

# PoL Bio-Image Analysis Symposium 2023

## Training School Early Career Track

# Introduction to Napari

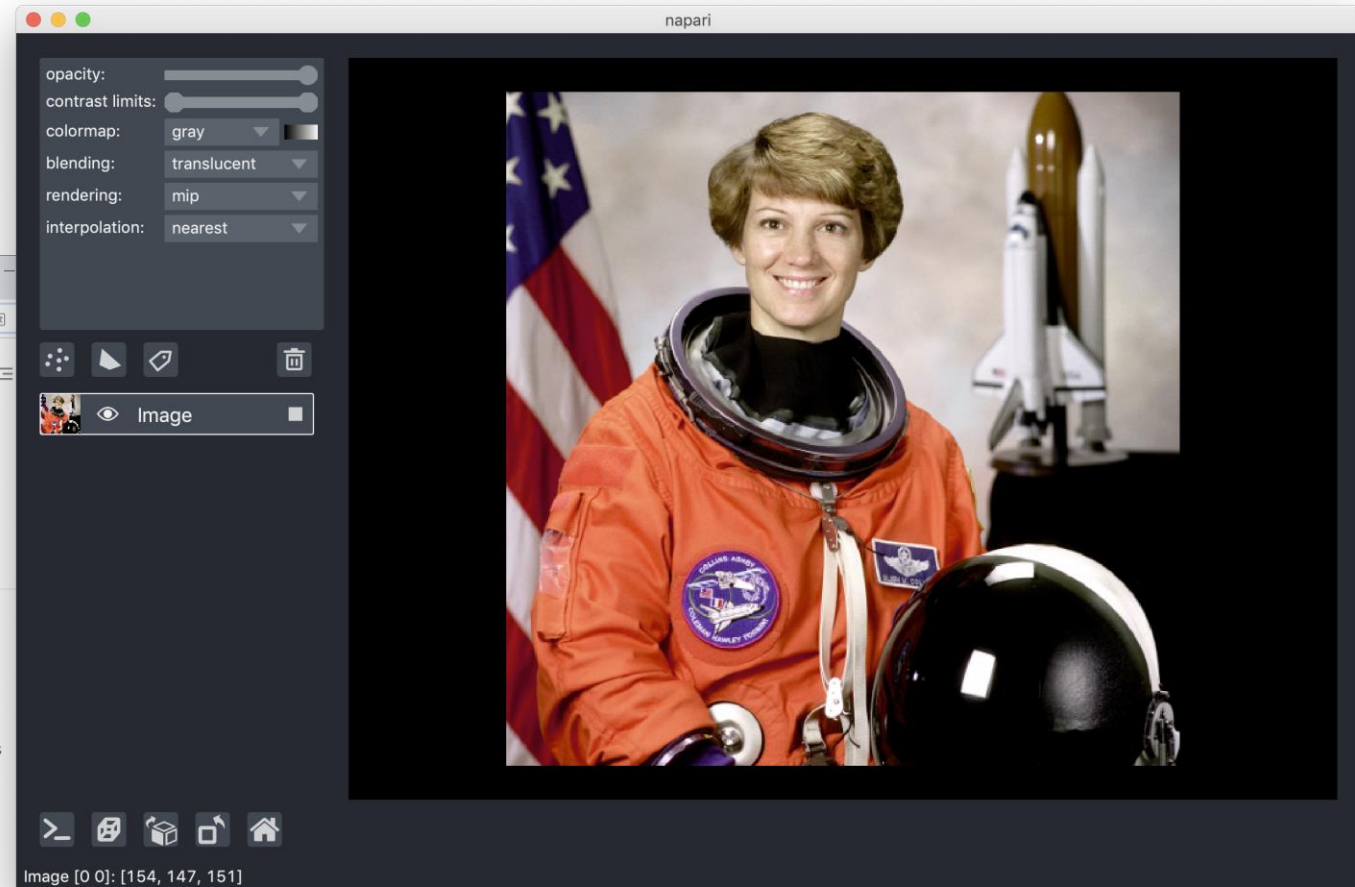
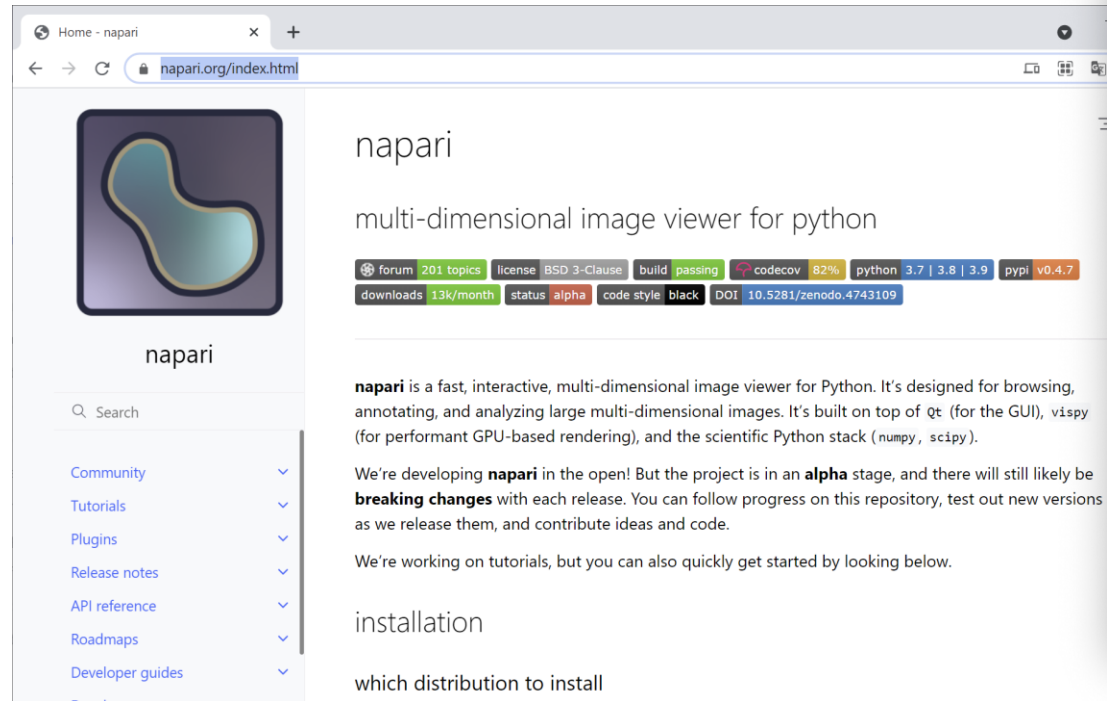
Marcelo Leomil Zoccoler

With material from:

Robert Haase, PoL; Guillaume Witz, Universität Bern

# Napari: 3D viewer for Python

- Multi-dimensional image viewer in python
- <https://napari.org/>



<https://napari.org/>

# Napari: 3D viewer for Python



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

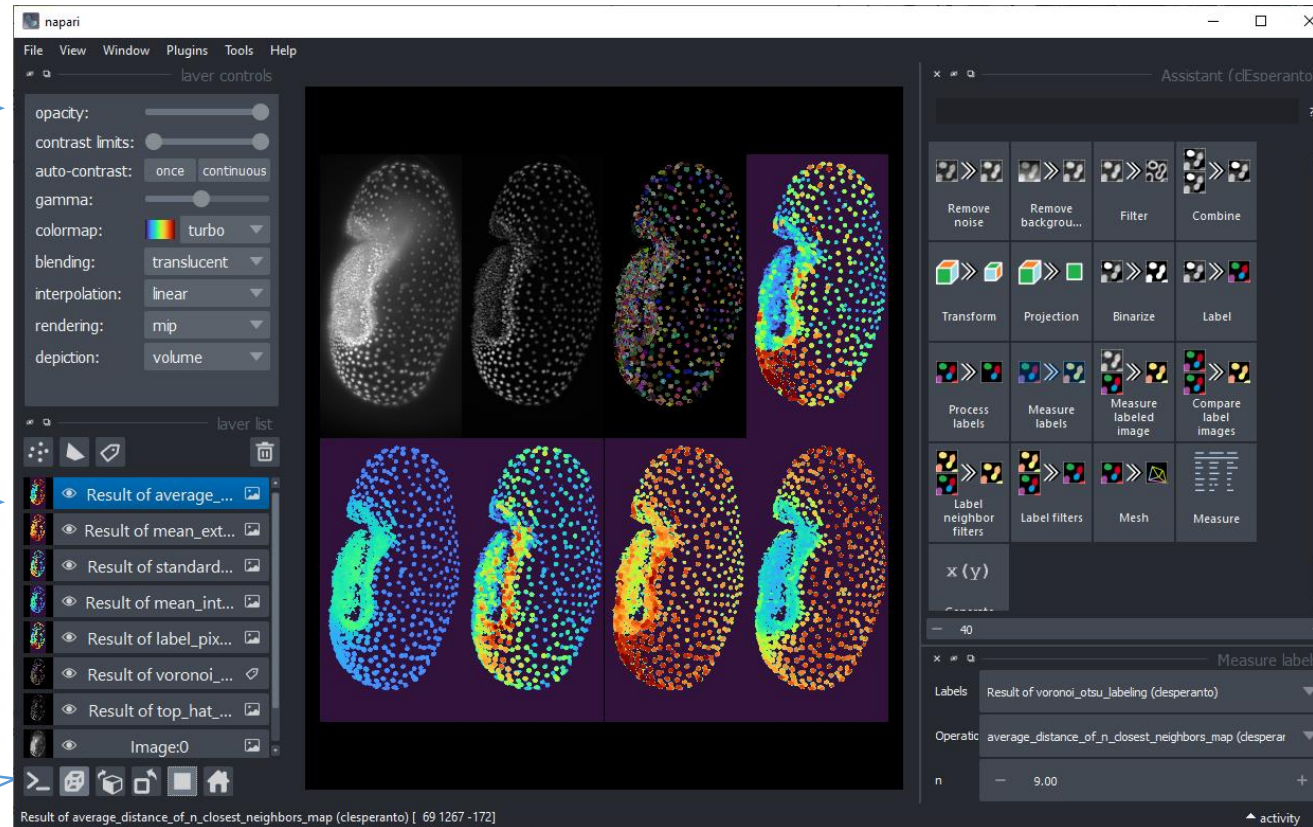
View configuration /  
tools

```
layer.opacity = 0.5
```

Layers

```
layer.visible = False
```

Viewer controls



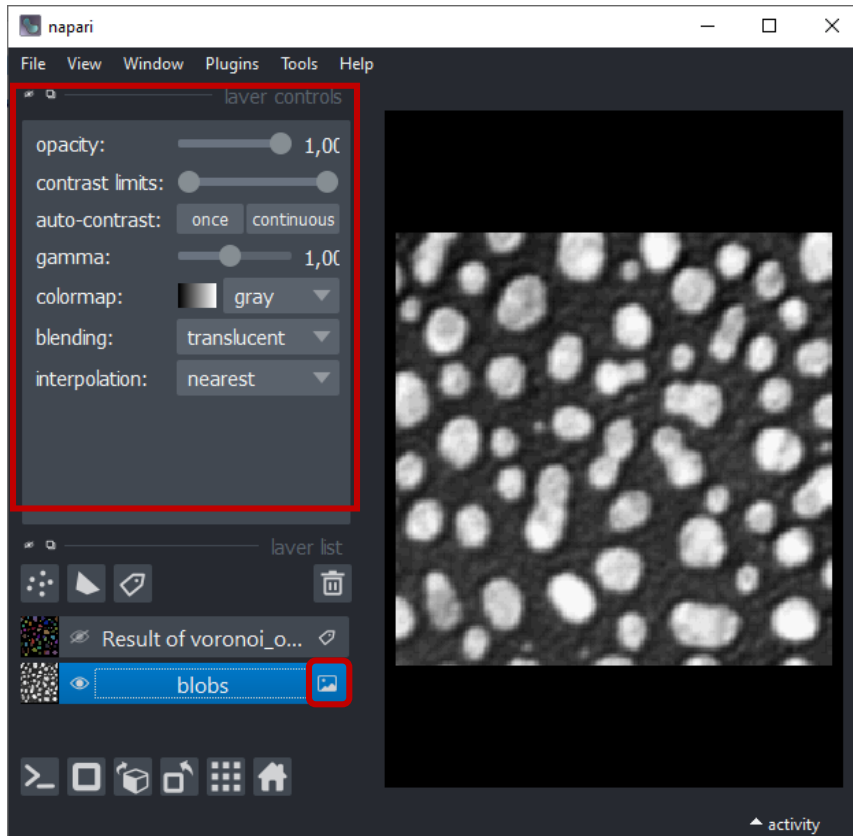
Dock widgets  
(custom plugins)

Function widgets  
(custom plugins)

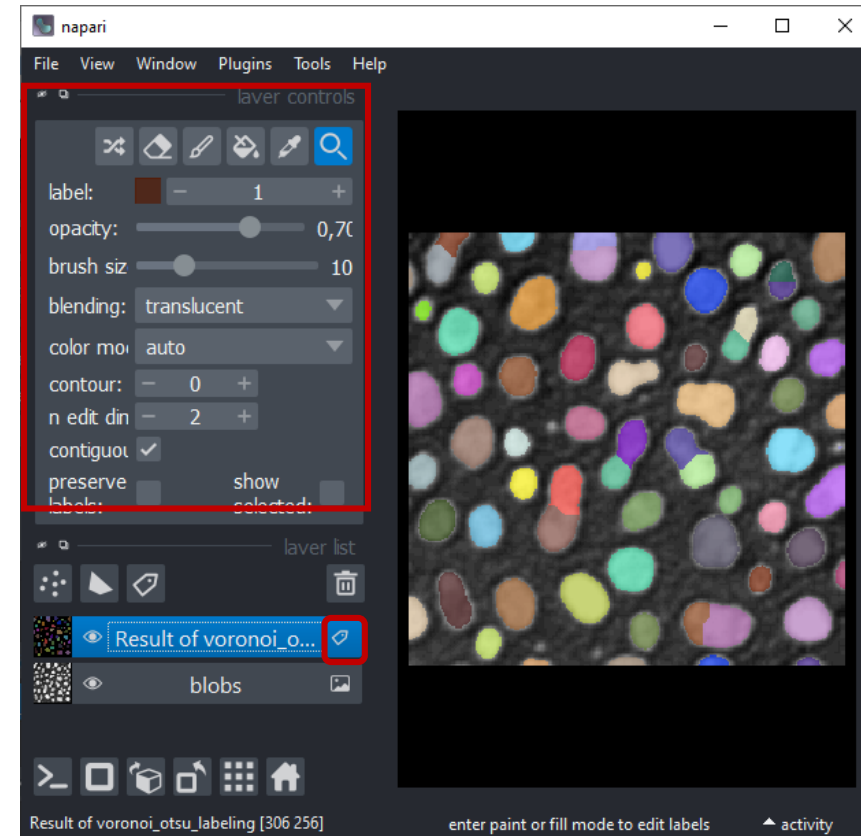


- Different layers have different tools and options

Image Layer

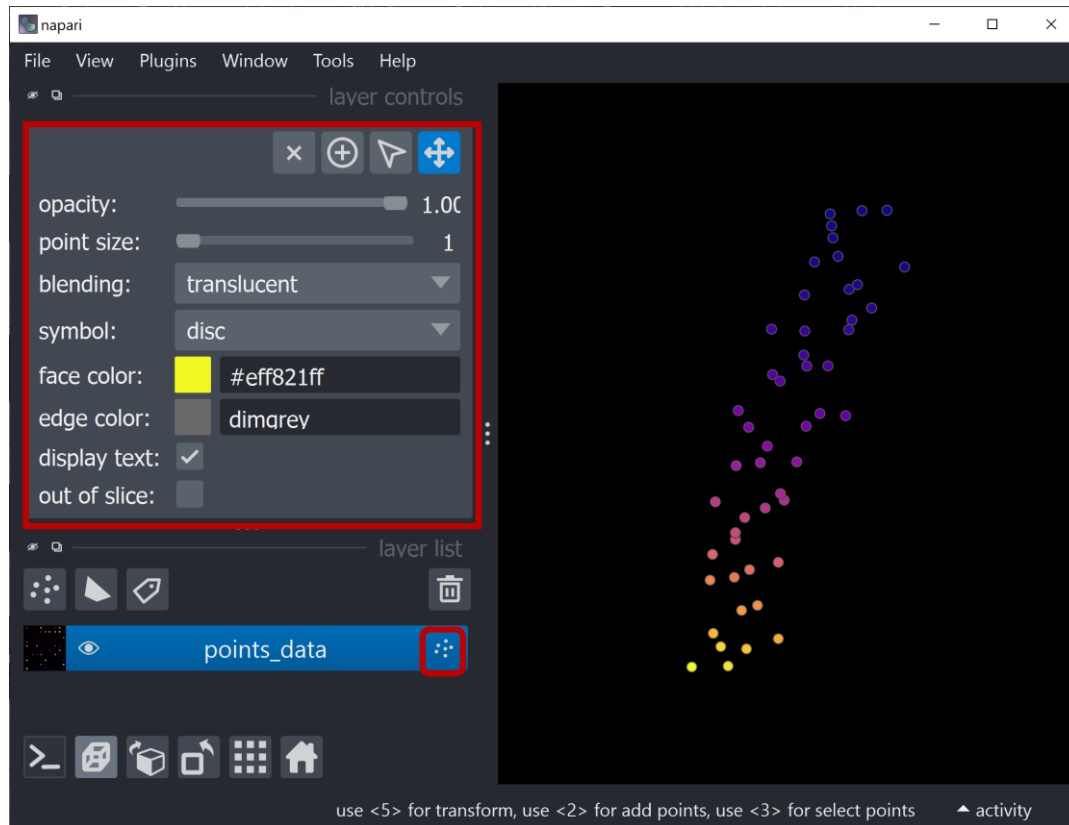


Labels Layer

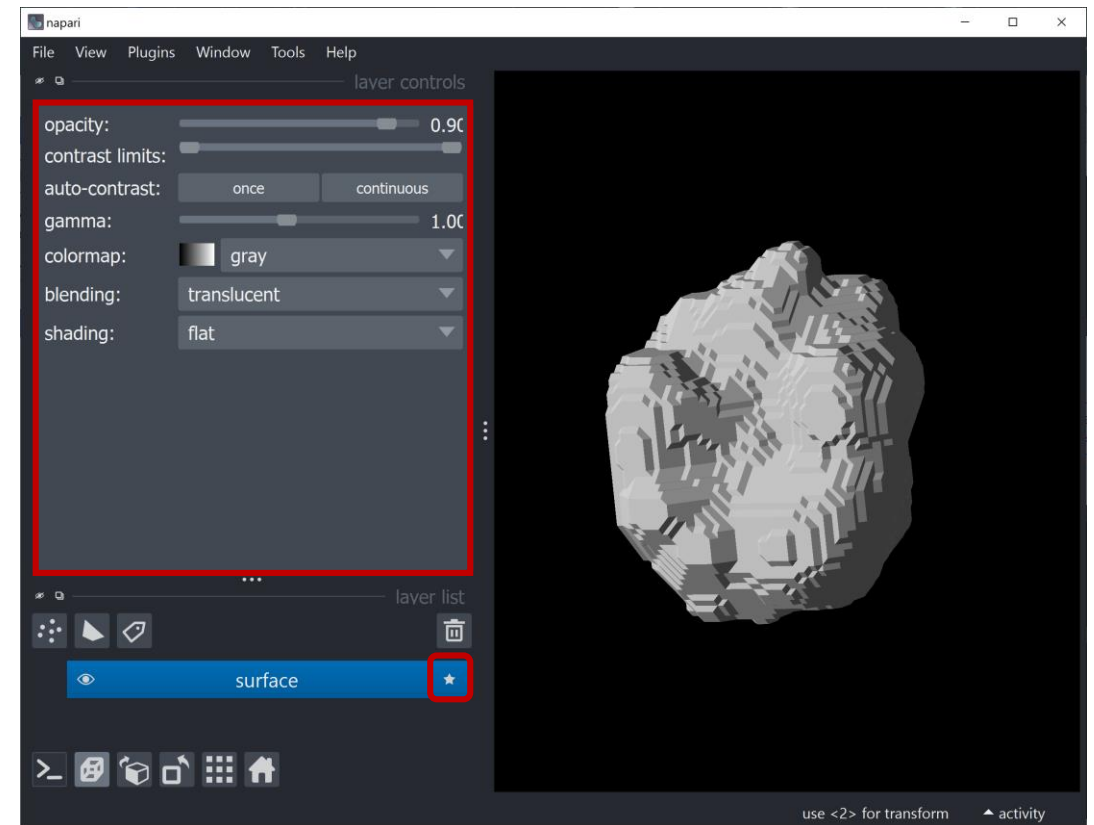


- Different layers have different tools and options

Points Layer



Surface Layer

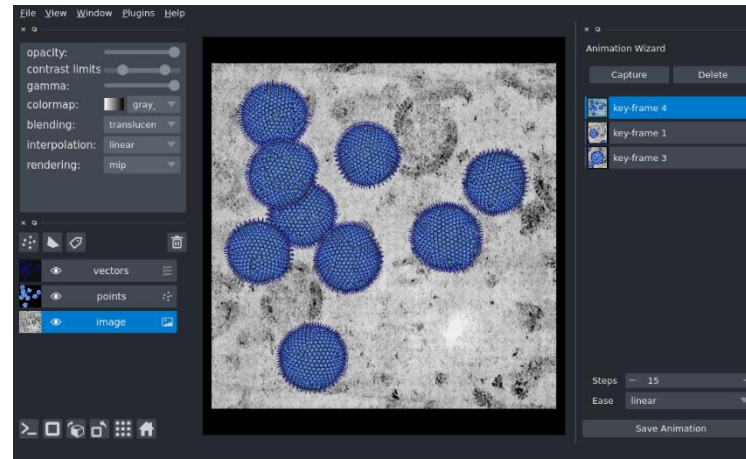


## clusters-plotter



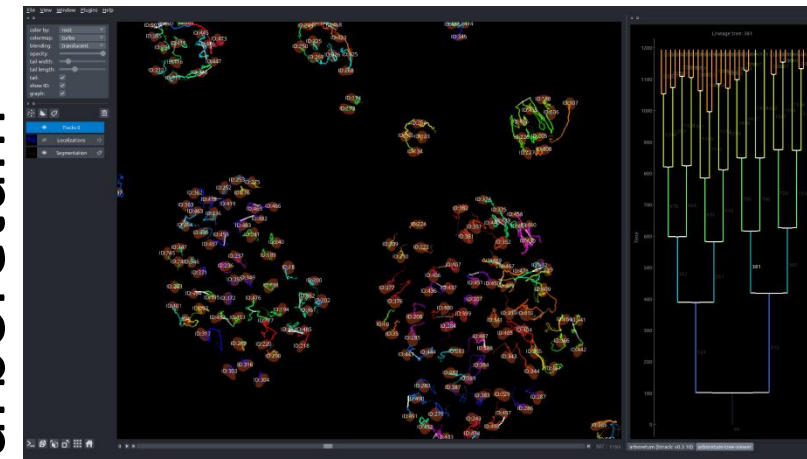
<https://github.com/BiAPoL/napari-clusters-plotter>

## animation



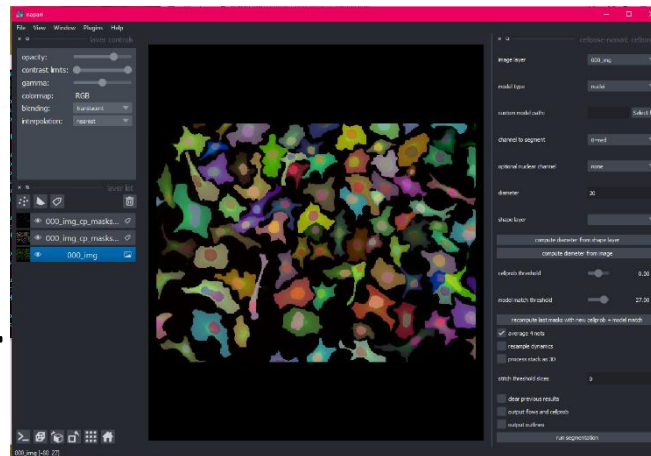
<https://github.com/napari/napari-animation>

## arboretum



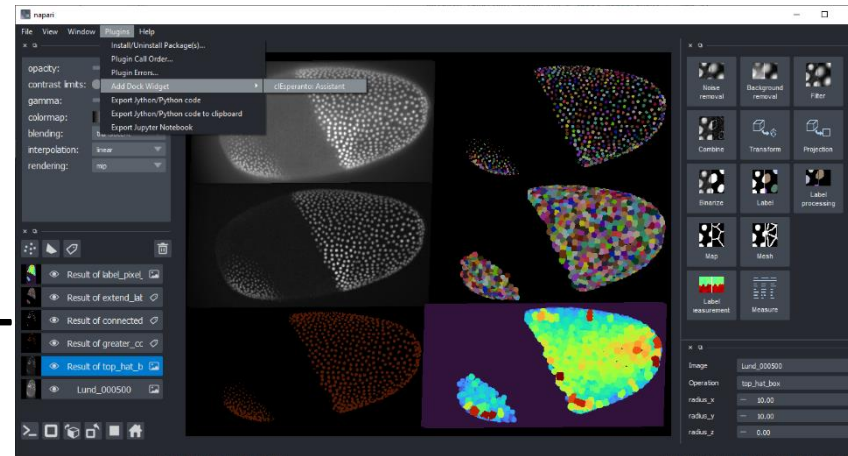
<https://github.com/quantumjot/arboretum>

## cellpose



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

## clesperanto




[https://github.com/clEsperanto/napari\\_pyclesperanto\\_assistant](https://github.com/clEsperanto/napari_pyclesperanto_assistant)

In development: <https://github.com/topics/napari-plugin>  
Released: <https://pypi.org/search/?q=&o=&c=Framework+%3A%3A+napari>

Correspondence | [Published: 18 August 2023](#)

## **napari-imagej: ImageJ ecosystem access from napari**

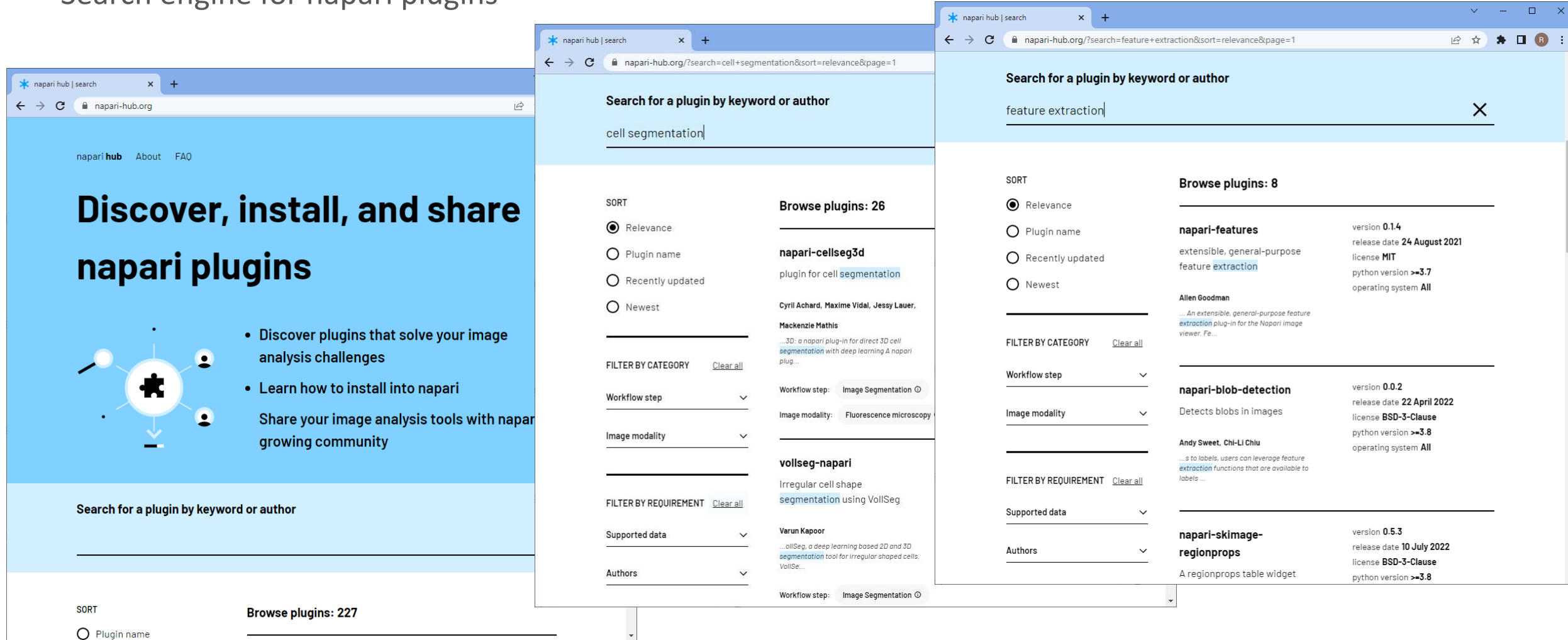
[Gabriel J. Selzer](#), [Curtis T. Rueden](#), [Mark C. Hiner](#), [Edward L. Evans III](#), [Kyle I. S. Harrington](#) & [Kevin W. Eliceiri](#) 

[Nature Methods](#) (2023) | [Cite this article](#)

**733** Accesses | **56** Altmetric | [Metrics](#)



- The plugin you are looking for may be near you!
- Search engine for napari plugins



The screenshots illustrate the Napari Hub search interface. The main page features a large heading "Discover, install, and share napari plugins" and a search bar. The search results are displayed in a grid, showing plugin details such as name, version, release date, license, and author. The search results are filtered by relevance, and the results are sorted by relevance. The search results are displayed in a grid, showing plugin details such as name, version, release date, license, and author.

**Search for a plugin by keyword or author**

cell segmentation

**Sort**

- ☒ Relevance
- ☐ Plugin name
- ☐ Recently updated
- ☐ Newest

**Filter by category** [Clear all](#)

Workflow step: Image Segmentation

Image modality: Fluorescence microscopy

**Filter by requirement** [Clear all](#)

Supported data

Authors

**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  
...ollSeg, a deep learning based 2D and 3D segmentation tool for irregular shaped cells, VollSe...

Workflow step: Image Segmentation

**Search for a plugin by keyword or author**

feature extraction

**Sort**

- ☒ Relevance
- ☐ Plugin name
- ☐ Recently updated
- ☐ Newest

**Filter by category** [Clear all](#)

Workflow step

Image modality

**Filter by requirement** [Clear all](#)

Supported data

Authors

**Browse plugins: 8**

**napari-features**  
extensible, general-purpose feature extraction  
version 0.1.4  
release date 24 August 2021  
license MIT  
python version >=3.7  
operating system All  
Allen Goodman  
... An extensible, general-purpose feature extraction plug-in for the Napari image viewer. Fe...

**napari-blob-detection**  
Detects blobs in images  
version 0.0.2  
release date 22 April 2022  
license BSD-3-Clause  
python version >=3.8  
operating system All  
Andy Sweet, Chi-Li Chiu  
...s to labels, users can leverage feature extraction functions that are available to labels ...

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

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

# Napari and Jupyter Notebooks

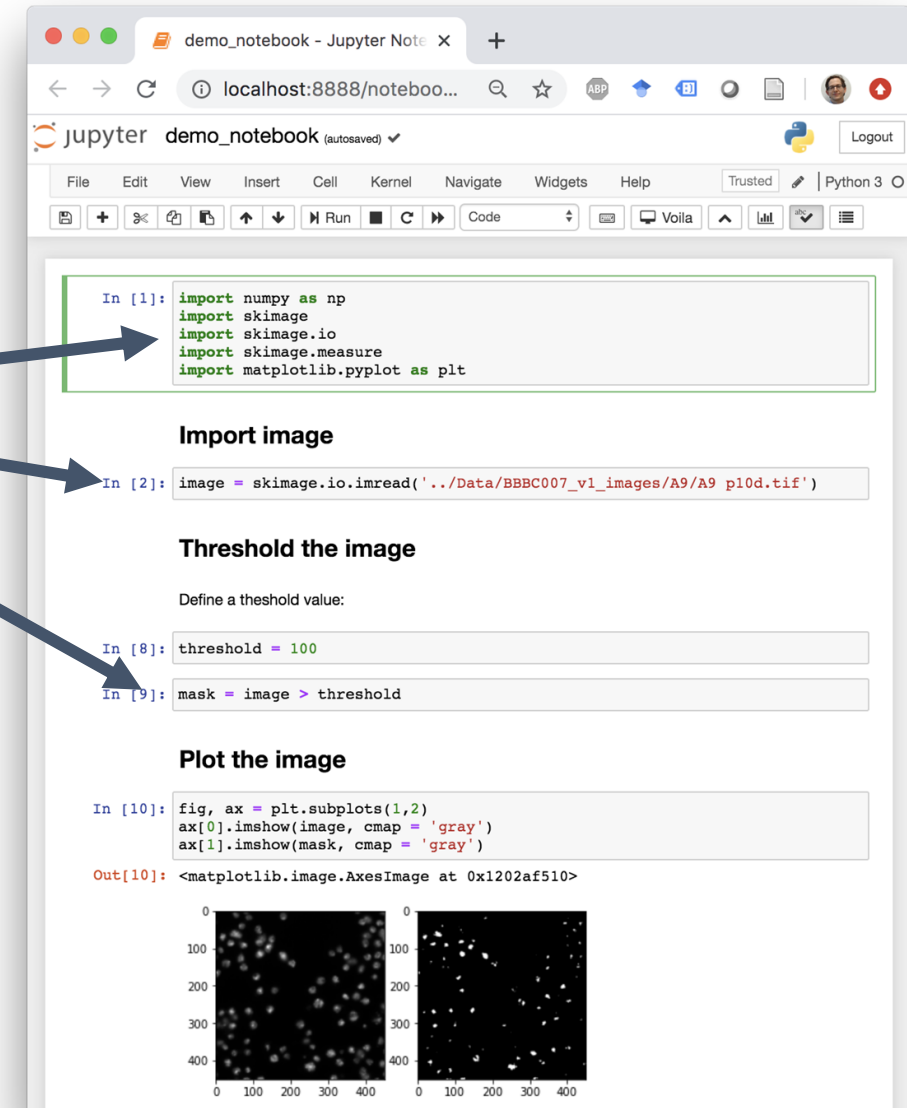
# Napari from jupyter notebooks

What is a jupyter notebook?

A text file (easily sent around)

Rendered by Jupyter in the browser

Split into sections called cells



What is a jupyter notebook?

A text file (easily sent around)

Rendered by Jupyter in the browser

Split into sections called cells

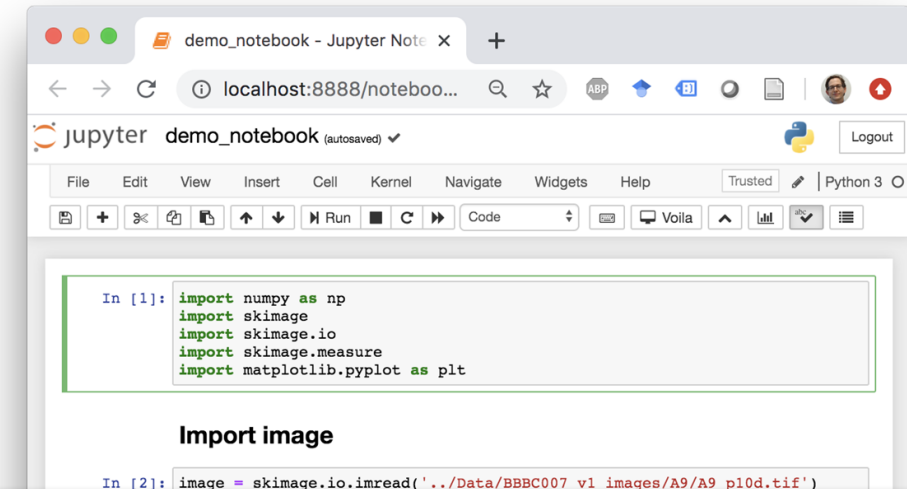
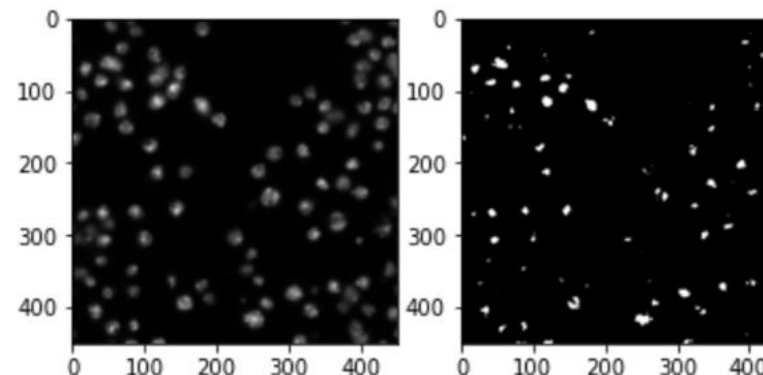
Cells can contain:

- Code
- Formatted text
- Rich output

In [2]: image

```
In [10]: fig, ax = plt.subplots(1,2)
ax[0].imshow(image, cmap = 'gray')
ax[1].imshow(mask, cmap = 'gray')

Out[10]: <matplotlib.image.AxesImage at 0x1202af510>
```





- Keep all the benefits from using code:
  - Batch processing
  - Running python functions/tools still unavailable as plugins

- Easy data interaction and visualization with napari:
  - Great for visualizing 3D (and more) data
  - Each processing step result can be displayed as a separate layer
- Data annotation

- From command line:

```
Miniforge Prompt

(base) C:\Users\mazo260d>mamba activate devbio-napari-env

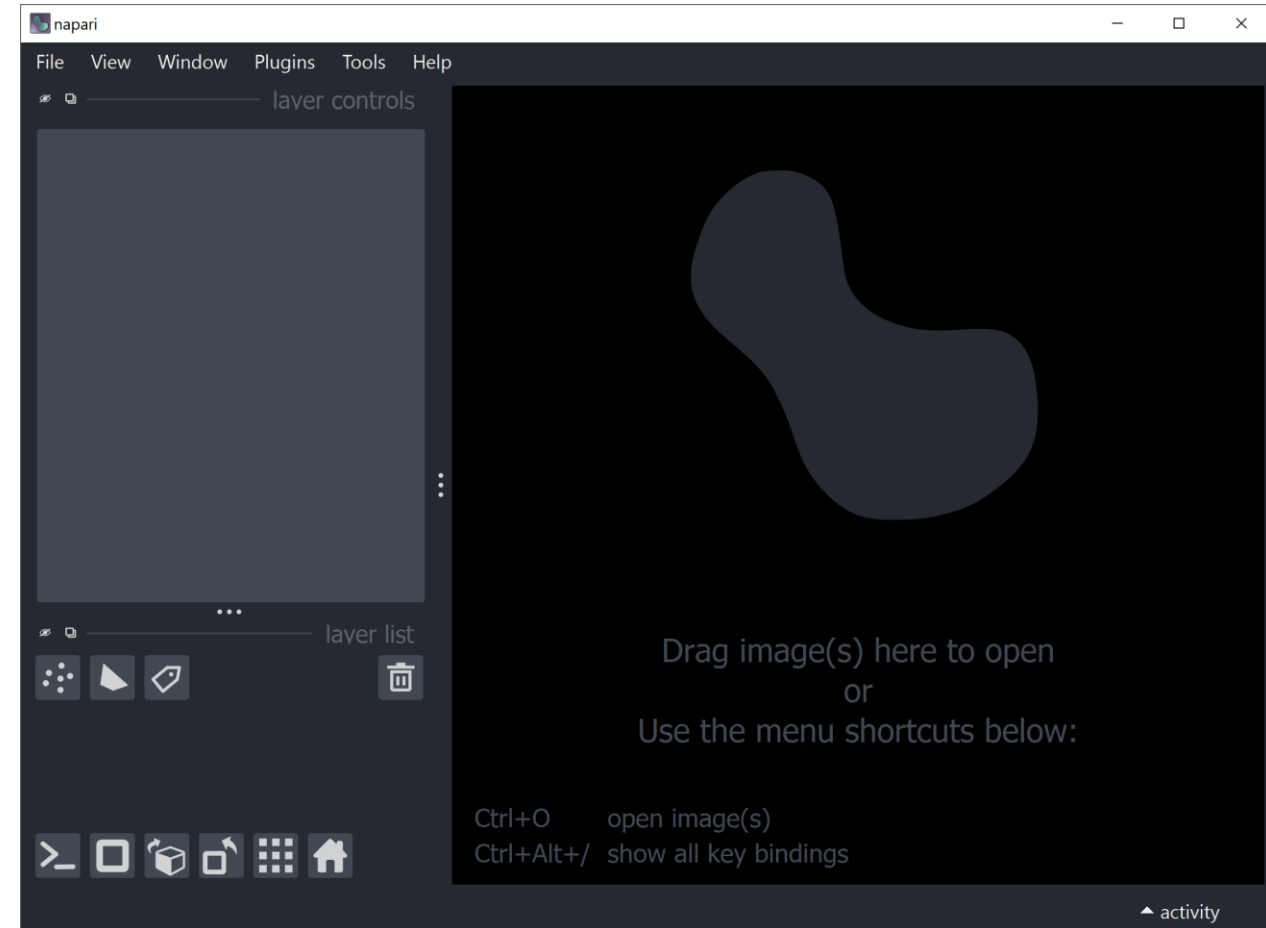
(devbio-napari-env) C:\Users\mazo260d>napari
```

- From code (jupyter notebook):

```
[1]: import napari
```

```
[*]: viewer = napari.Viewer()
```

```
[ ]:
```

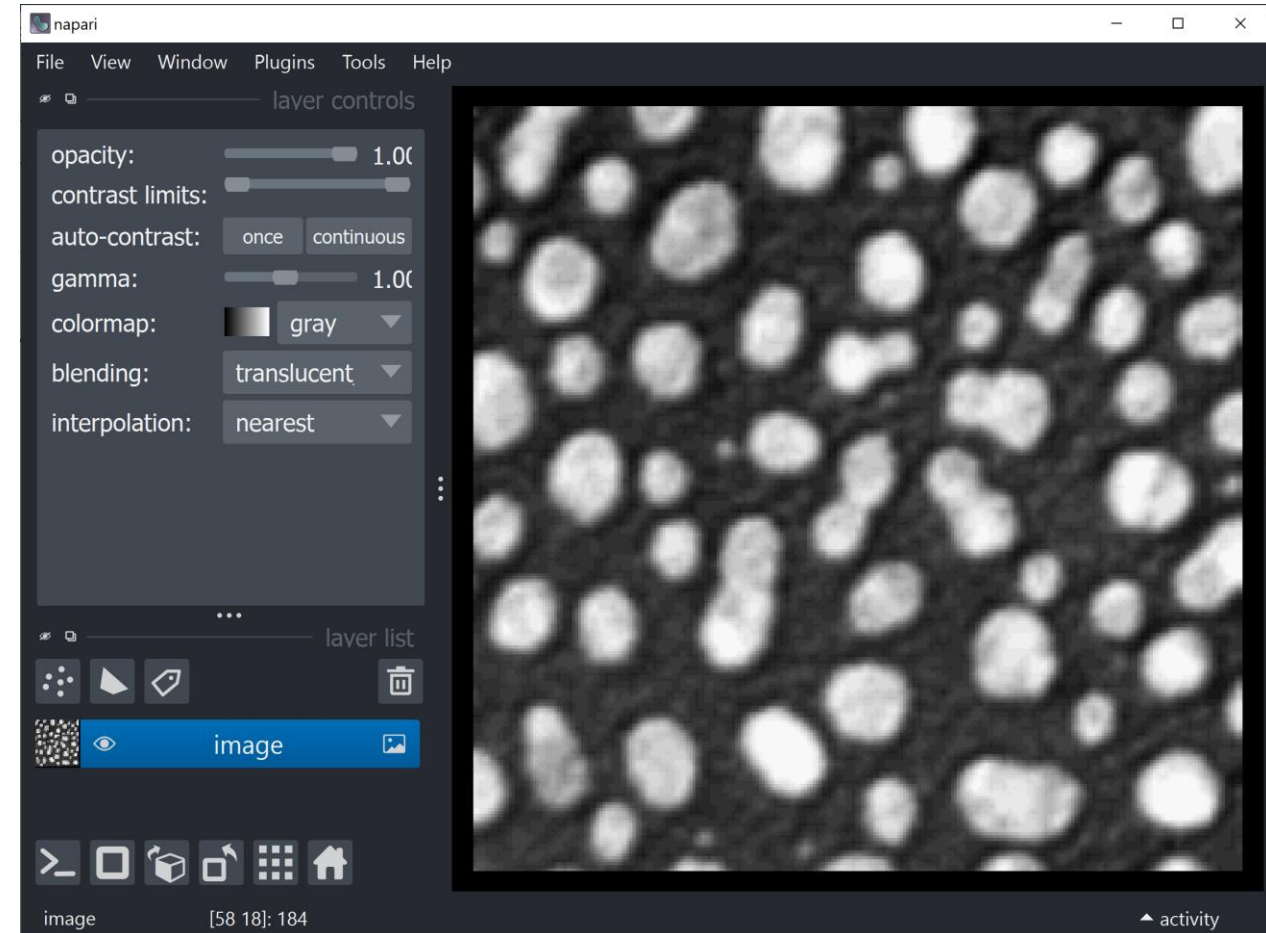


- Drag and drop
- File > Open File...
- From code:
  - Open an image with an “imread” function

```
image = imread("image.tif")
```

- Load the image to the viewer

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





# Scripting napari in notebooks

- Add layers to napari to visualize intermediate processing results on top of each other or side by side.
- Change layer visualization within napari...

... or via code in a jupyter notebook:

```
viewer.layers[0].contrast_limits
```

```
[0, 255]
```

1. Access the viewer

2. Access the layers

3. Choose a layer (by index or name)

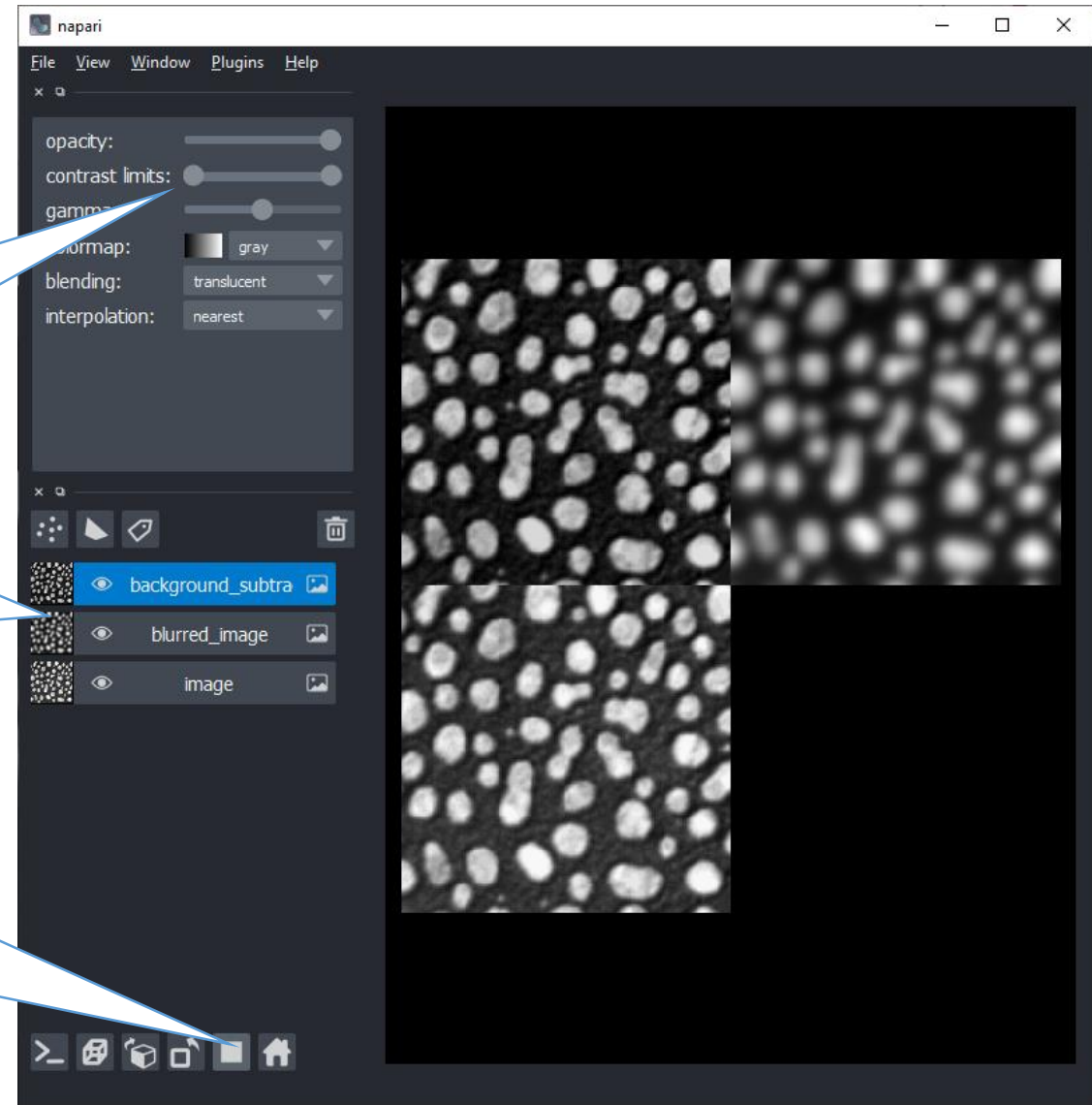
4. "Press TAB" and check out available properties

```
viewer.layers[0].contrast_limits = [30,170]
```

Change Brightness and contrast here

Toggle visualization of layers here

Gallery view



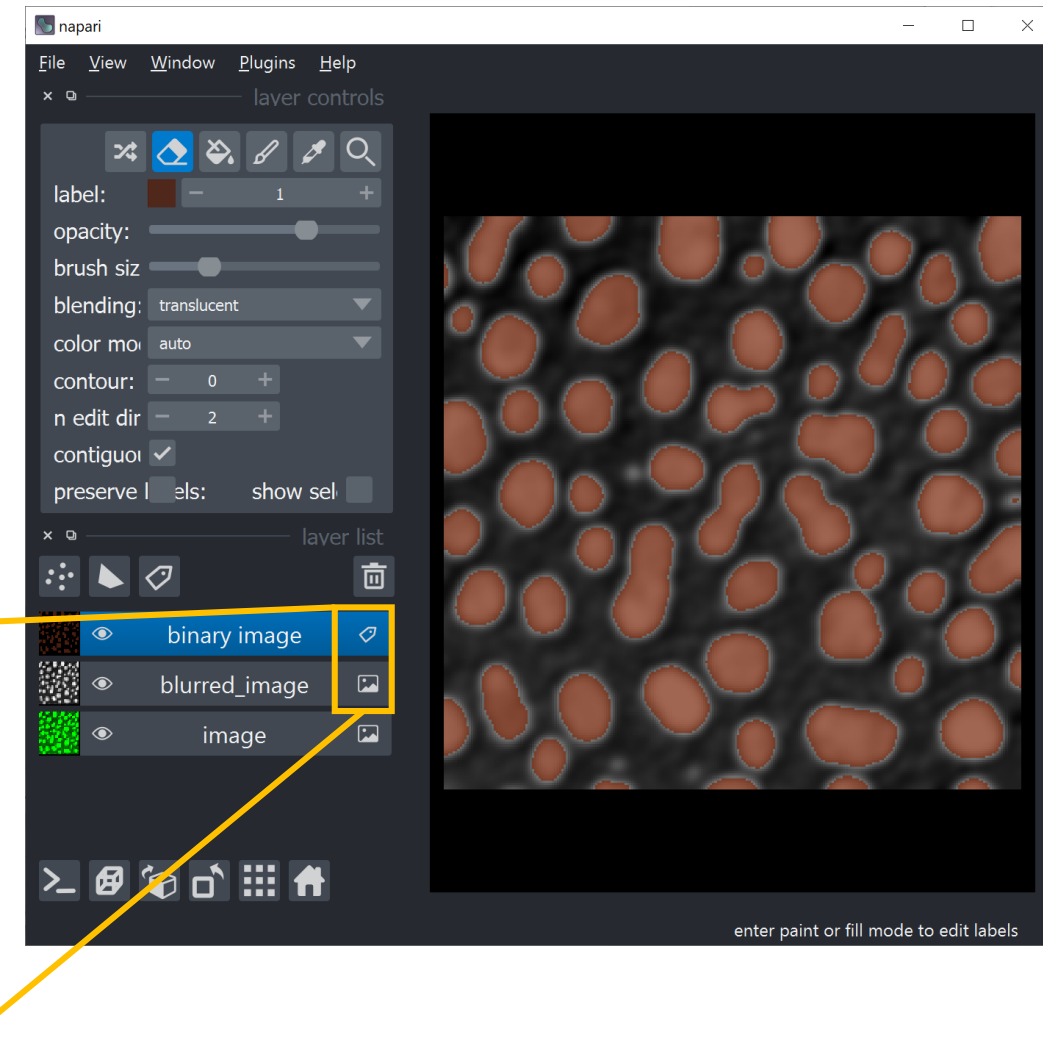
# Visualizing image segmentation

- Binary images and `label` images visualized as label layers
- # 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

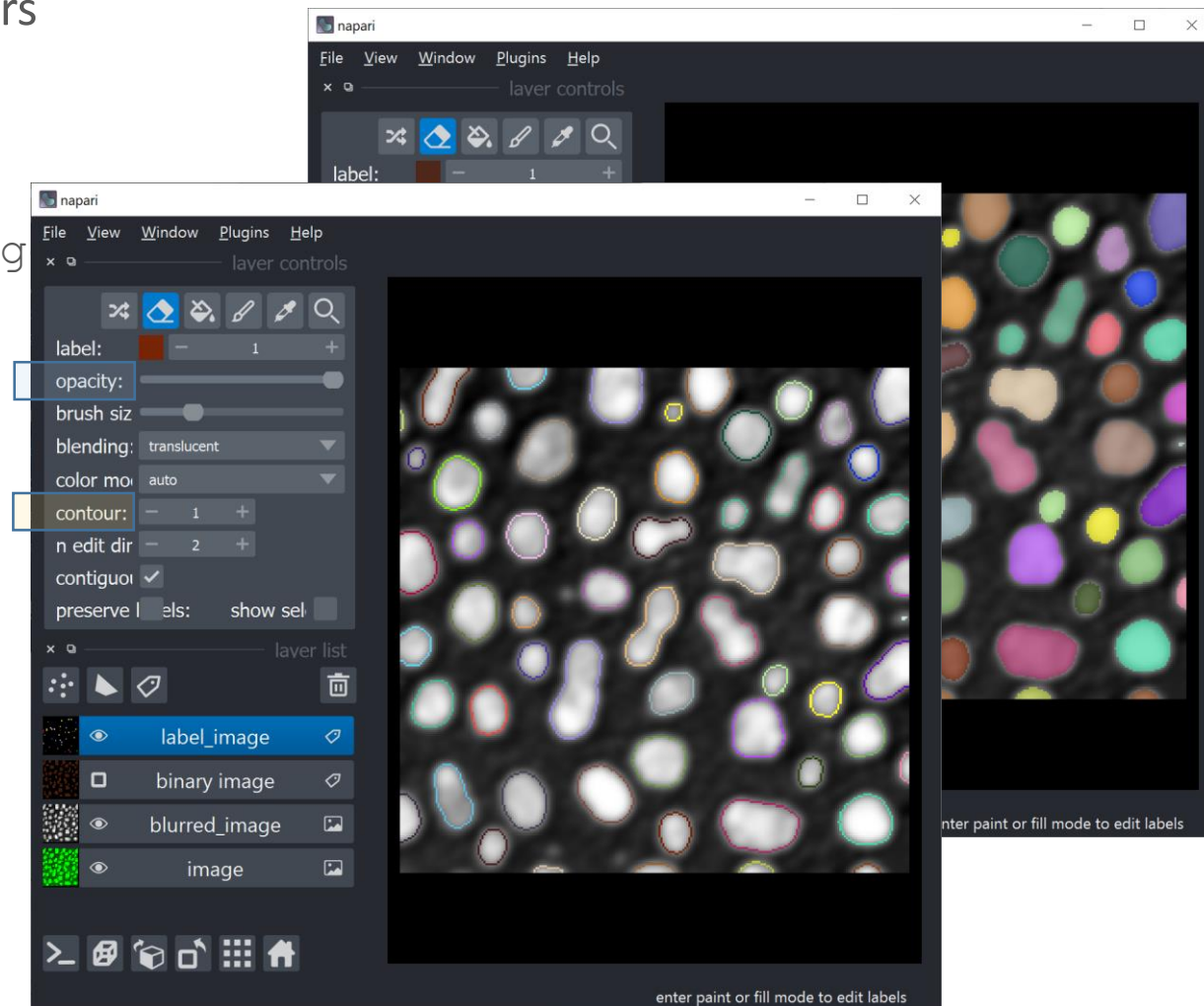
Labels Layer

Image Layer



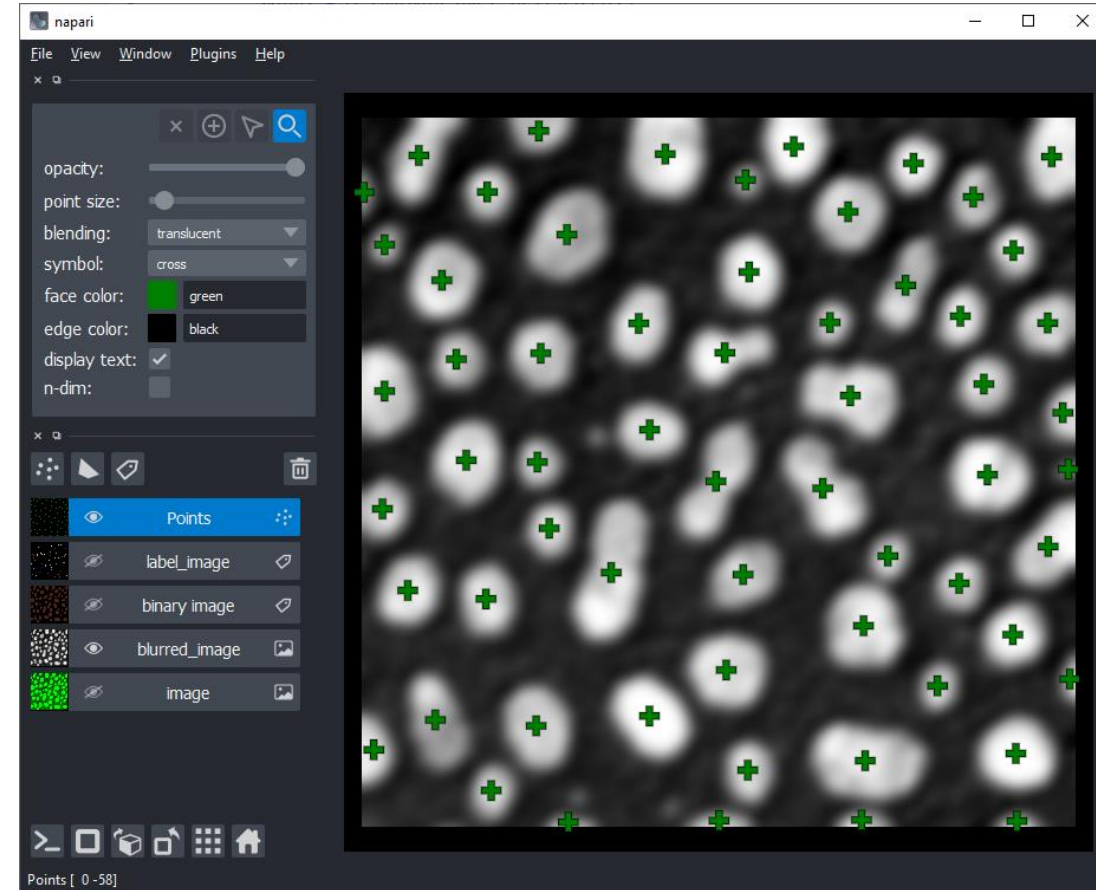
# Visualizing image segmentation

- Binary images and label images visualized as label layers
- `# 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`



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

```
# add points to viewer  
label_layer = viewer.add_points(points,  
    face_color='green', symbol='cross', size=5)
```





# 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

