

Napari

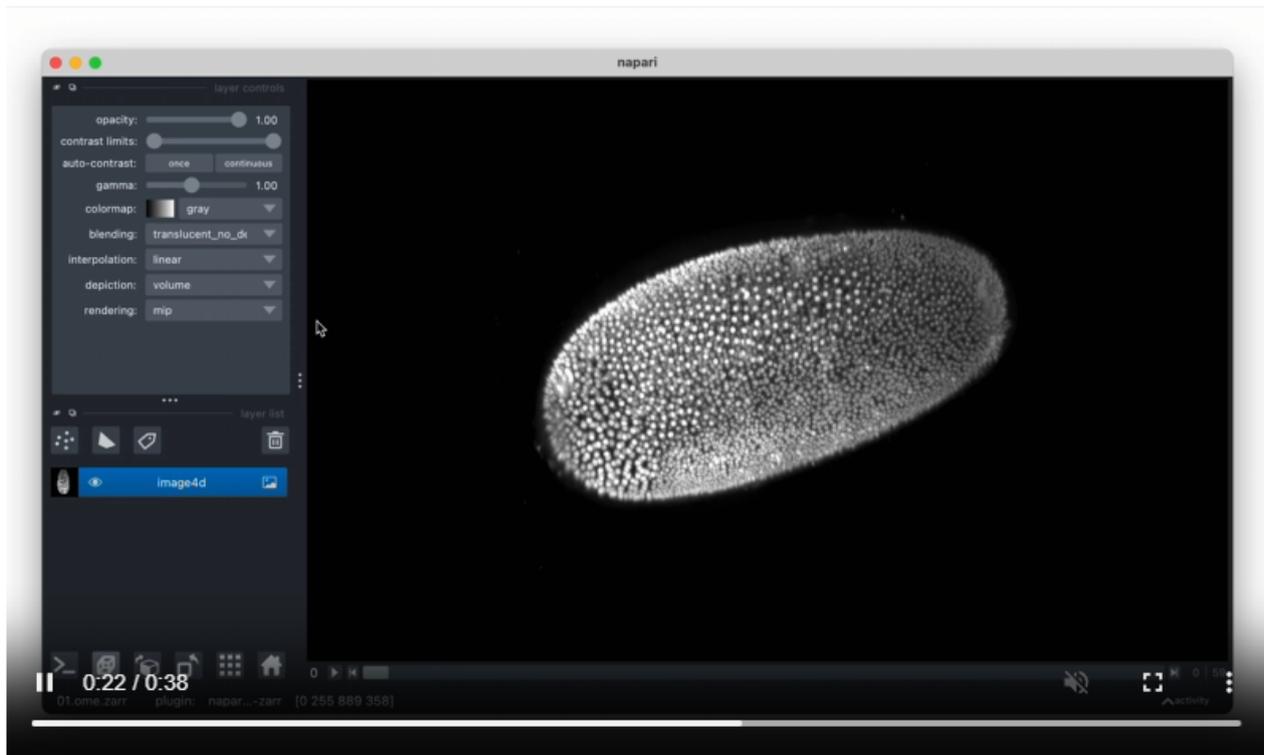
- [Napari: a fast, interactive viewer for multi-dimensional images in Python](#)

Napari: a fast, interactive viewer for multi-dimensional images in Python

If you are using Python for image analysis already, Napari is a great package to help you visualize and render your data!

Documentation and downloads can be found on [this page](#).

Below is an example of 3D rendering in Napari.



```
napari.imshow(image4d)
```

Napari offers a lot of usage options such as the ones listed below.



🔍 search

Ctrl K

- ▼ Getting started
- ^ napari tutorials
 - ▼ Annotation
 - ▼ Processing
 - ▼ Segmentation
 - ▼ Tracking
- ^ How-to guides
 - ▼ Using layers
 - ▼ Extending napari
 - napari + ImageJ how-to guide
 - Napari in Docker
 - Performance monitoring
 - Running napari headlessly
 - Creating and testing themes
- ▼ In-depth explanations
 - Glossary
 - napari workshops
 - Sample databases
 - Troubleshooting
- ▼ Gallery
- ▼ Release notes

The Gallery gives examples of what you can do in terms of basic and advanced visualization. Check out the examples and see if you can use some of these tools to make awesome images and renderings for your papers and talks!

^ visualization-basic (22)

Add grayscale image
Add image
Add image transformed
Add labels
Add points
Add points with features
Add points with multicolor text
Add points with text
Add shapes
Add shapes with features
Add shapes with text
Add surface 2D
Add vectors
Add vectors image
Image depth
Labels 2D
Layers
Minimum blending
Pass colormaps
Point cloud
Set colormaps

visualization-advanced (17)

3D Paths
3D image plane rendering
Add multiscale image
Add vectors color by angle
Affine transforms
Dynamic projections dask
Export Figure
Export regions of interest (ROIs) to png
nD multiscale image
nD multiscale image non-uniform
Points over time
Comparison of Screenshot and Figure Export
Show points based on feature
To screenshot
Tracks 3D
Tracks 3D with graph
Visualizing optical flow in napari