cellpose-napari

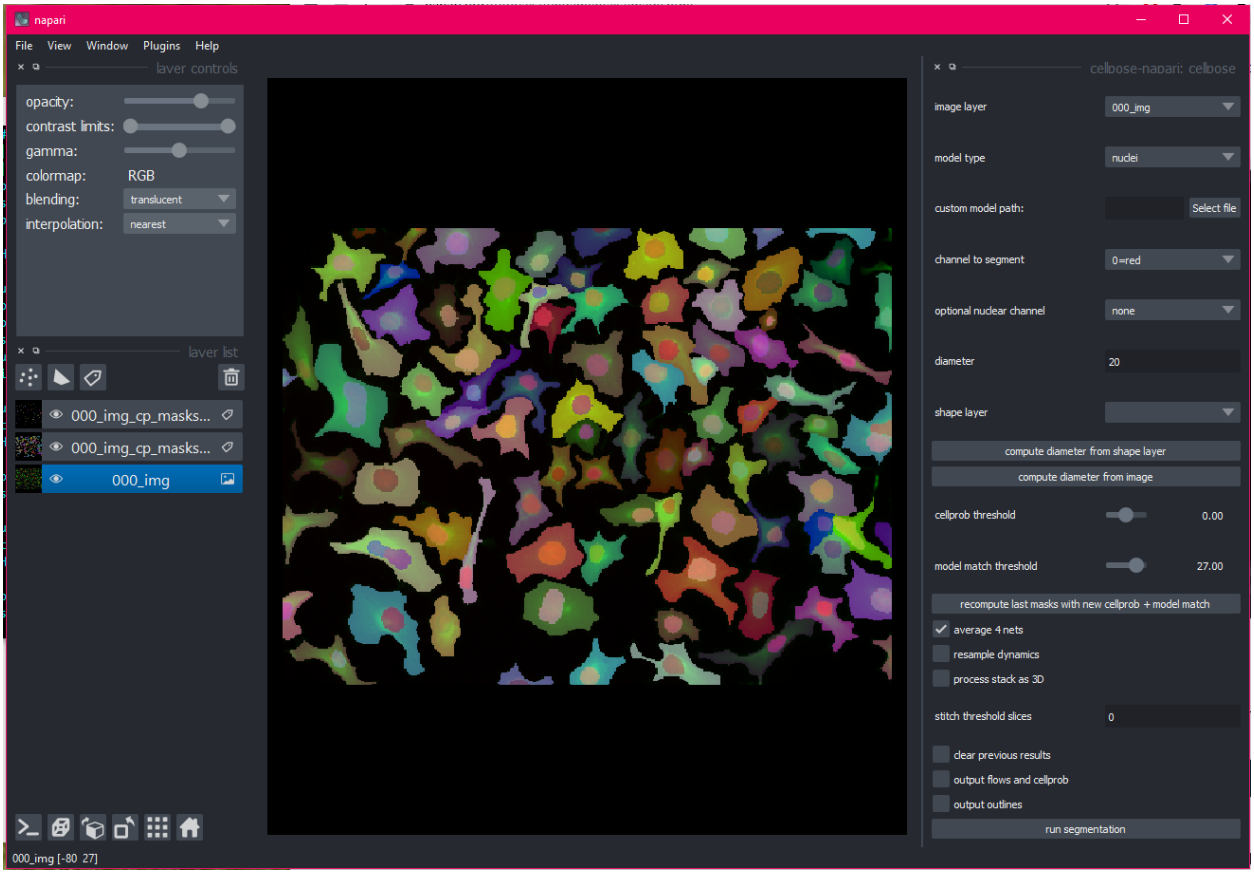
Release 0.1

Carsen Stringer & Marius Pachitariu

Nov 04, 2021

BASICS:

cellpose-napari is a plugin to enable usage of cellpose in napari.



Please see cellpose documentation for more information on the algorithm and the settings.

If you use this plugin please cite

```
@article{stringer2021cellpose,
title={Cellpose: a generalist algorithm for cellular segmentation},
author={Stringer, Carsen and Wang, Tim and Michaelos, Michalis and Pachitariu, Marius},
journal={Nature Methods},
volume={18},
number={1},
pages={100--106},
year={2021},
publisher={Nature Publishing Group}
}
```

CHAPTER

ONE

INSTALLATION

For full napari install instructions, look at their install instructions.

For full cellpose install instructions, look up the main cellpose github readme.

1.1 Common issues

If you receive the error: `Illegal instruction (core dumped)`, then likely mxnet does not recognize your MKL version. Please uninstall and reinstall mxnet without mkl:

```
pip uninstall mxnet-mkl
pip uninstall mxnet
pip install mxnet==1.4.0
```

If you receive the error: `No module named PyQt5.sip`, then try uninstalling and reinstalling pyqt5

```
pip uninstall pyqt5 pyqt5-tools
pip install pyqt5 pyqt5-tools pyqt5.sip
```

If you have errors related to OpenMP and libiomp5, then try

:: conda install nomkl

If you receive an error associated with **matplotlib**, try upgrading it:

```
pip install matplotlib --upgrade
```

If you receive the error: `ImportError: _arpack DLL load failed`, then try uninstalling and reinstalling scipy

```
pip uninstall scipy
pip install scipy
```

If you are having issues with the graphical interface, make sure you have **python 3.7** and not python 3.8 installed.

If you are on Yosemite Mac OS or earlier, PyQt doesn't work and you won't be able to use the graphical interface for cellpose. More recent versions of Mac OS are fine. The software has been heavily tested on Windows 10 and Ubuntu 18.04, and less well tested on Mac OS. Please post an issue if you have installation problems.

CHAPTER

TWO

RUNNING PLUGIN

You can drag and drop a variety of images into napari. You can also open a folder of images to process together (or sequentially). See napari image documentation for more advanced image loading.

2.1 2D images

Here is an example of the plugin running on the 2D sample image.

2.2 Batch processing in 2D

Cellpose will process all images in an opened folder with the same settings and return masks for each image with the name `folder_name_fileid`. You can export all the images in the napari File menu.

Here is an example of the plugin running on a folder of 2D data.

2.3 3D segmentation

Cellpose can segment 3D and 4D stacks. If there is a channel axis, specify it by labelling it with 'c'.

To run 3D processing, check `process stack as 3D`.

If the 3D segmentation is not working well and there is inhomogeneity in Z, try stitching masks using the `stitch threshold slices` option instead of checking `process stack as 3D`. In this setting, cellpose will create masks in 2D on each XY slice and then stitch them across slices if the IoU between the mask on the current slice and the next slice is greater than or equal to the `stitch threshold slices`.

Here is an example of the plugin running on the 3D sample image.

2.4 Time segmentation

Stitching can also be applied in time by setting `stitch threshold slices`>0.

Here is an example of the plugin running on cell tracking challenge data.

CHAPTER

THREE

SETTINGS

There are more settings for cellpose that can be accessed using the CLI or through a jupyter notebook. See details at cellpose docs.

Listed are settings available through napari widget. Please submit an issue if you would like a new setting available in the widget.

3.1 Channels

3.1.1 Cytoplasm model (*'cyto'*)

The cytoplasm model in cellpose is trained on two-channel images, where the first channel is the channel to segment, and the second channel is an optional nuclear channel. Here are the options for each: 1. 0=grayscale, 1=red, 2=green, 3=blue, 4 ... 2. 0=None (will set to zero), 1=red, 2=green, 3=blue, 4 ...

3.1.2 Nucleus model (*'nuclei'*)

The nuclear model in cellpose is trained on two-channel images, where the first channel is the channel to segment, and the second channel is always set to an array of zeros. Therefore set the first channel as 0=grayscale, 1=red, 2=green, 3=blue; and set the second channel to 0=None

3.2 Diameter

The cellpose models have been trained on images which were rescaled to all have the same diameter (30 pixels in the case of the *cyto* model and 17 pixels in the case of the *nuclei* model). Therefore, cellpose needs a user-defined cell diameter (in pixels) as input, or to estimate the object size of an image-by-image basis.

The automated estimation of the diameter is a two-step process using the *style* vector from the network, a 64-dimensional summary of the input image. We trained a linear regression model to predict the size of objects from these style vectors on the training data. On a new image the procedure is as follows.

1. Run the image through the cellpose network and obtain the style vector. Predict the size using the linear regression model from the style vector.
2. Resize the image based on the predicted size and run cellpose again, and produce masks. Take the final estimated size as the median diameter of the predicted masks.

Click `compute diameter from image` to start diameter estimation. However, if this estimate is incorrect please set the diameter by hand.

You can also get an estimate from clicking on the image: create a napari 'Shapes' layer and draw circles or squares. If you click `compute diameter from shape layer`, the plugin will set the diameter to the average diameter of the drawn shapes.

Changing the diameter will change the results that the algorithm outputs. When the diameter is set smaller than the true size then cellpose may over-split cells. Similarly, if the diameter is set too big then cellpose may over-merge cells.

3.3 Resample

The cellpose network is run on your rescaled image – where the rescaling factor is determined by the diameter you input (or determined automatically as above). For instance, if you have an image with 60 pixel diameter cells, the rescaling factor is 30./60. = 0.5. After determining the flows (dX, dY, cellprob), the model runs the dynamics. The dynamics can be run at the rescaled size (`resample=False`), or the dynamics can be run on the resampled, interpolated flows at the true image size (`resample=True`). `resample=True` will create smoother masks when the cells are large but will be slower in case; `resample=False` will find more masks when the cells are small but will be slower in this case.

3.4 Model match threshold

Note there is nothing keeping the neural network from predicting horizontal and vertical flows that do not correspond to any real shapes at all. In practice, most predicted flows are consistent with real shapes, because the network was only trained on image flows that are consistent with real shapes, but sometimes when the network is uncertain it may output inconsistent flows. To check that the recovered shapes after the flow dynamics step are consistent with real masks, we recompute the flow gradients for these putative predicted masks, and compute the mean squared error between them and the flows predicted by the network.

The `model match threshold` parameter is inverse of the maximum allowed error of the flows for each mask. Decrease this threshold if cellpose is not returning as many masks as you'd expect. Similarly, increase this threshold if cellpose is returning too many ill-shaped masks.

3.5 Cell probability threshold

The network predicts 3 outputs: flows in X, flows in Y, and cell "probability". The predictions the network makes of the probability are the inputs to a sigmoid centered at zero (1 / (1 + e^-x)), so they vary from around -6 to +6. The pixels greater than the `cellprob_threshold` are used to run dynamics and determine masks. The default is `cellprob_threshold=0.0`. Decrease this threshold if cellpose is not returning as many masks as you'd expect. Similarly, increase this threshold if cellpose is returning too masks particularly from dim areas.