速習 napari ―基礎知識と書き方

* 20 pages
* Before my article: Learning Jython
* After my article: practical Jython – (ImageJ + Python)

## Introduction (序論)

* Napari is a visualization interface build up with the python language, aiming for multi-dimensional image data visualization… In the recent years many new development of the package on python… for example, pandas for tabular data processing, network for network analysis and scikit-learn and pytorch for ML/ DL implementation and is becoming one of the most powerful programming language … application of image analysis on python would take the advantage of …
* Napari is able to help the visualization of multiple data type. Other then its native multi-dimensional data visualization ability, it is able to visualize labels, tracks and particles …
* Many “plugins” were built on top of napari, which makes napari become a platform for image analysis tool development, just like fiji-imageJ, but in the python language …
* Recent development such as the Napari-assistant and the chat-GPT implemented Napari also makes Napari a powerful tool for bioimage analysis
* In this article, I will explain how to get started with Napari, do some simple analysis with python-Napari just like what we did in the last chapter. Then we will try a few visualization options of napari. Finally I will go through how to install plugins on Napari, with a brief introductions on some widely useful plugins.

## The first step

* Install a Python Environment for Napari
  + Do pip install everything
  + Reference: Beginners’ guide for python command line: what is virtual environment? How does the pip install work? Cd? Mkdir?
* Open an image in Napari:
* Two options:
  + Drag and drop an image (try blob-2D then image 3D) and explain the display options on the panel *(3 pages on this)*
  + call the image in commands (“But it would be nice to show the image programmatically
    - Call Napari from Jupyter notebook
      * “… in the previous chapter all the python scripts were written in the Fiji script editor, however for Napari and any other python programming, it is recommended to use a Python IDE for scripting and testing …”
      * Introduction to Jupyter …
      * “ … for Napari, and for most scientific programming, I recommend using a Jupyter notebook as a coding interface. Jupyter notebook is a coding environment where you could have the code and the documentation (with the markdown language… introduce what is the markdown language) at the same place, therefore encourage a careful documentation of the code. Throughout this article, we will use the Jupyter notebook as the IDE …”
  + おまけ: Learning other methods to display images in Jupyter: matplotlib.pyplot.plot… imshow…

## Layers concept in Napari

* Image object is added to Napari in layers
* Image is just one of the few possible properties of layers… there are many layers to be added…
  + Image Layer
  + Labels Layer
  + Points Layer
  + Shape Layer
  + Surface Layer
  + Track Layer
  + Vector Layer
* Getting access to the layers via scripting

## Install plugins to Napari

* What are plugins and where do they live?
* Some useful plugins
  + Napari-assistant

Something to ask Miura san

* I saw that for the 速習 Jython ―基礎知識と書き方 article he wants to reuse the article on github – would he keep the python tutorials?
* Any useful Napari plugins he think would be nice to mention?
* I am thinking about how to make this article connect to the later chapters. For example, for the “layers” introductions, I will use the images from the later chapters. Any other ideas?
* Are any later chapters using Napari as the processing tools?
* I want to make this article more than a bare translational work. – any ideas how to?

Hello, I hope you are doing well! I have a few questions about the Napari articles that I am writing, I wonder if I can get some thoughts/comments from you:

1. I saw in the plan that you are thinking of using the article you put on your github as a model for the *速習 Jython ―基礎知識と書き方* article, and I am doing my writing with an assumption that the readers will read your article if they have something unclear about python. I wonder would you change any parts of that Jython article? Any parts you would remove/ add? Thanks.
2. I am thinking about how to make my article connect well with the earlier/ later chapters. One of my ideas to do so is to use the images from the later chapters (for example, those in 型2 and 型3 on tracking) to demonstrate the track layers of napari. And for the label layer, I can use the example image from your chapter (核膜article) as an example. Any other ideas/ suggestions?
3. Are there any other chapters using Napari as the visualization tools? And is Jupyter notebook being used in any later chapters? I am thinking to introduce Jupyter in my chapter.
4. I am thinking to introduce some of the useful Napari plugins in my article. For example, Napari-assistance. Any other Napari plugins that you think more people should know?