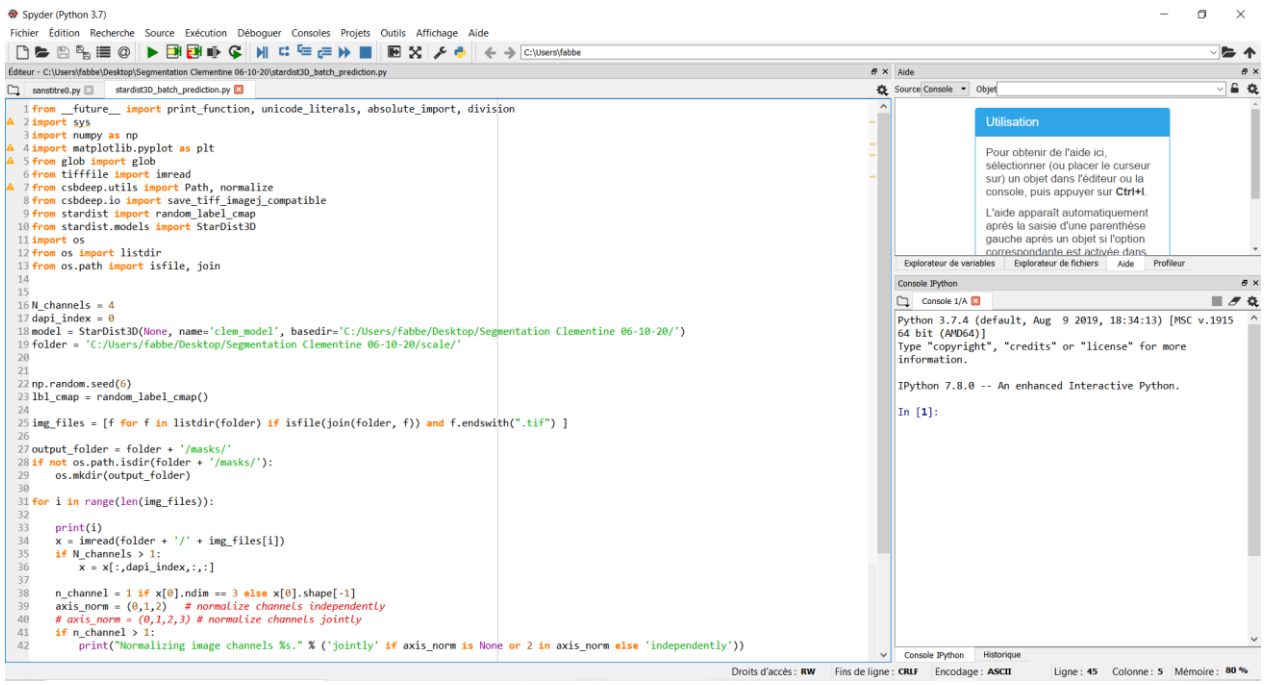
Tutorial:

How to segment 3D images with StarDist

If you haven't installed Python yet please first go to the tutorial *how to install python for StarDist*

1) Open Spyder with the environment you have created
2) Load the script stardist3D_batch_prediction.py

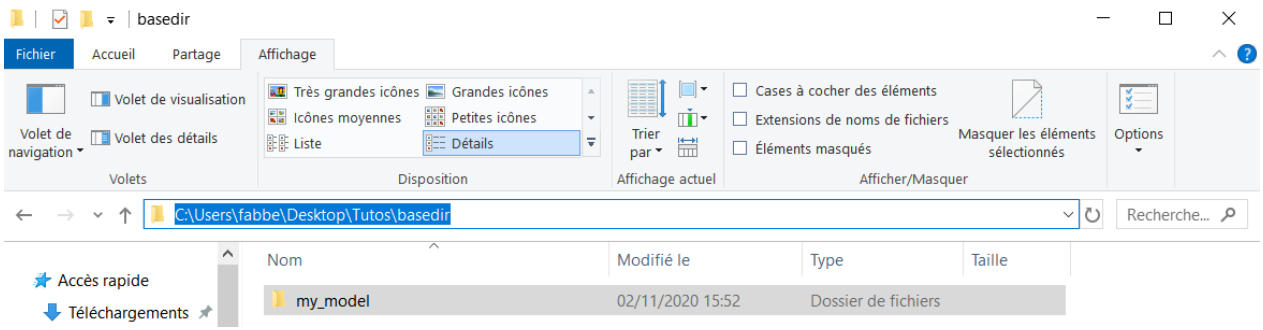
You should see something like this:



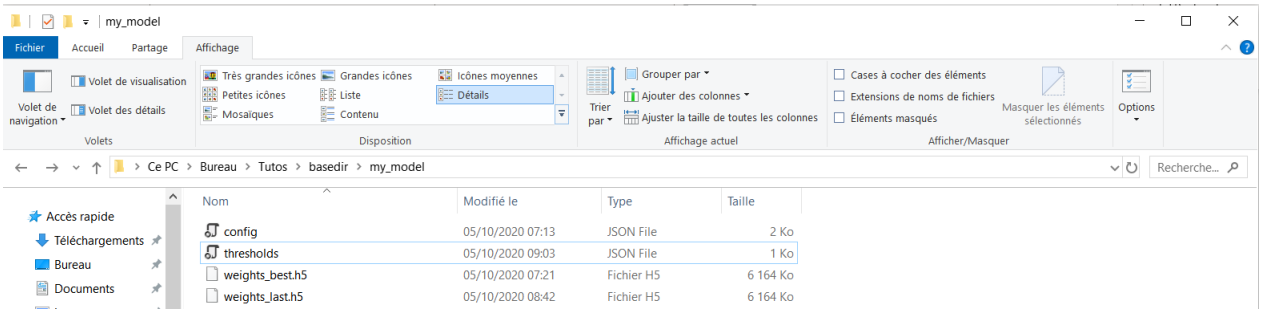
3) For the segmentation you can use your original images with all channels, set how many channels have your images on line 16.
4) Set which channel is the DAPI on line 17. In Python the first index starts at 0. If your DAPI channel is the first channel in imageJ, set 0 in Python.
5) Edit the folder path of your StarDist model on line 18.

We have already several models in the lab, you should try them first. Otherwise you can go to the tutorial *How to generate 3D training samples using LabKit to train a StarDist model* first and then *How to train my model using StarDist* and come back here after!

Basedir is the directory where your model folder is located
name is the folder name of your model



Your model folder should have the following files:



Always use the '/' separator when specifying a folder Path and ends your path with a separator.

6) To edit the thresholds file, right click and open the file with a text-editor such as notepad

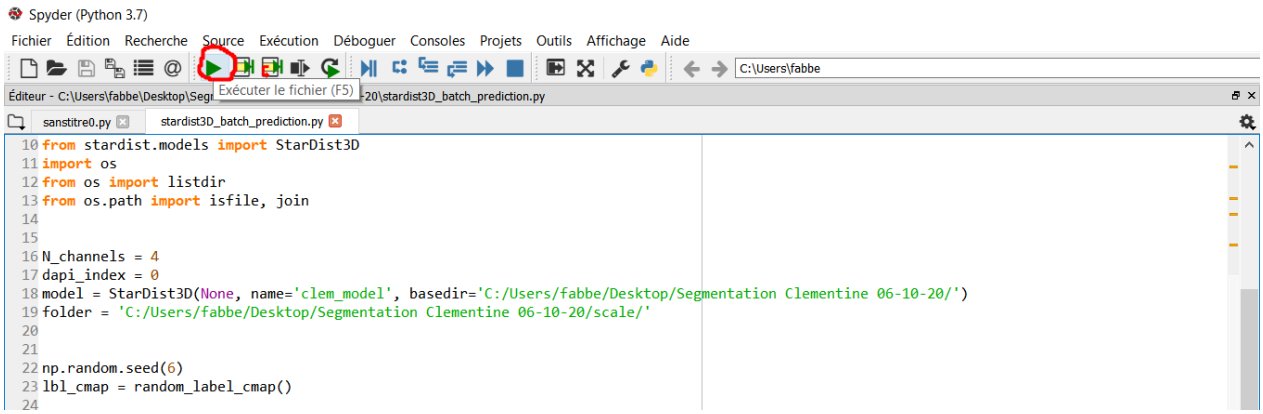
The first threshold parameter "prob" controls the object probability.
The second threshold parameter "nms" controls the overlapping between nucleis.

Go the paper for more details about those parameters:
https://openaccess.thecvf.com/content_WACV_2020/papers/Weigert_Star-convex_Polyhedra_for_3D_Object_Detection_and_Segmentation_in_Microscopy_WACV_2020_paper.pdf

7) Edit the folder path on line 19

Always use the '/' separator when specifying a folder Path and ends your path with a separator.

8) Click the Run button as shown bellow



9) If you get the error as shown below, please increase the number of tiles to [1,9,9] on line 46 n_tiles=[1,3,3] error and re-run StarDist.

```
LAPTOP-A2EGBGGG
2020□□ 16:21:36.871353: I
tensorflow/core/platform/cpu_feature_guard.cc:142] Your CPU
supports instructions that this TensorFlow binary was not compiled
to use: AVX2
2020□□ 16:21:37.838906: W
tensorflow/core/framework/cpu_allocator_impl.cc:81] Allocation of
134217728 exceeds 10% of system memory.
2020□□ 16:21:38.572973: W
tensorflow/core/framework/cpu_allocator_impl.cc:81] Allocation of
134217728 exceeds 10% of system memory.
2020□□ 16:21:44.524620: W
tensorflow/core/framework/cpu_allocator_impl.cc:81] Allocation of
134217728 exceeds 10% of system memory.
2020□□ 16:21:45.372287: W
tensorflow/core/framework/cpu_allocator_impl.cc:81] Allocation of
201326592 exceeds 10% of system memory.
2020□□ 16:21:45.736931: W
tensorflow/core/framework/cpu_allocator_impl.cc:81] Allocation of
134217728 exceeds 10% of system memory.
```

If you get the “kernel has died” error when re-running please close and restart StarDist.

10) If no errors came you should see something like that:

```
0
Normalizing image channels jointly.
WARNING:tensorflow:From C:\Users\fabbe\Anaconda3\lib\site-packages\keras\backend
\tensorflow_backend.py:422: The name tf.global_variables is deprecated. Please use
tf.compat.v1.global_variables instead.

100%|██████████| 9/9 [01:07<00:00,  7.48s/it]
1
Normalizing image channels jointly.
100%|██████████| 9/9 [00:48<00:00,  5.35s/it]
2
Normalizing image channels jointly.
 78%|███████   | 7/9 [01:08<00:21, 10.85s/it]
```

If the program is stuck for more than 10minutes on the first image (as shown below) please go to step 14 and consider resizing your images or train a new model.

```
0
Normalizing image channels jointly.
WARNING:tensorflow:From C:\Users\fabbe\Anaconda3\lib\site-packages\keras\backend
\tensorflow_backend.py:422: The name tf.global_variables is deprecated. Please use
tf.compat.v1.global_variables instead.

100%|██████████| 9/9 [01:07<00:00,  7.48s/it]
```

11) Once finished go to your image folder, you should see a “masks” folder.

12) Check the segmentation result:
    - Open ImageJ and load your DAPI image and your mask.
    - Convert the DAPI image to 16bit
    - Use Color/Merge channel to merge the image and the mask
    - In the Merge channel

Alternatively, you can also use LabKit to inspect the segmentation

13) If you are not happy with the segmentation, you can first try to play with parameters in step 6 and re-run the segmentation.

    If each segmented nuclei look as a bunch of fragmented labels see next step 14.

    If none of them improved the segmentation, consider to train your own model, please go to the tutorial *How to generate 3D training samples using LabKit to train a StarDist model* first and then *How to train my model using StarDist* and come back here after 😊

14) If each segmented nuclei look as a bunch of fragmented labels it means that the model you are using has been trained for smaller objects. In that case one possibility is to resize your images using Image/Scale on imageJ and re-run StarDist using the rescaled images. Otherwise if you need the original resolution for your quantification, then you will need to train a new model from scratch using your dataset.