napari: a Python Multi-Dimensional Image Viewer Platform for the Research Community

Chi-Li Chiu[1*], Nathan Clack[1], and the napari community[3]

[1.] Chan Zuckerberg Initiative, Redwood City, CA, United States.
[2.] Yikes LLC, Baltimore, MD, United States.
[3.] https://github.com/napari/napari/graphs/contributors
* Corresponding author: c.chiu@chanzuckerberg.com

Advances in imaging technologies such as the development of clearing methods and light sheet microscopy have made it routine for biologists to generate large volume and high dimensional data that are challenging to visualize and analyze [1]. Simultaneously, computational tools and methods for addressing these challenges have grown dramatically, including the expanded machine learning and large data handling libraries in the Scientific Python ecosystem. However, the limited large multi-dimensional image visualization options native in Python hamper researchers' ability to fully leverage these tools effectively.

To address this gap, we are contributing to the development of napari: a fast, interactive, multi-dimensional image viewer, with a vibrant plugin ecosystem that expands its capability to tackle various domain-specific visualization and analysis needs. It is built on Qt [2] for the GUI, vispy [3] for performant GPU-based rendering through OpenGL library, Dask [4] for memory-efficient large data handling, and the scientific Python stack (numpy, scipy, and scikit-image).

napari is a community-driven open-source project on GitHub developed to facilitate transparency, reuse, and extensibility (https://github.com/napari/napari). At its core, it provides critical viewer features out-of-the-box. These include support for large multi-dimensional data through asynchronous rendering, image pyramid, and chunked file formats (Zarr [5]). “Layers” are used to simultaneously visualize images, models, and analysis results enabling manual, interactive annotation in 3D.

In addition, we are aiming for flexible, developer/user-friendly plugin architecture to facilitate technology dissemination, thereby providing biologists easy access to advanced image analysis workflows through a performant image viewer. Squidpy [6] and brainreg [7] are several examples of using napari to handle complex, large datasets.

References:

[1] EHK Stelzer, E.H.K., Strobl, F., Chang, BJ. et al. Nat Rev Methods Primers **1, 73** (2021), https://doi.org/10.1038/s43586-021-00069-4
[2] Qt, https://www.qt.io/ (accessed Feb 16, 2022).
[3] vispy, https://vispy.org/ (accessed Feb 16, 2022).
[4] Dask, https://dask.org/ (accessed Feb 16, 2022).
[5] Zarr, https://zarr.readthedocs.io/en/stable/ (accessed Feb 16, 2022).
[6] Palla, G., Spitzer, H., Klein, M. et al. Nat Methods 19, 171–178 (2022), https://doi.org/10.1038/s41592-021-01358-2

https://doi.org/10.1017/S1431927622006328 Published online by Cambridge University Press

[7] Tyson, A.L., Vélez-Fort, M., Rousseau, C.V. et al. Sci Rep 12, 867 (2022), https://doi.org/10.1038/s41598-021-04676-9

https://doi.org/10.1017/S1431927622006328 Published online by Cambridge University Press