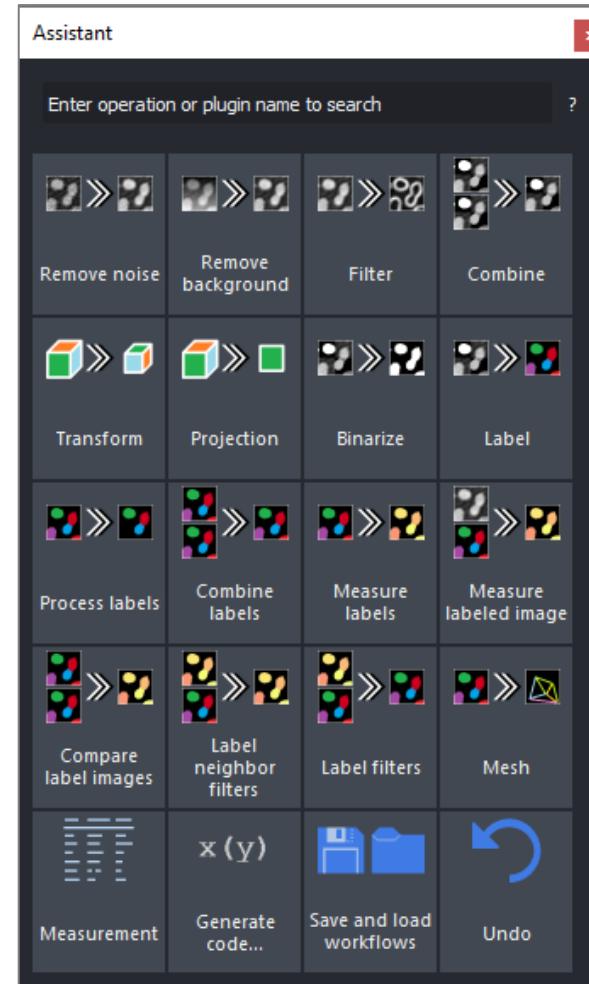
1. Create a labels layer
2. Annotate first object
3. Increase label, annotate more objects



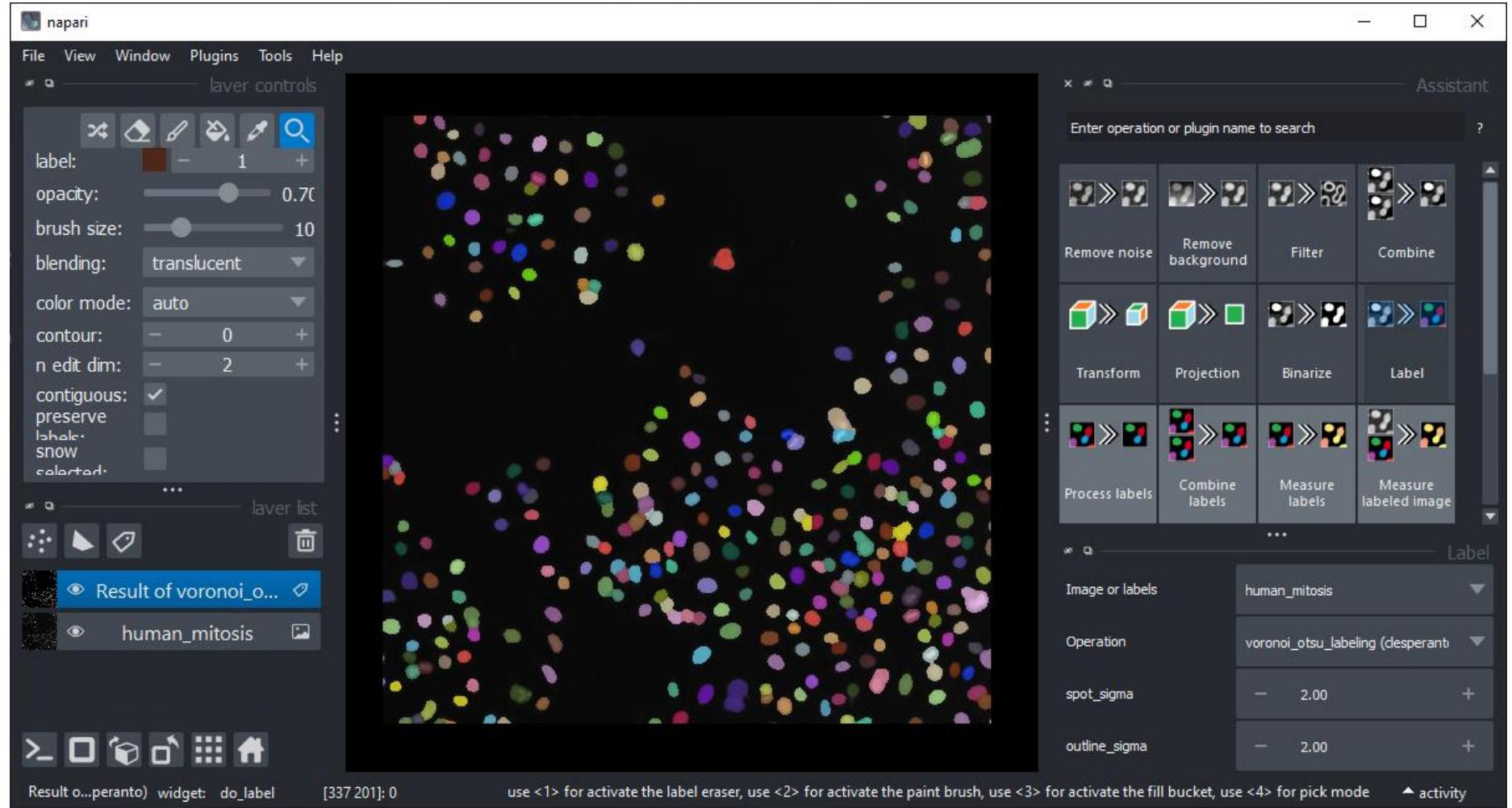
Demo: Drawing manual annotations in Napari

The Napari Assistant

- A pocket-calculator-like interface to build image analysis workflows



The Napari Assistant



Viewer
controls
(Napari core)

Image
Analysis
(Napari Assistant)

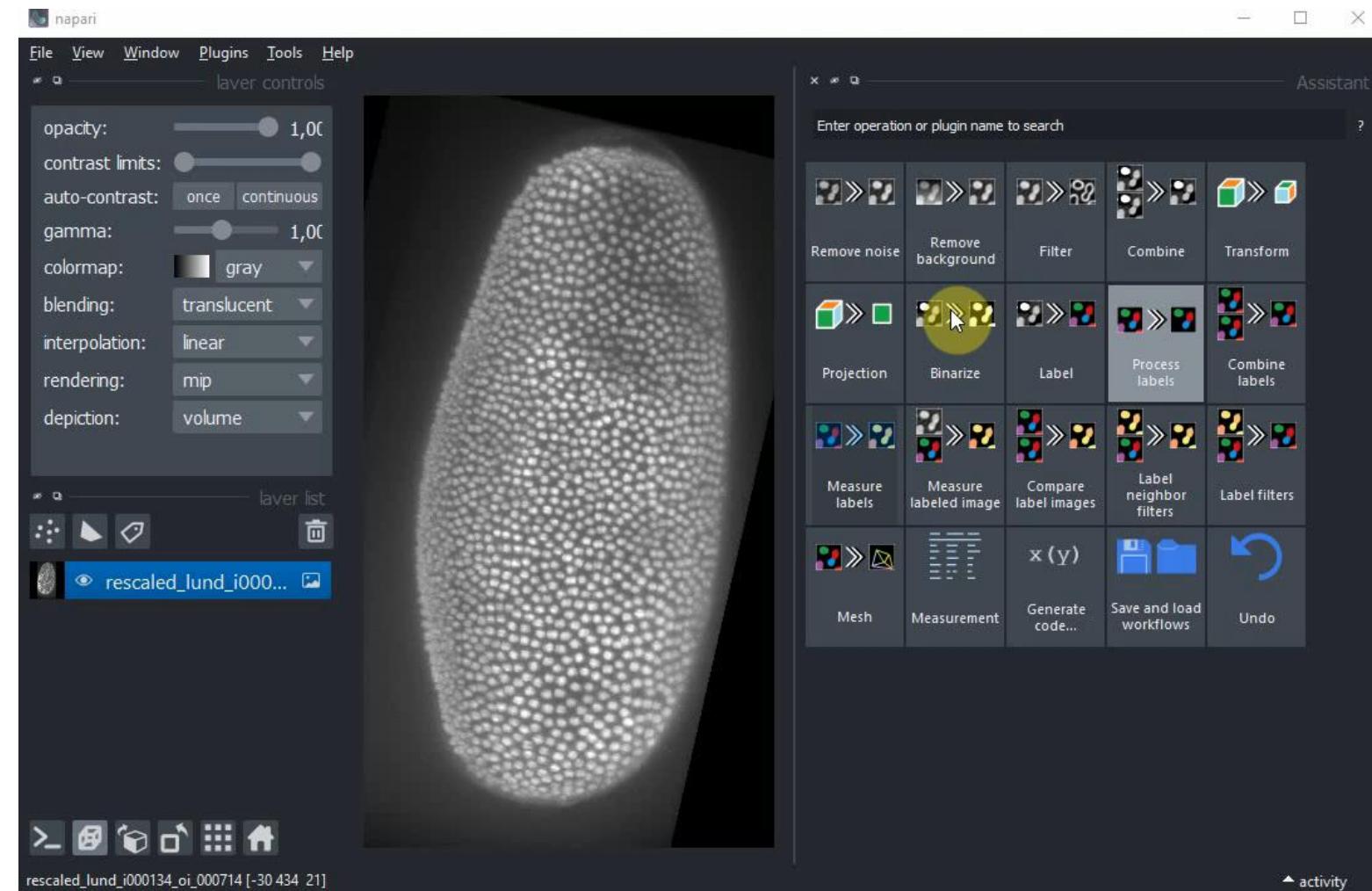
The Napari Assistant

- Classical image processing operations + advanced tools
- Saving&loading supported
- Undo [redo]
- Hints for next steps
- ...

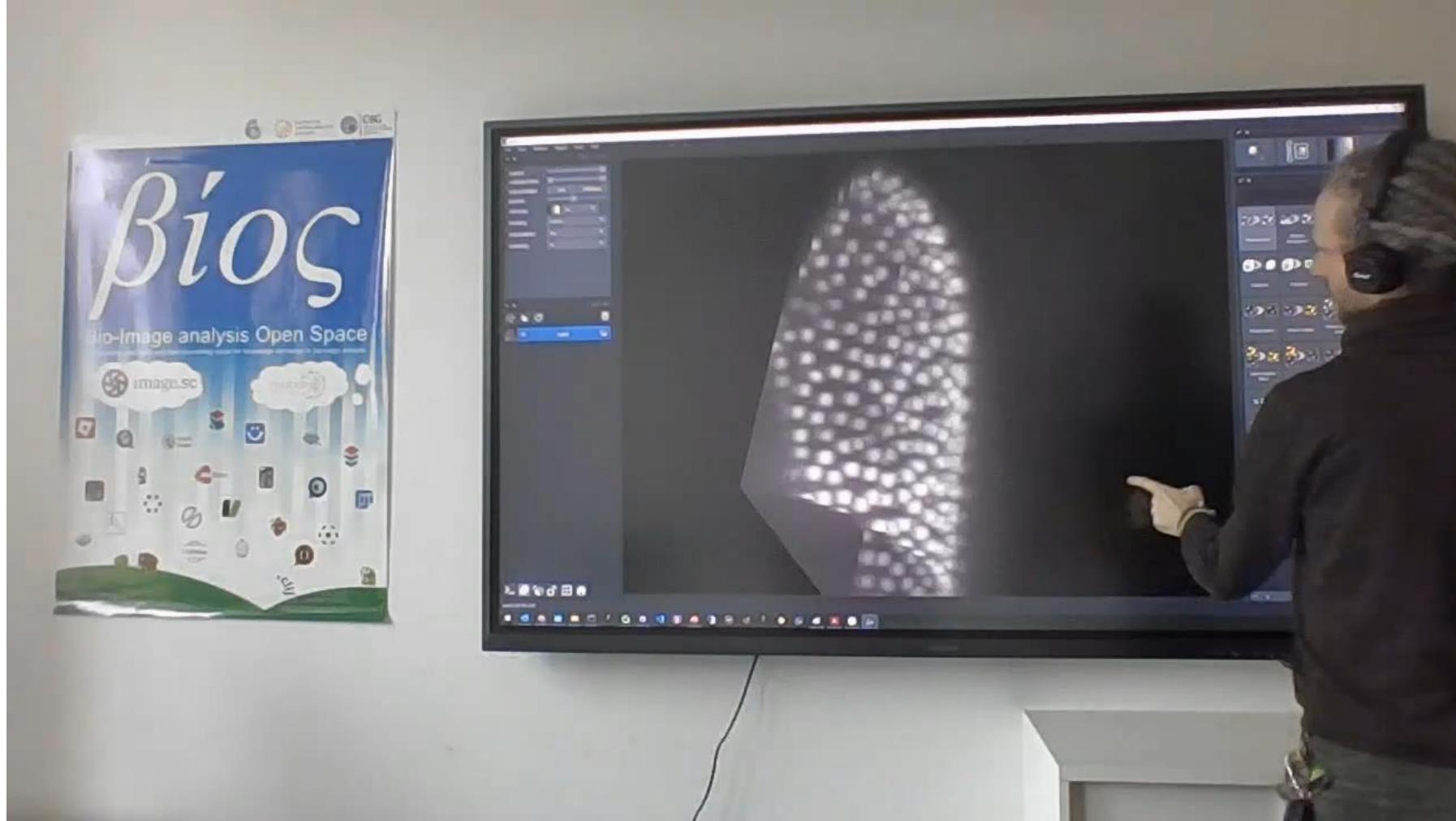
Big thanks to:



Ryan Savill
(now at MPI-CBG) @RyanSavill4

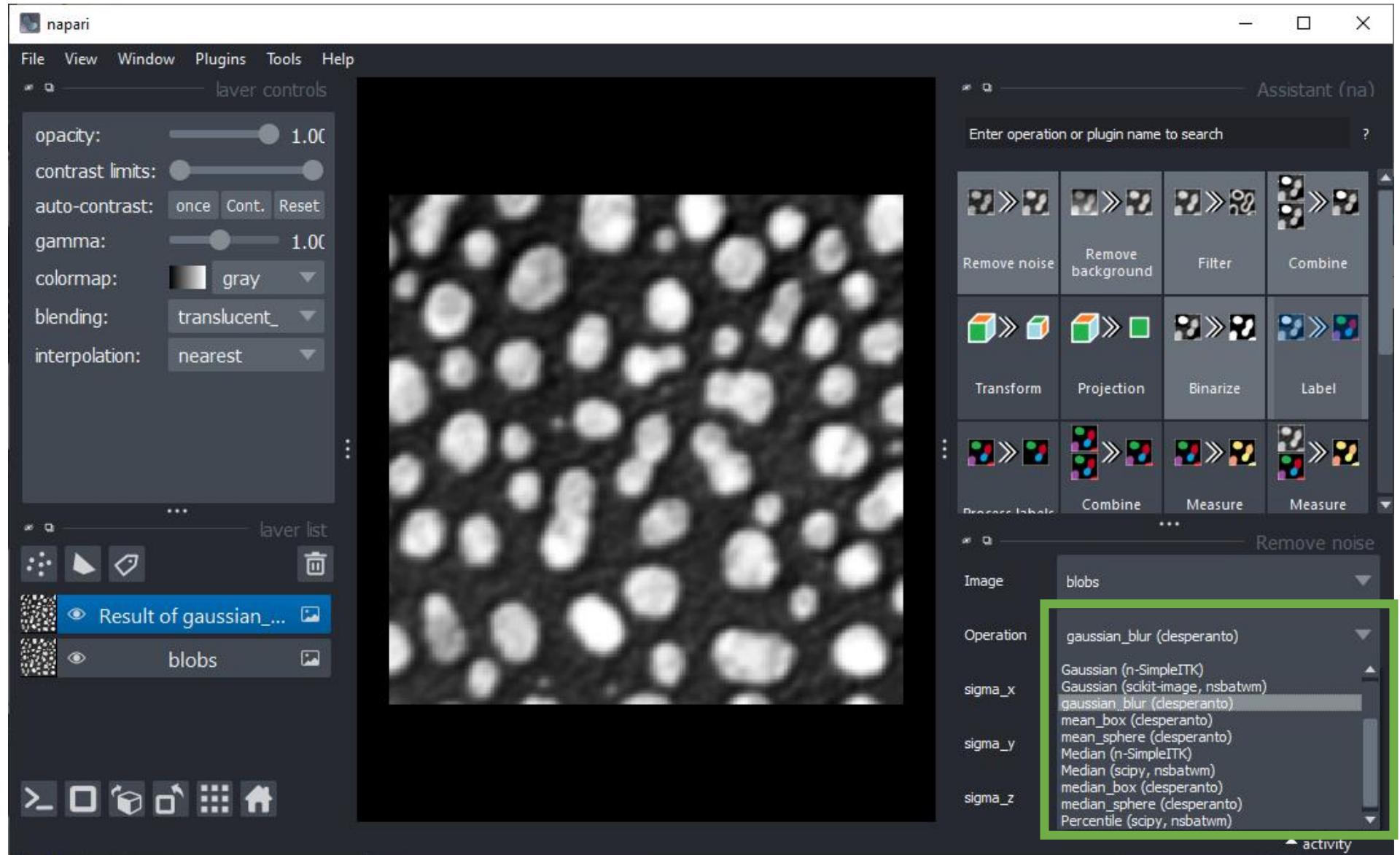


3D image visualization + interaction



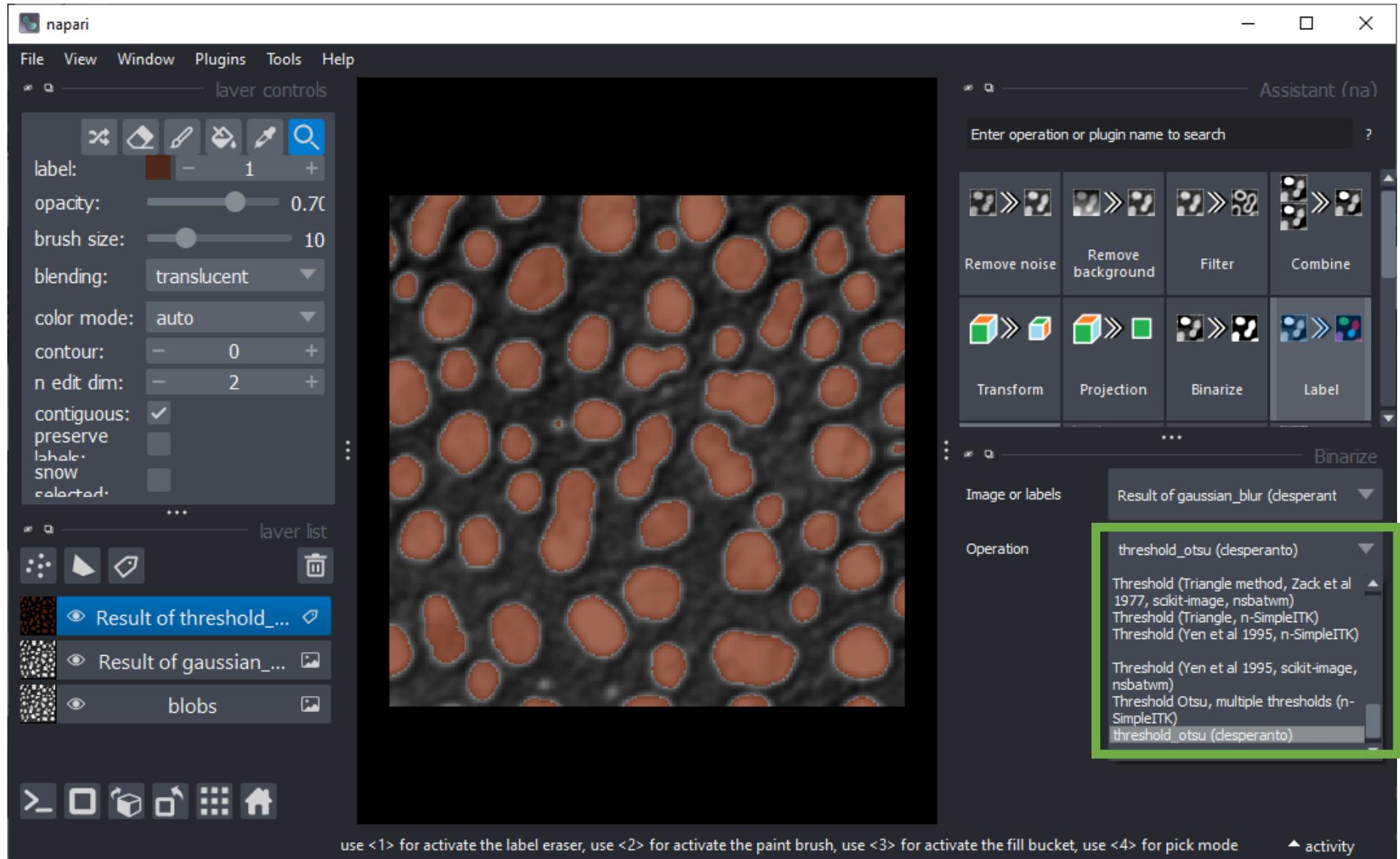
Workflow building

- Try different algorithms, e.g. for removing noise
- Find them in the pulldown



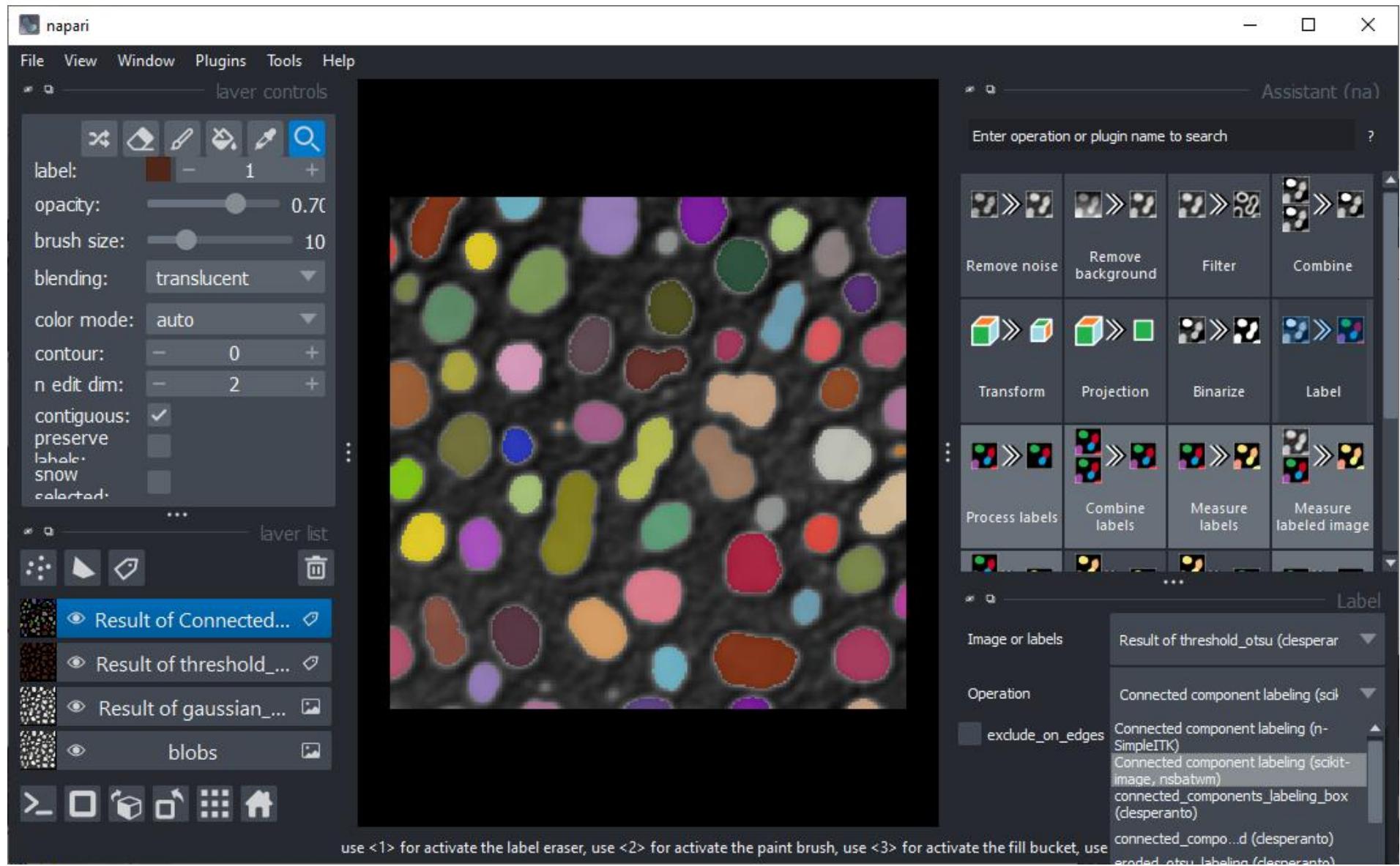
Workflow building

- Try different binarization algorithms



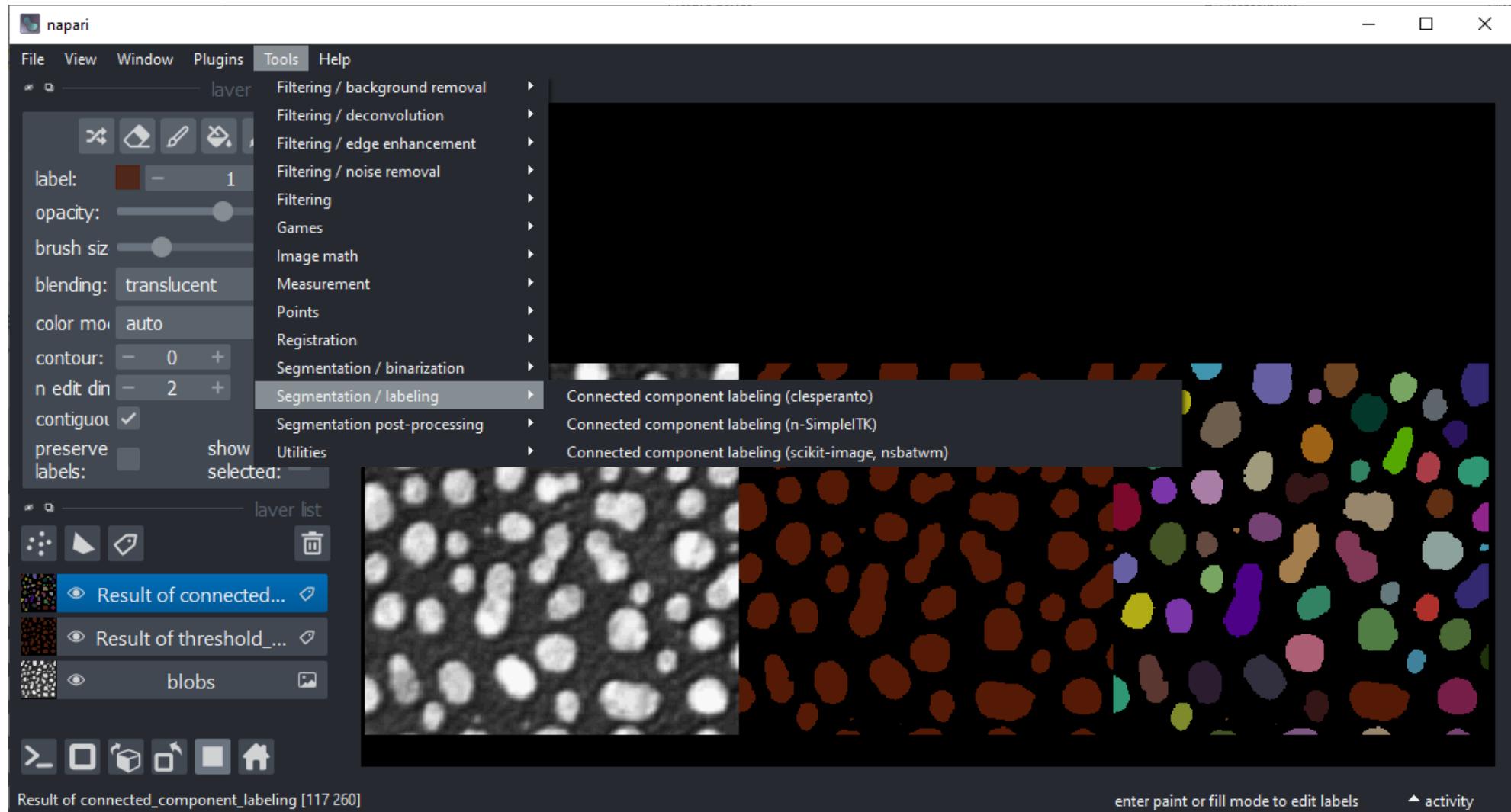
Workflow building

- Try different labeling algorithms



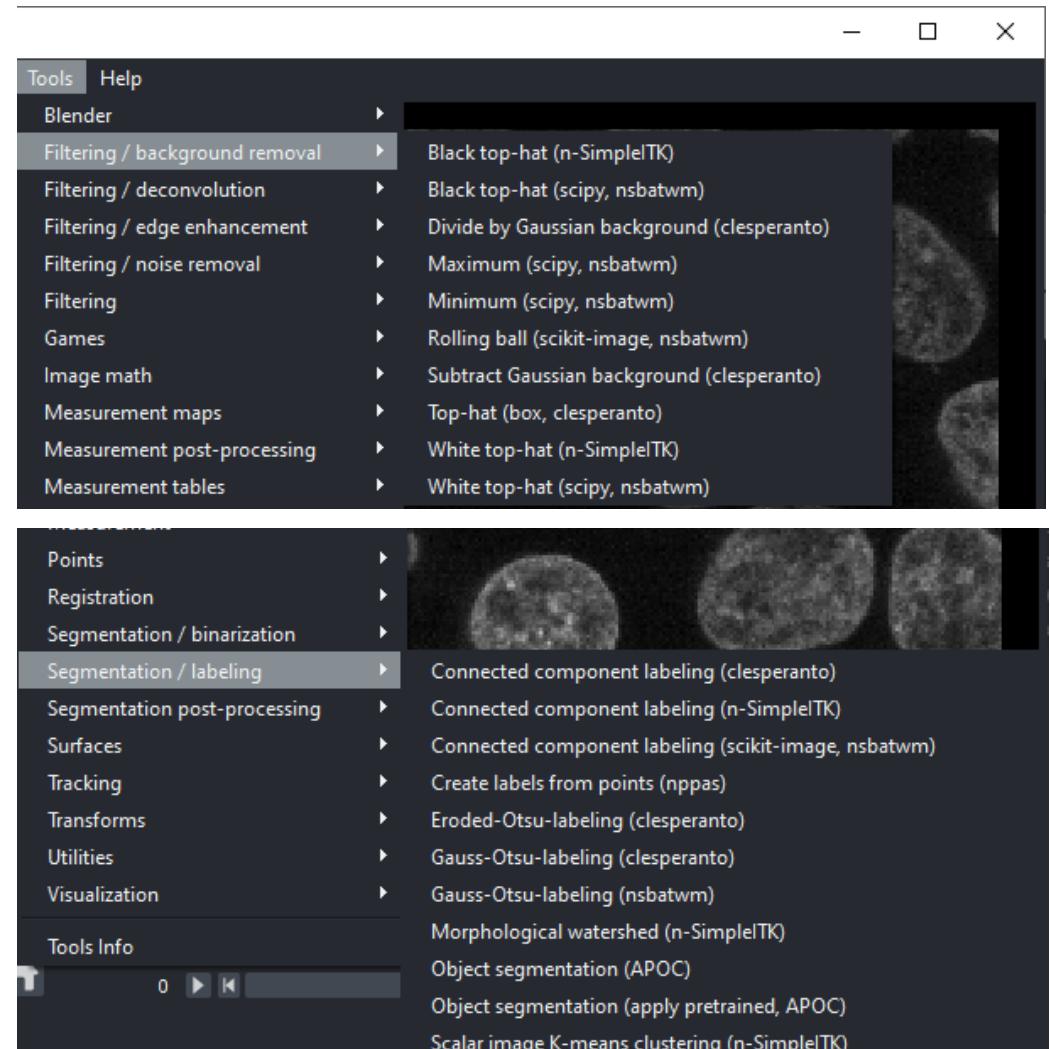
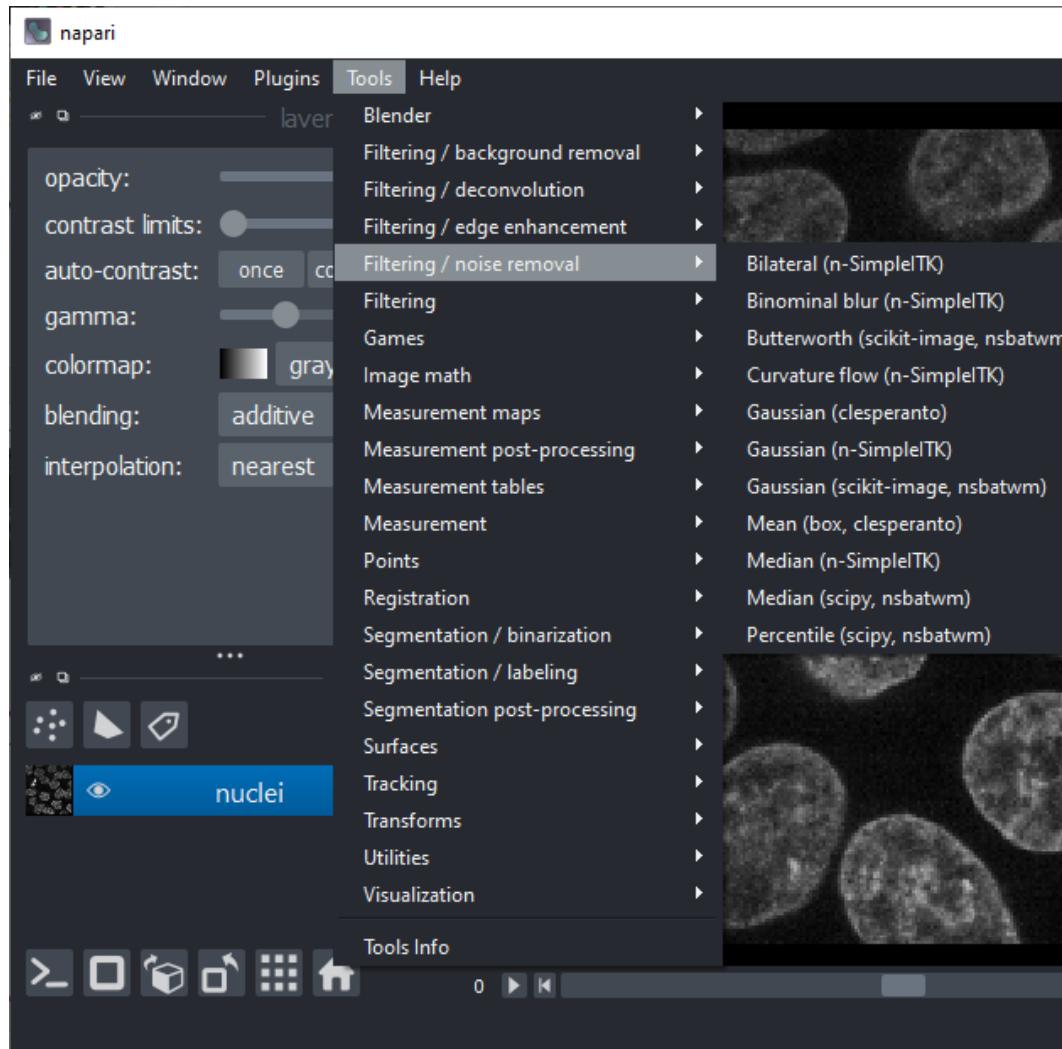
Workflow building

Also check out the Tools > Segmentation / labeling menu



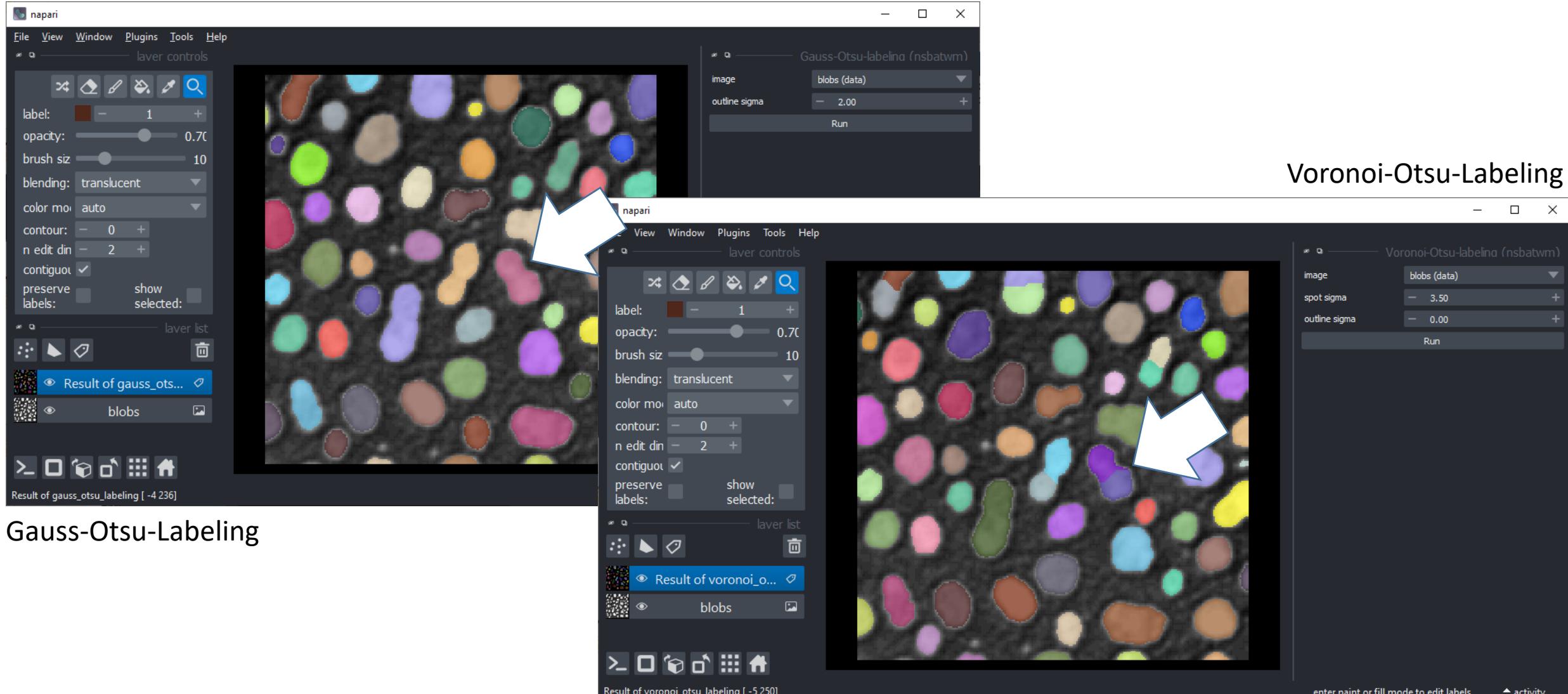
The Tools menu

- Organized in categories similar to what you learned last week



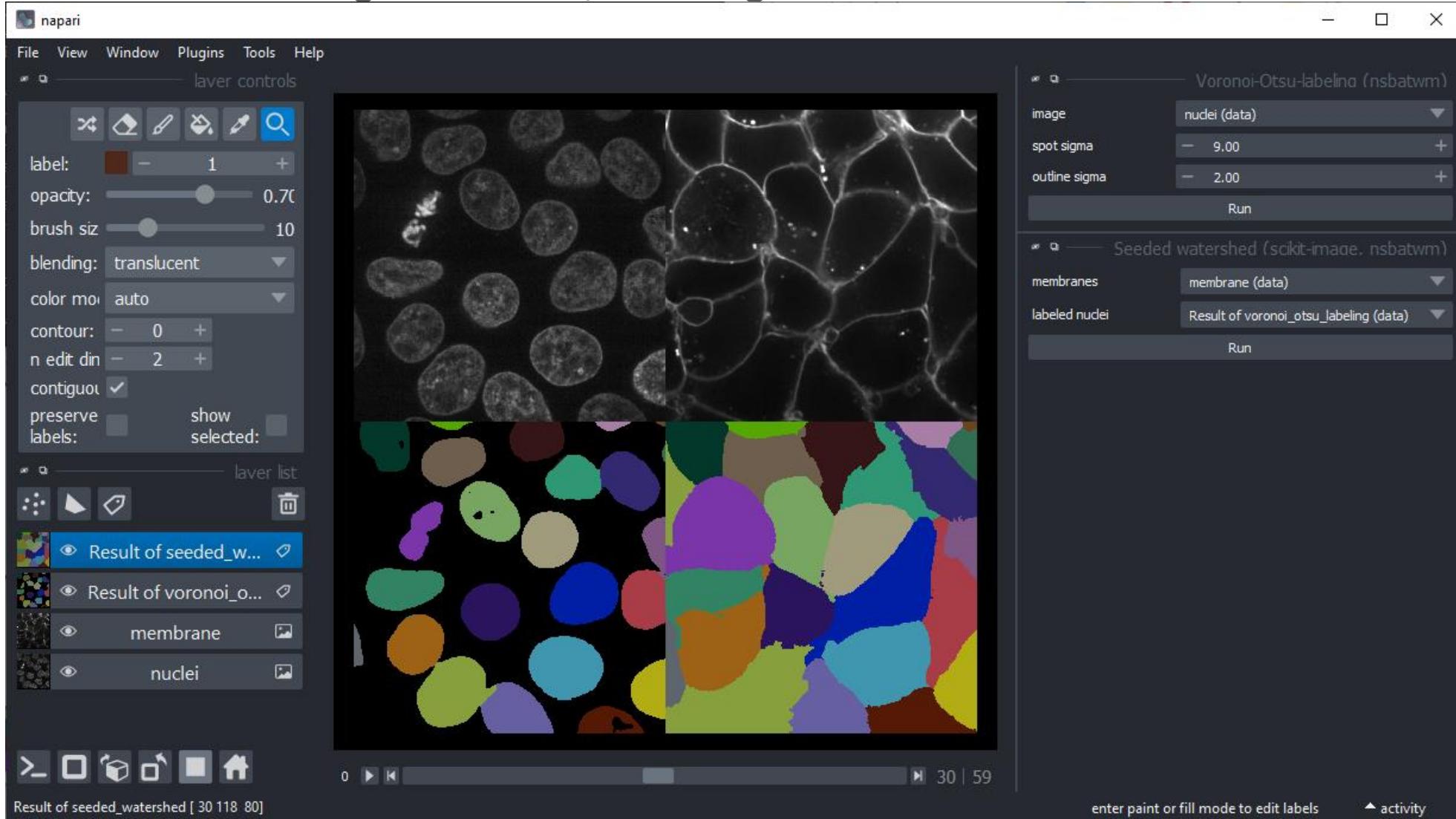
Short-cuts: Voronoi-Otsu-Labeling

Also check out the Tools > Segmentation / labeling menu



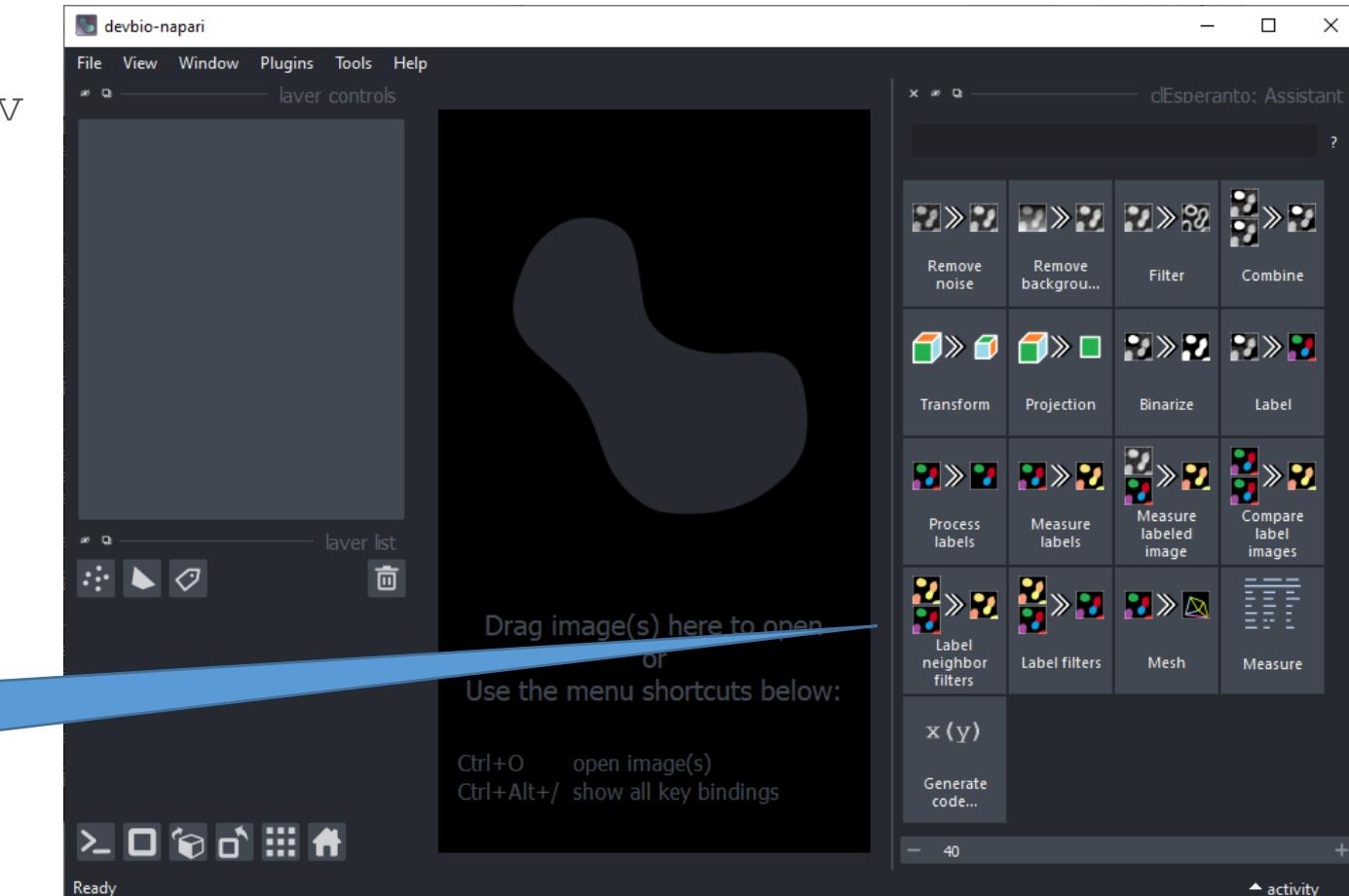
Demo: Nuclei + Cell Segmentation

Also check out the Tools > Segmentation / labeling menu



Demo: run napari-assistant

1. Start up a terminal
2. Activate the environment using
conda activate devbio-napari-env
3. Run:
naparia

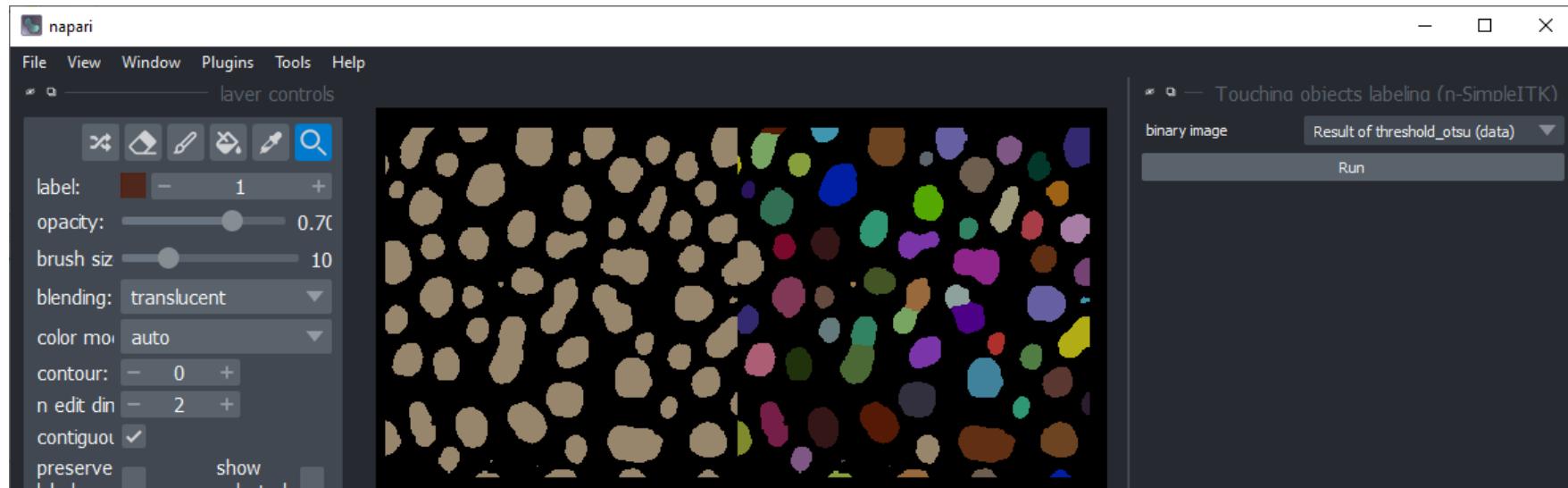


Demo: Nuclei + Cell Segmentation

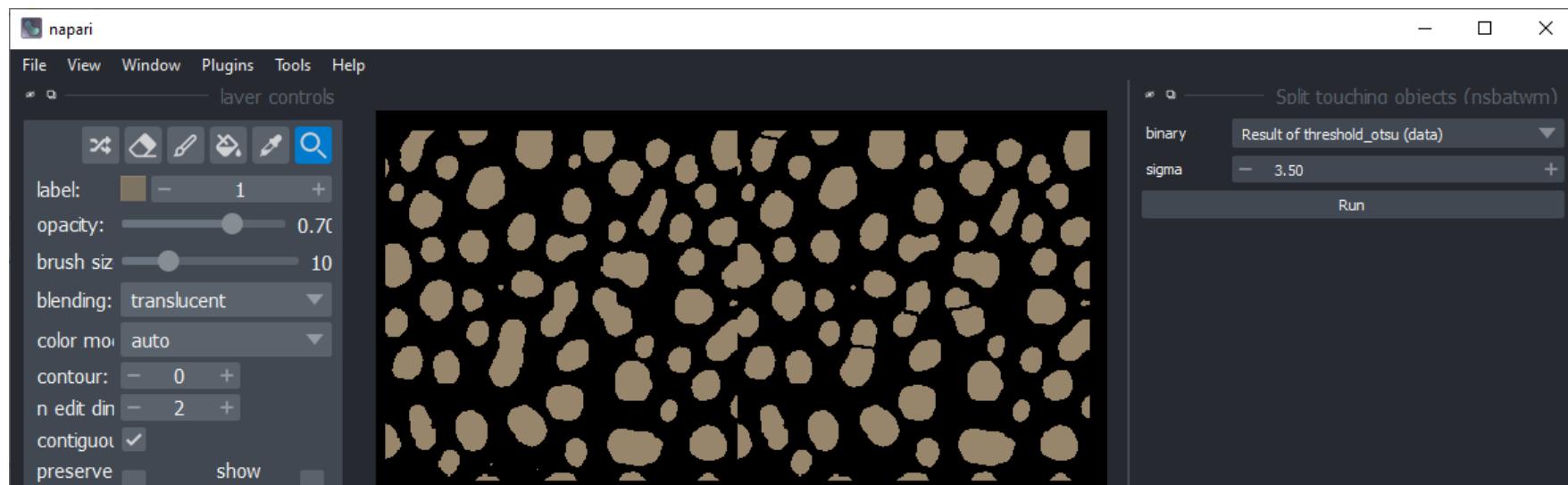
Watershed

- From binary images

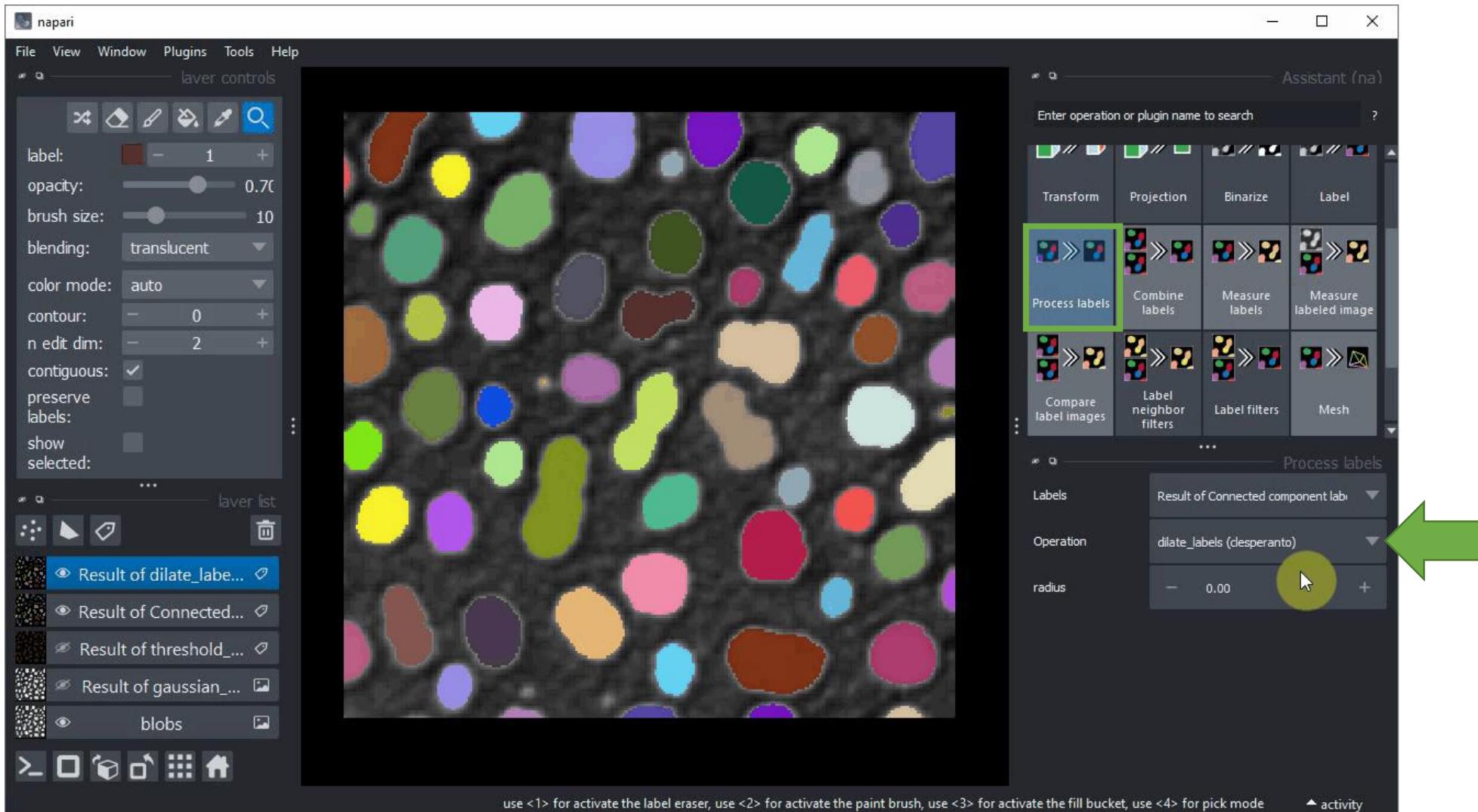
Tools > Segmentation / labeling >
Label touching objects



Tools > Segmentation post-
processing >
Split touching objects
(Similar to ImageJ's Watershed)

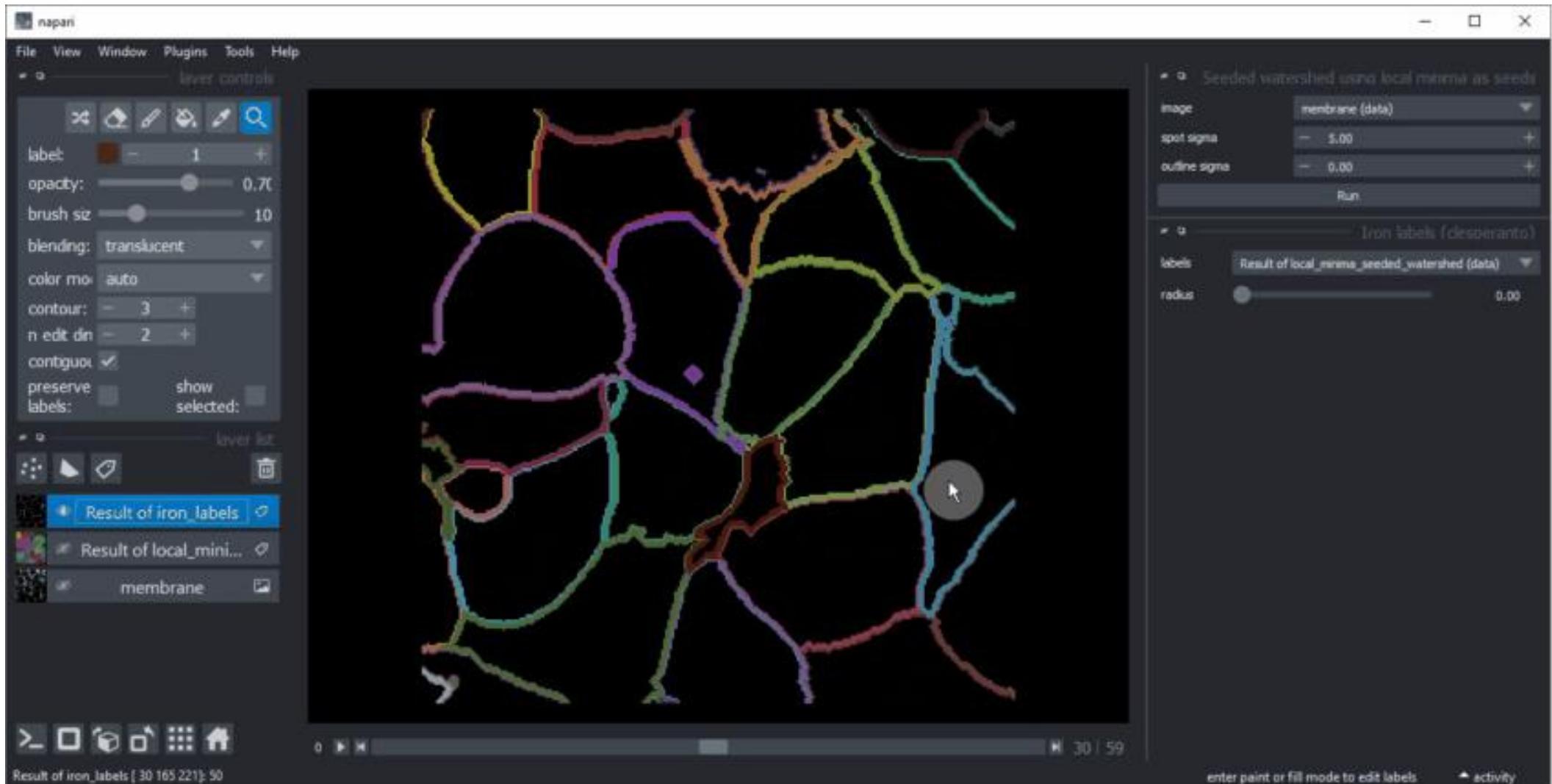


- In Napari Assistant: Process labels



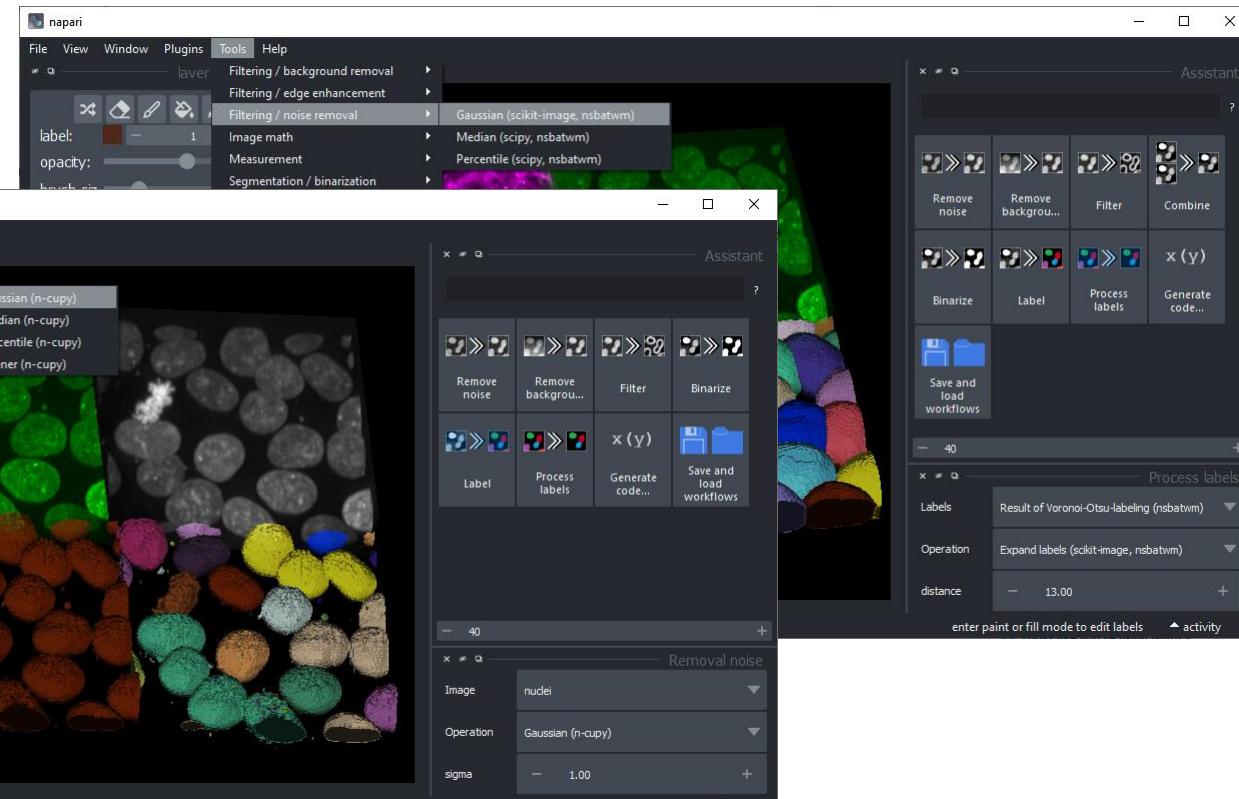
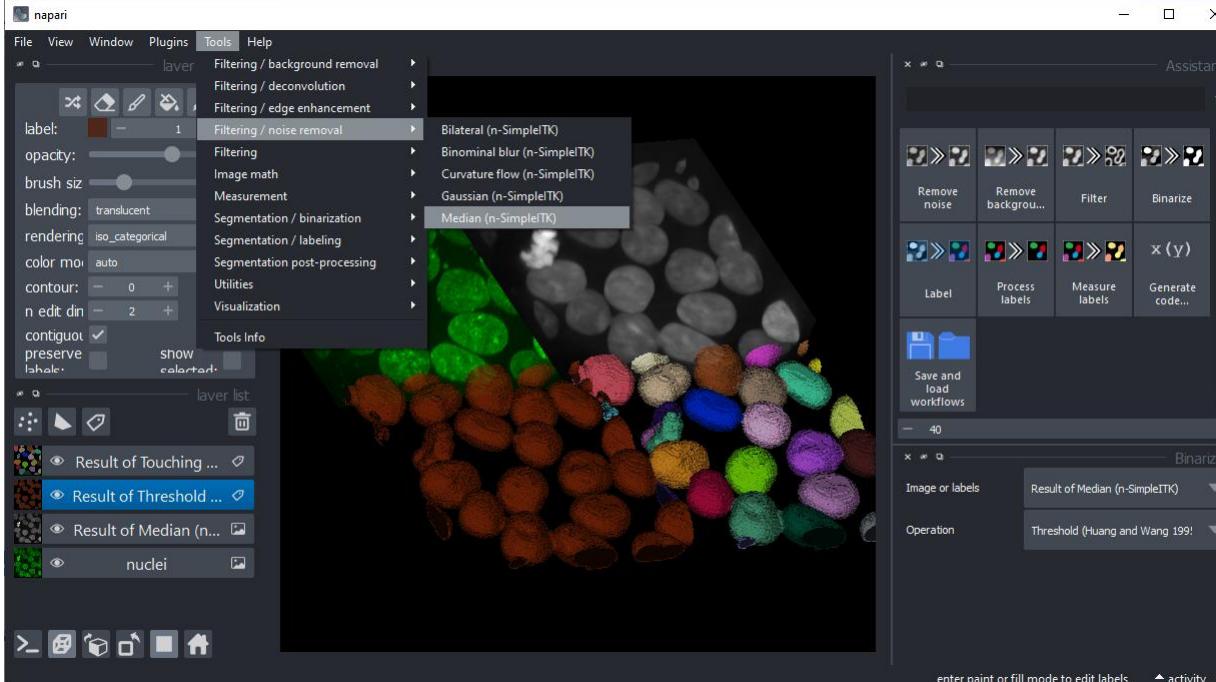
Label post-processing / morphological operations

- In Napari menu Tools > Segmentation post-processing > Smooth labels (clEsperanto)



Napari-Assistant compatible plugins

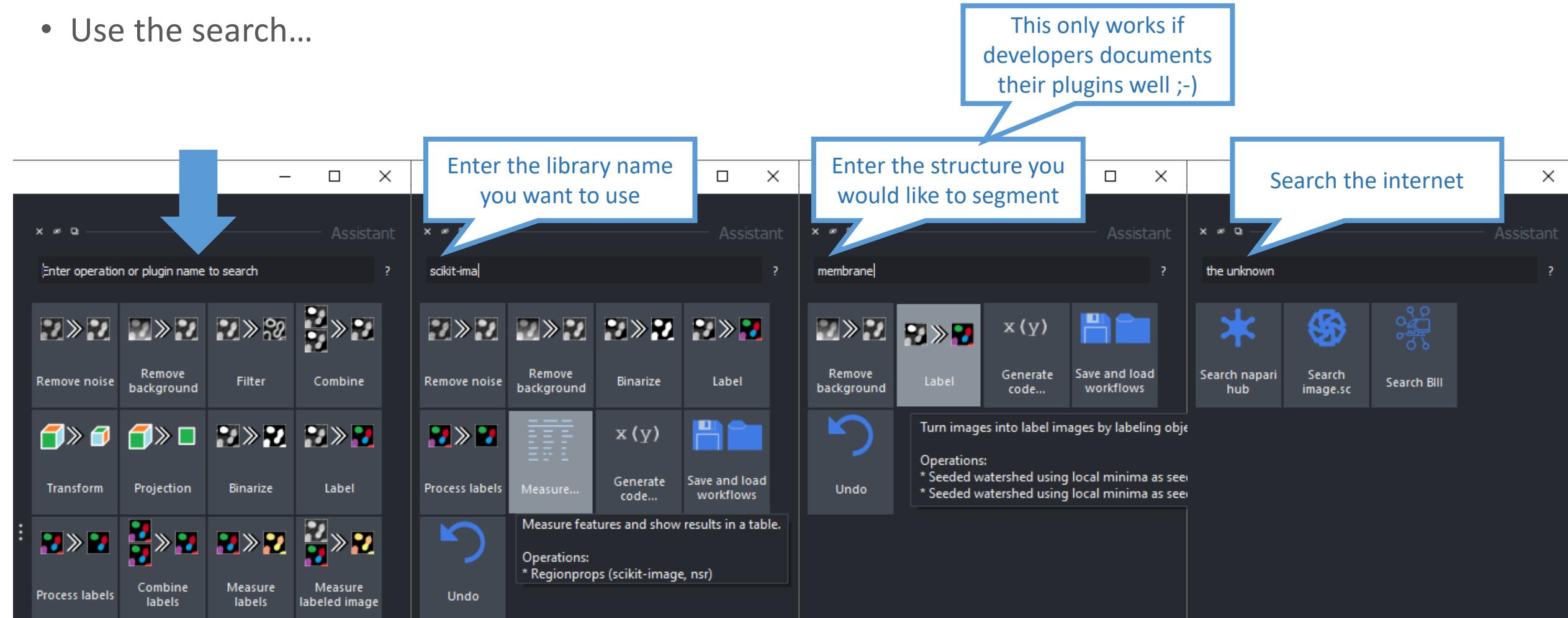
- Classical image-processing algorithms
- Based on scikit-image, scipy, numpy, cupy, clesperanto and SimpleITK



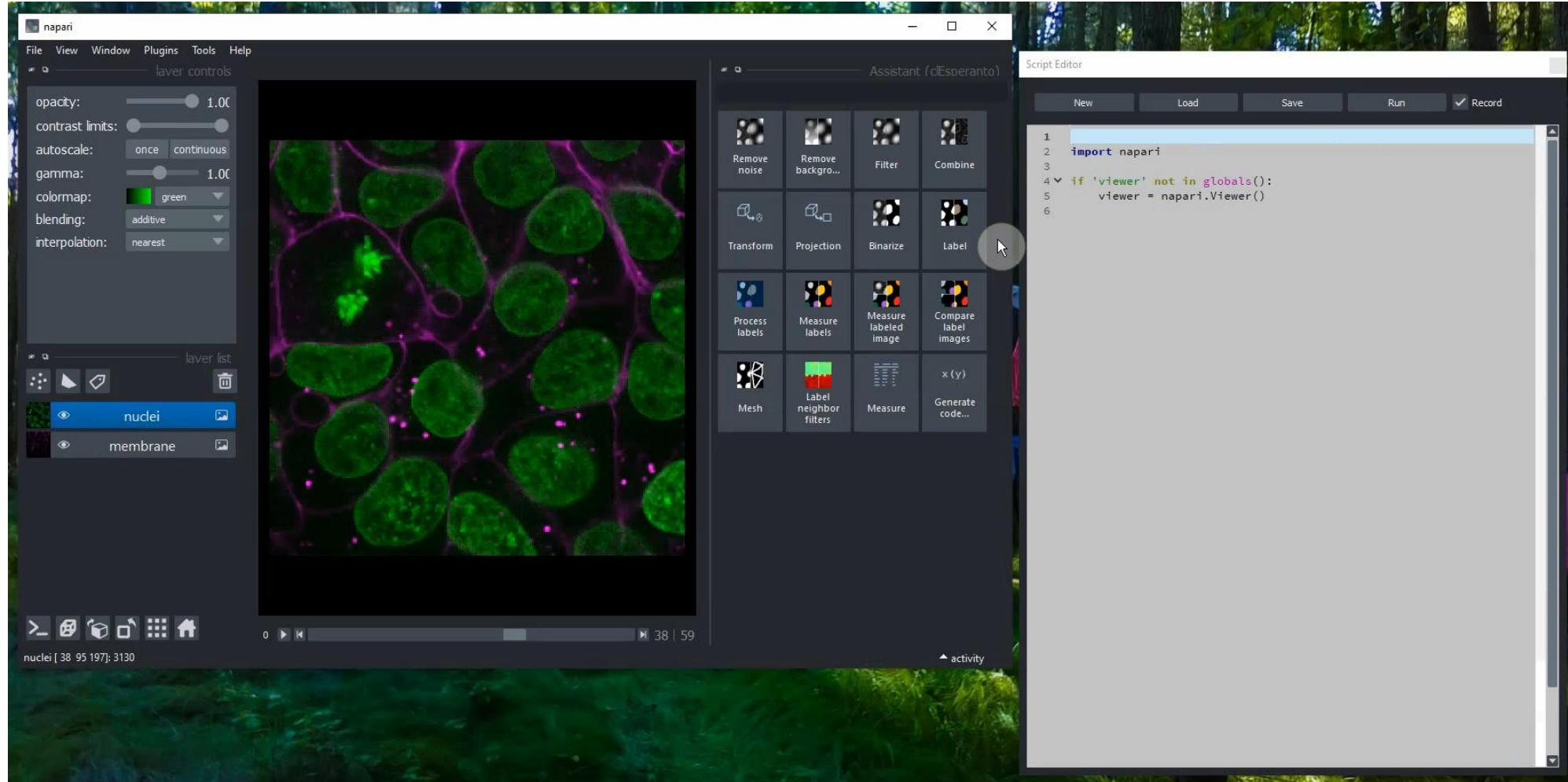
<https://www.napari-hub.org/plugins/napari-segment-blobs-and-things-with-membranes>
<https://www.napari-hub.org/plugins/napari-cupy-image-processing>
<https://www.napari-hub.org/plugins/napari-simpleitk-image-processing>

Browse operations

- Use the search...

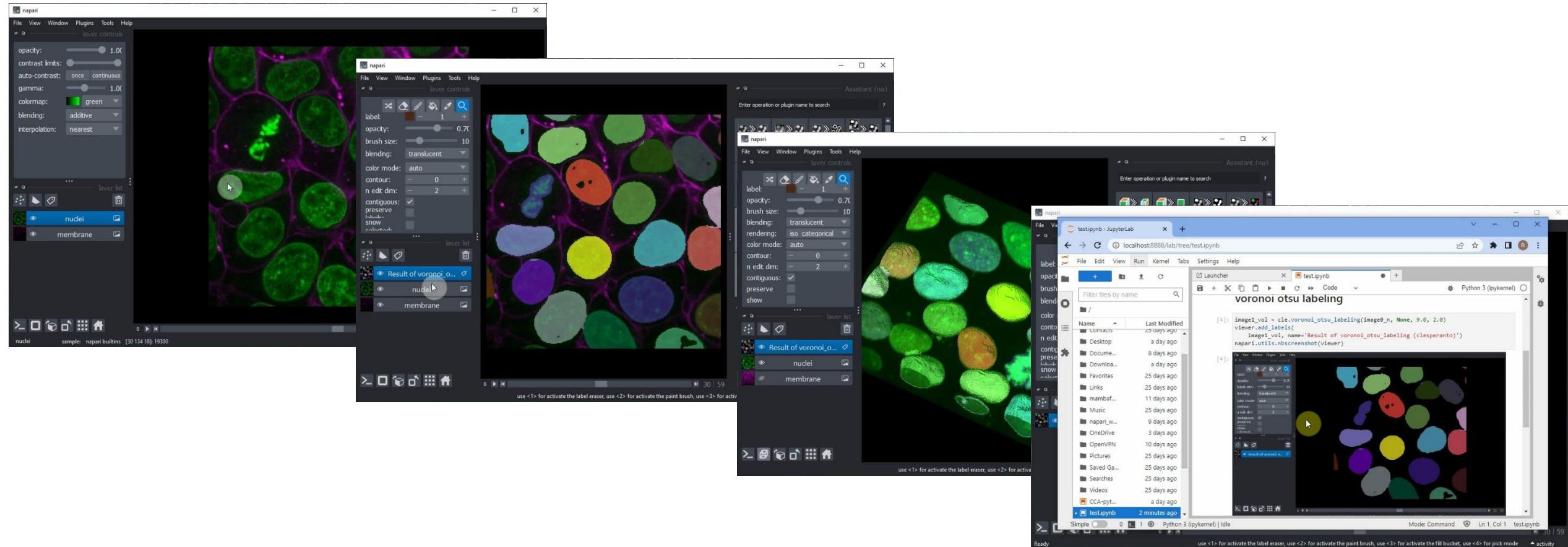


Export code to Python

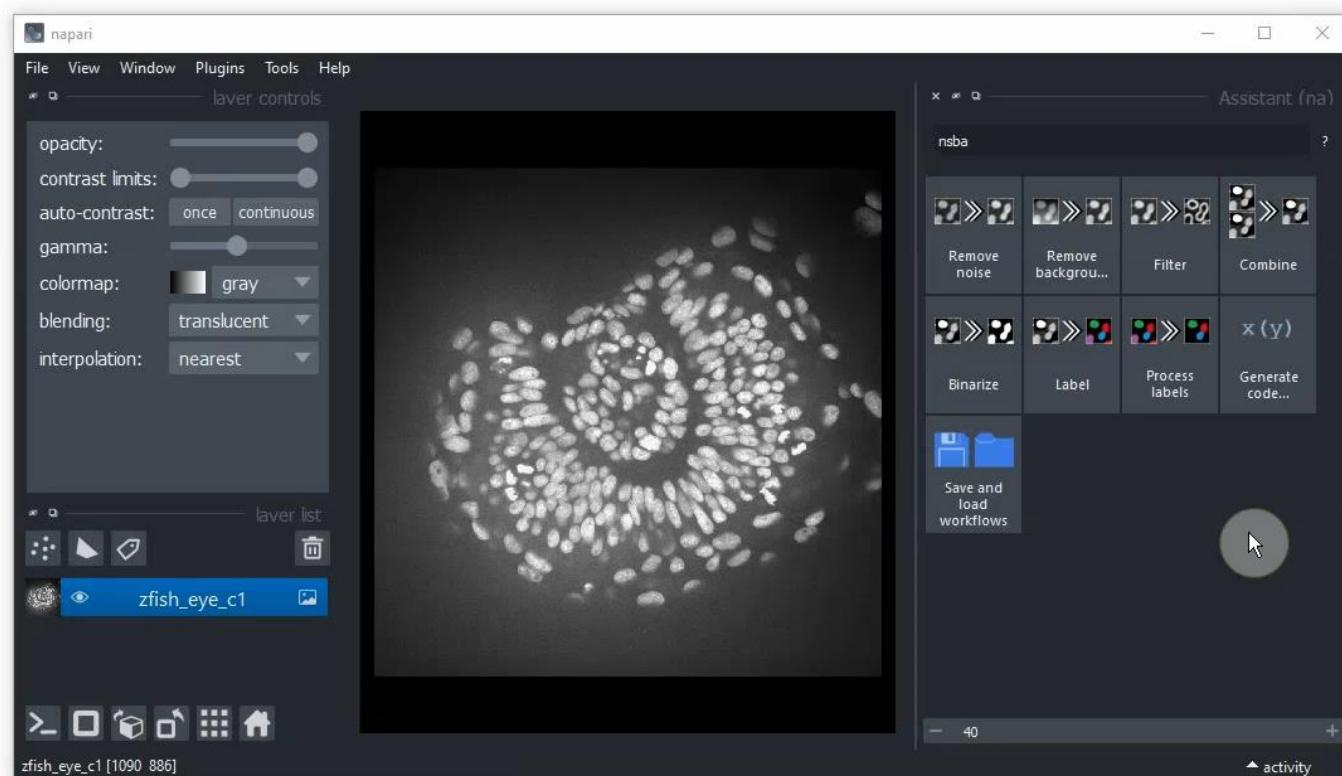


Exercise

- Use the Napari Assistant to generate a Jupyter Notebook



Export code to Jupyter Notebooks

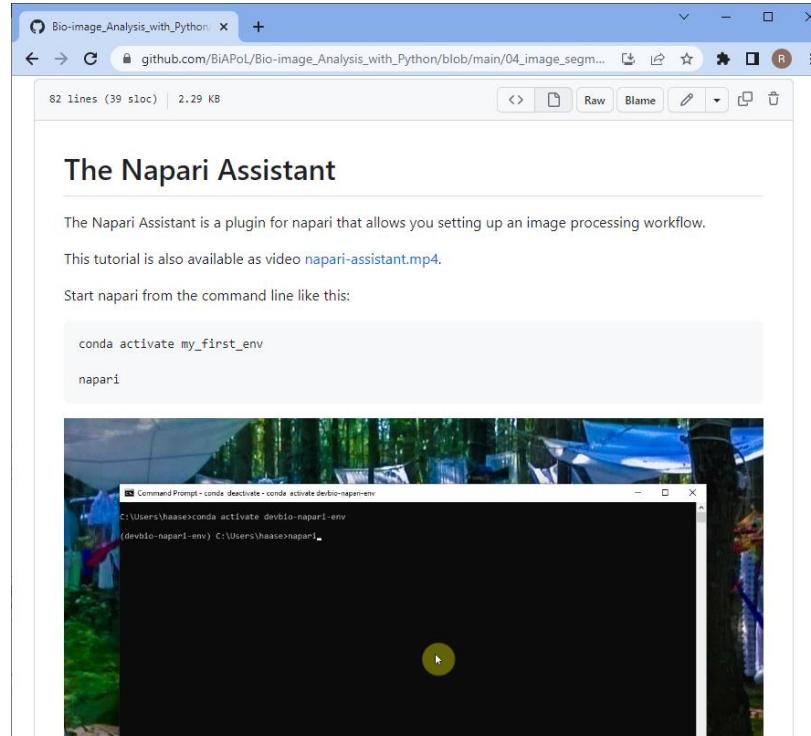


<https://github.com/haesleinhuepf/napari-assistant>

Image data source: Mauricio Rocha Martins, Norden lab, MPI CBG (now at IGC Oeiras)

Exercise

- Follow the online instructions
- https://github.com/BiAPoL/Bio-image Analysis with Python/blob/main/04_image_segmentation/06_napari-assistant.md
- https://github.com/BiAPoL/Bio-image Analysis with Python/blob/main/04_image_segmentation/07_notebook_export.md



Generating Jupyter Notebooks from the Napari Assistant

After setting up a workflow using the Napari Assistant, we can export Python code, e.g. as Jupyter Notebook. This tutorial is also available as video [export_notebooks.mp4](#).

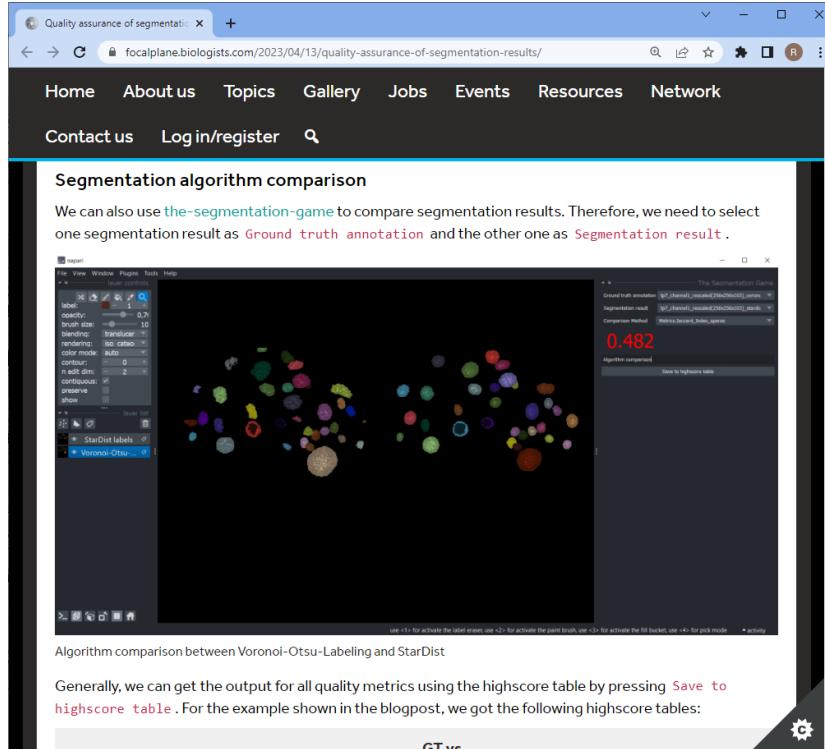
In the Assistant panel, click on the `Generate Code...` button and the `Export Jupyter Notebook using Napari` menu.

Afterwards, rerun the notebook and inspect the result. The Napari viewer that opened in the background will also be shown within the notebook.

Further reading

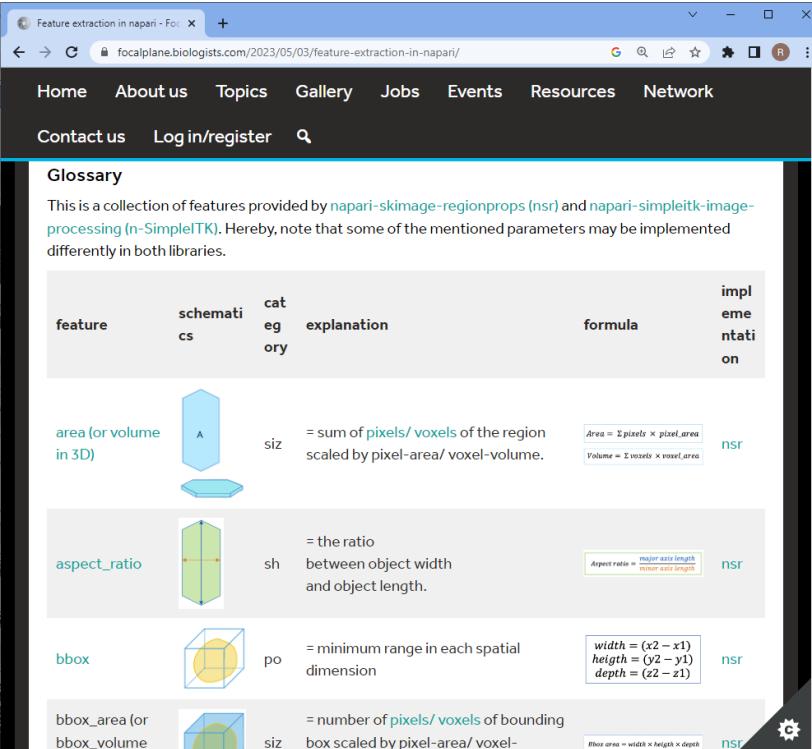
- Blog posts on the Focalplane about...

Segmentation Quality Assurance



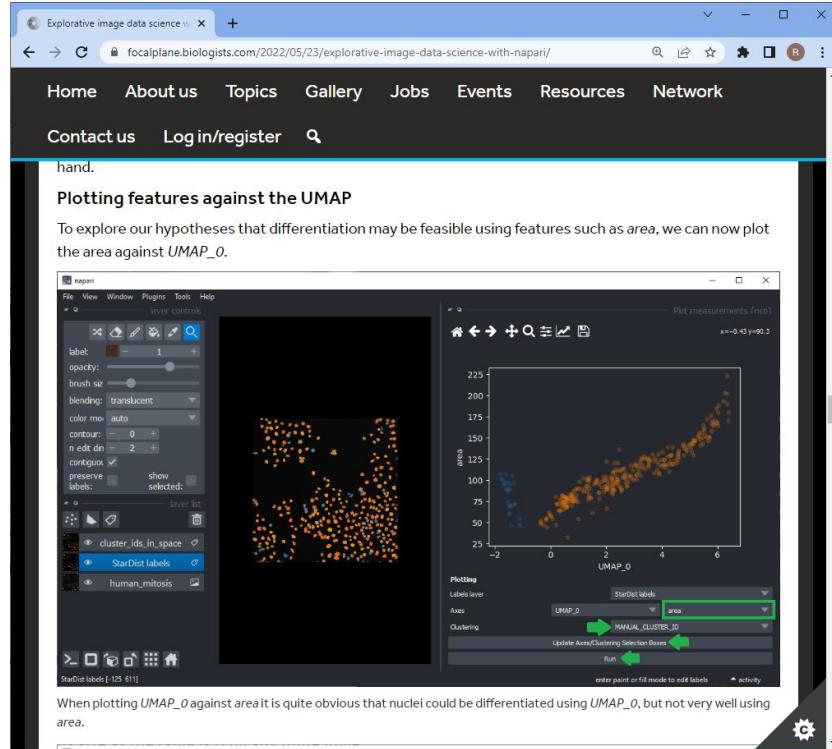
<https://focalplane.biologists.com/2023/04/13/quality-assurance-of-segmentation-results/>

Feature Extraction



<https://focalplane.biologists.com/2023/05/03/feature-extraction-in-napari/>

Unsupervised machine learning



<https://focalplane.biologists.com/2022/05/23/explorative-image-data-science-with-napari/>

Acknowledgements



BiAPoL team

- Mara Lampert
 - Marcelo Zoccoler
 - Johannes Soltwedel
 - Maleeha Hassan
 - Allyson Ryan
 - Till Korten
 - Stefan Hahmann
 - Somashekhar Kulkarni
- Former lab members:
- Ryan George Savill
 - Laura Zigutyte

Networks



Funding

