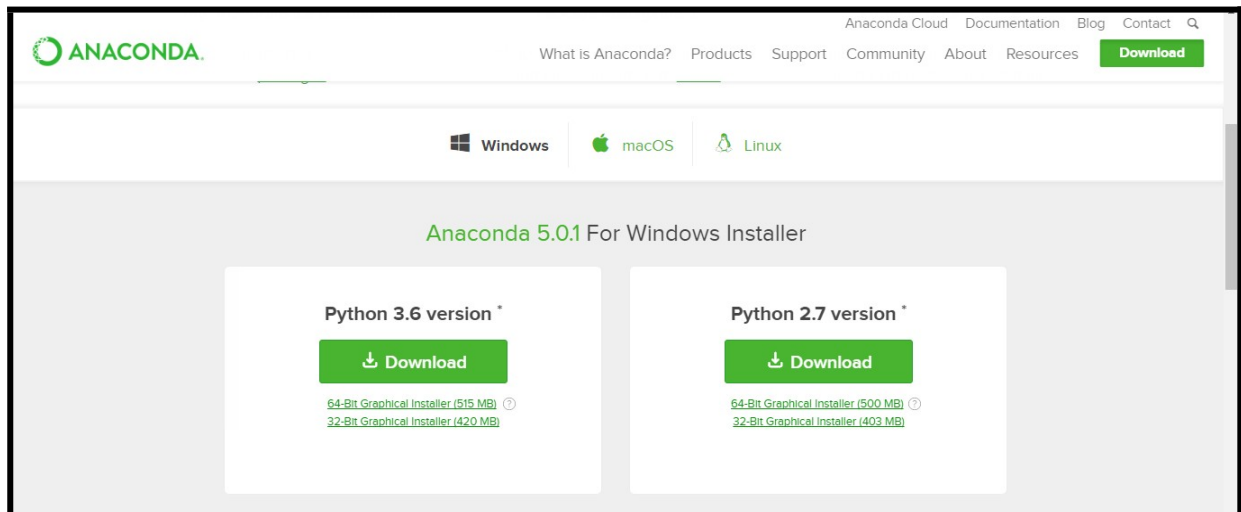# FindMyCells Handbook

WINDOWS

Written by: Tamas Balassa
Last updated: Oct 19th, 2018
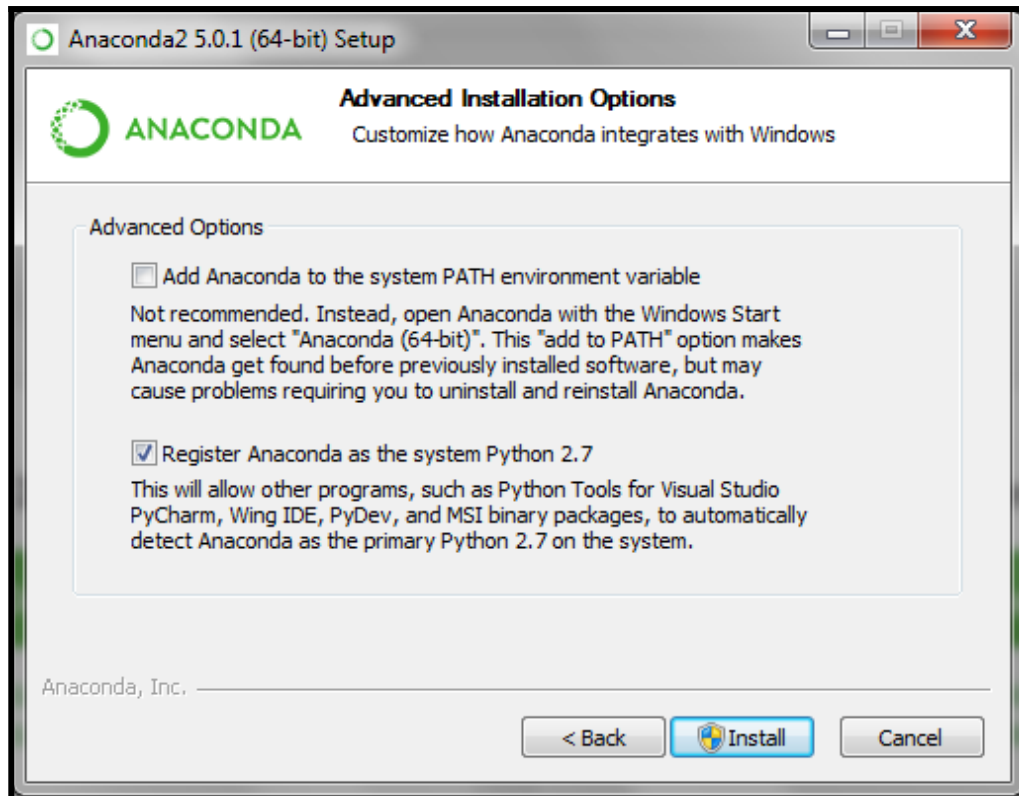
Requirements:

The following software components are required to run the software:
- Recommended installer: *Anaconda* - https://www.anaconda.com/download/ (choose Python 3.6 version graphical installer with the correct type of your platform)

● We recommend to register Anaconda as the system Python:


● Required Python packages:
  ○ Numpy
  ○ Scipy
  ○ PyQt5
  ○ PIL
  ○ OpenCV
  ○ Protobuf
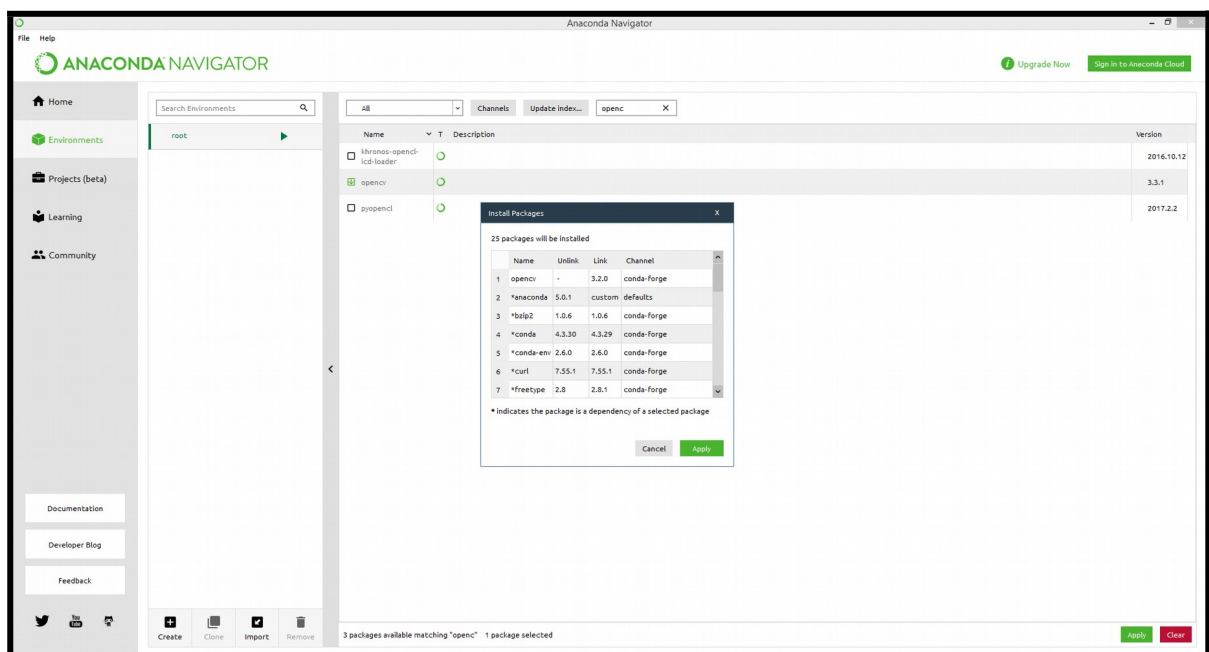 ○ Caffe

Installation:

- To install a package in Anaconda:
  - run Anaconda Navigator as Administrator;
  - select Environments on the left side;
  - (Update Index before searching for the packages);
  - search among 'All' for the package and install it if it's not installed yet;
  - the only exception is the caffe package, see below how to install it

- Notes:
  - for PyQt5 package search 'pyqt'
  - if You search for 'protobuf', 'libprotobuf' package is not neccessary only the exact 'protobuf'
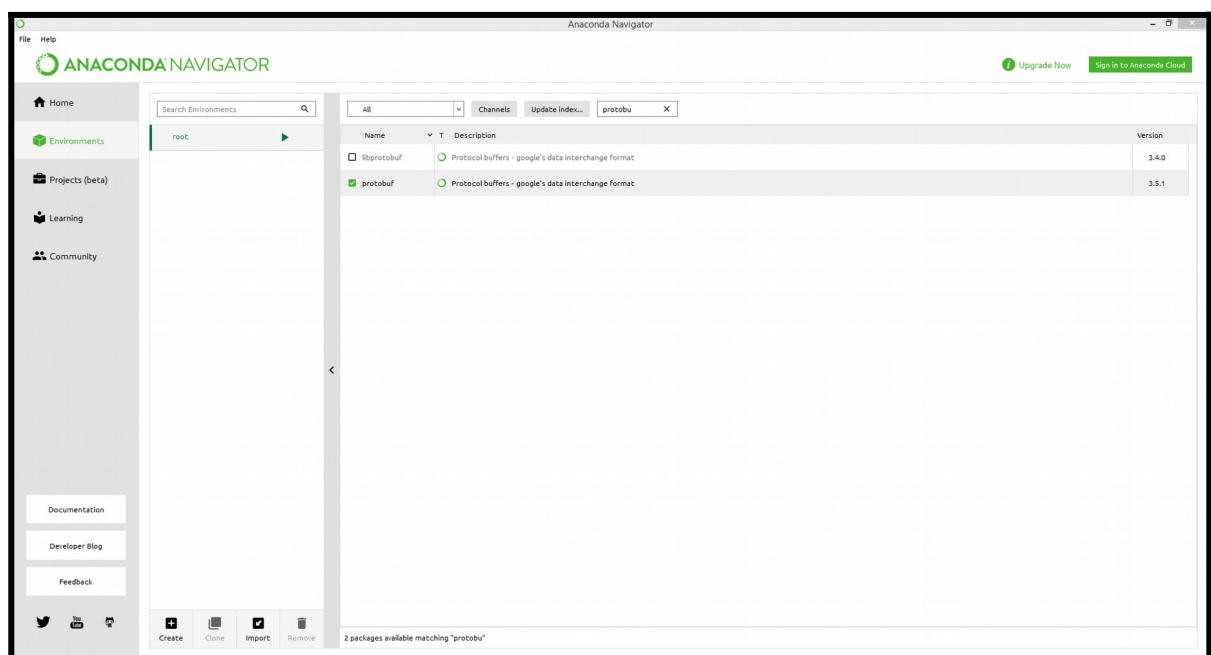
Select 'Environment' on the left side, and select 'All' at the top.
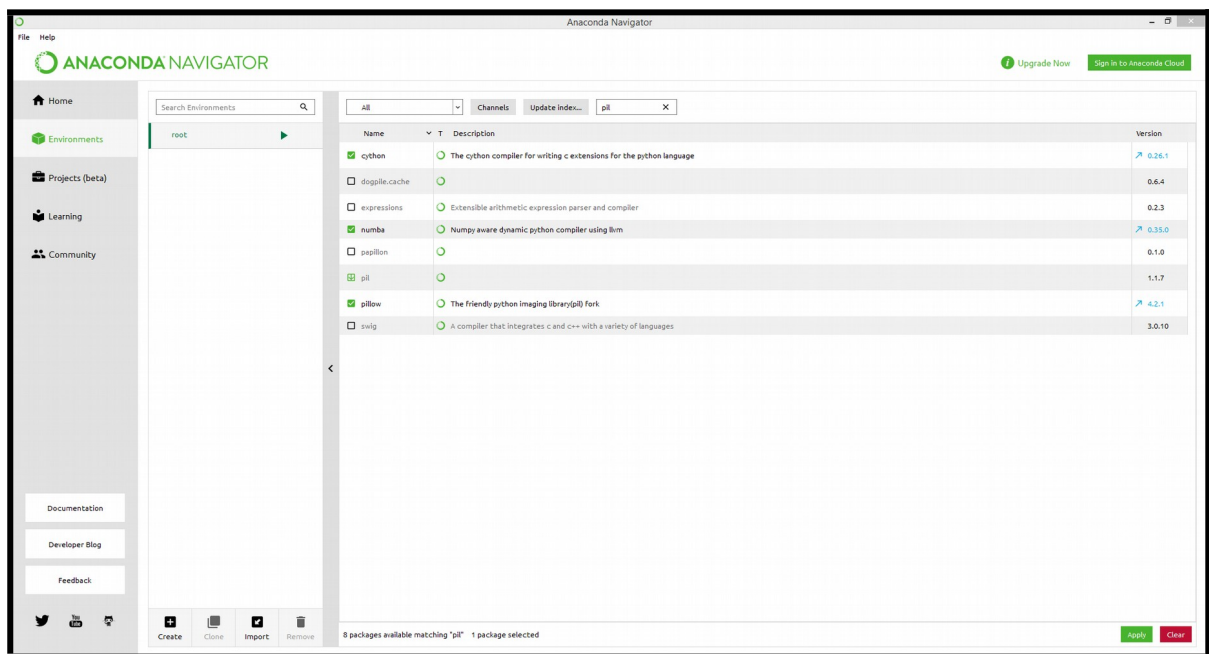
Search for the listed packages. If they are not installed, install them.

Anaconda will detect all the dependencies for each packages, install them too.

- **Caffe package Installation**:
  - You can choose either to build caffe for yourself (advanced way) or You can download an existing built ('prebuilt')[1]
  - If You choose to build for yourself, please visit https://github.com/BVLC/caffe for further instructions
  - You can download prebuilt binaries that are recommended from here: https://github.com/BVLC/caffe/tree/windows (please notice that You will need an installed Visual studio 2015 [recommended] or 2013 - https://www.visualstudio.com/vs/older-downloads/)

    Prebuilt binaries

    Prebuilt binaries can be downloaded from the latest CI build on appveyor for the following configurations:

    - Visual Studio 2015, CPU only, Python 3.5: Caffe Release, Caffe Debug
    - Visual Studio 2015, CUDA 8.0, Python 3.5: Caffe Release
    - Visual Studio 2015, CPU only, Python 2.7: Caffe Release, Caffe Debug
    - Visual Studio 2015,CUDA 8.0, Python 2.7: Caffe Release
   - Visual Studio 2013, CPU only, Python 2.7: Caffe Release, Caffe Debug

  - You will also need to download the *layers* directory from here: http://findmycells.org/uploads/3/4/9/3/34939463/layers.zip and copy it to the `path_to_the_extracted_folder\caffe\python\caffe\` – next to the proto and imagenet directories)

---

1   For GPU prebuilt binaries You will need an installed CUDA. We recommend to try out the CPU versions first.
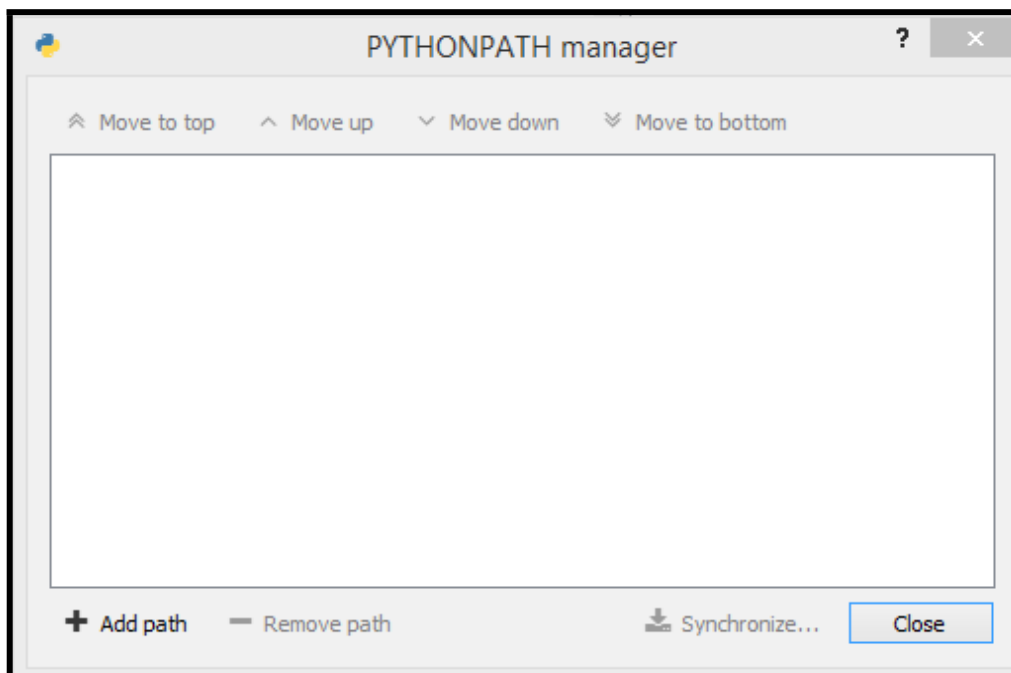
Running the software:
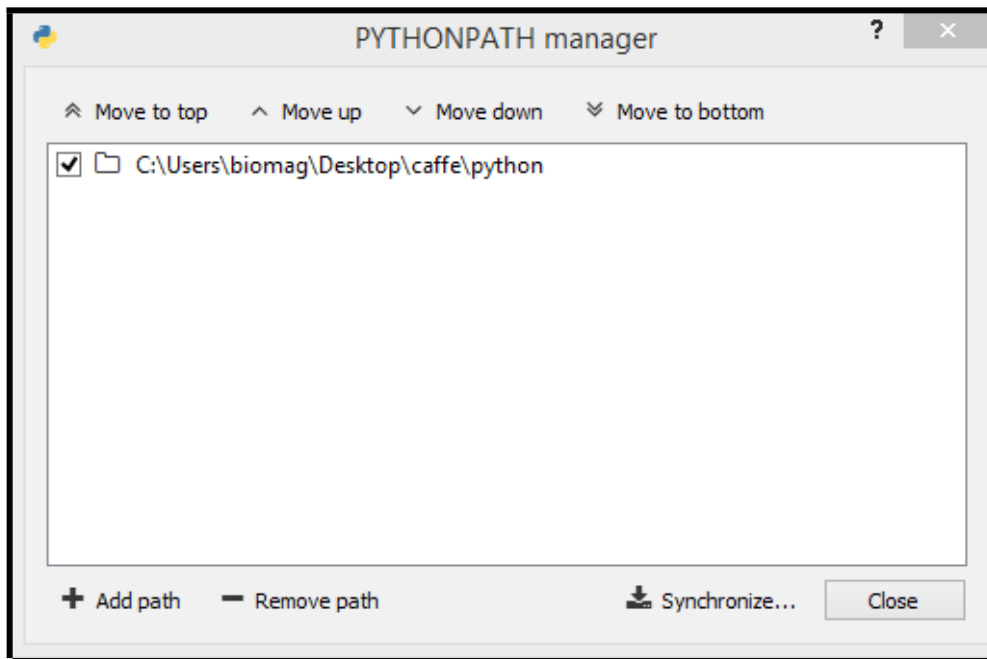
To run the software in Anaconda:

- Launch Spyder in the Anaconda Navigator
- In Tools -> PYTHONPATH manager set the path to the downloaded and extracted folder (`path_to_the_extracted_folder\caffe\python`) – **this need to be set only once**
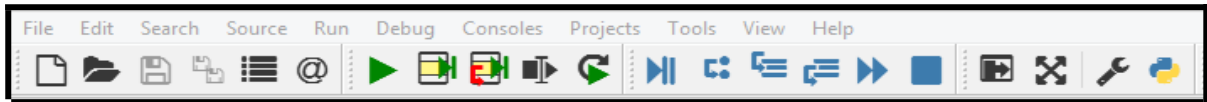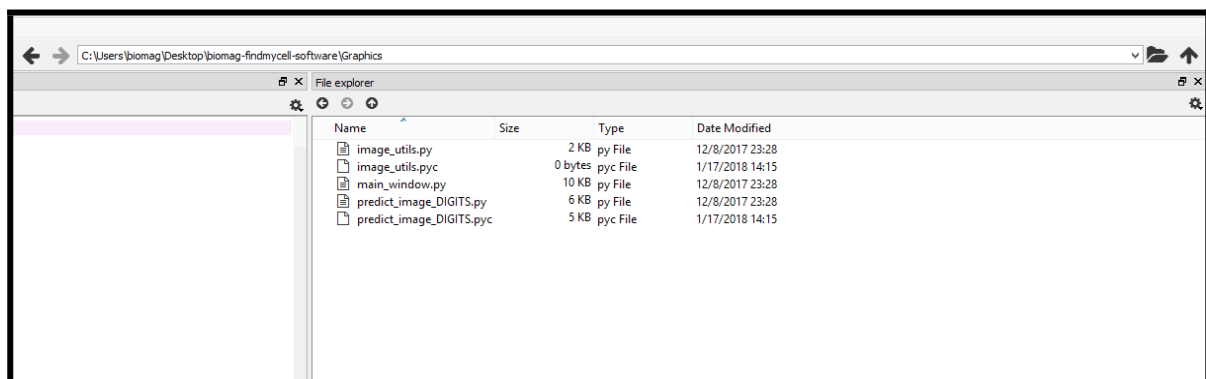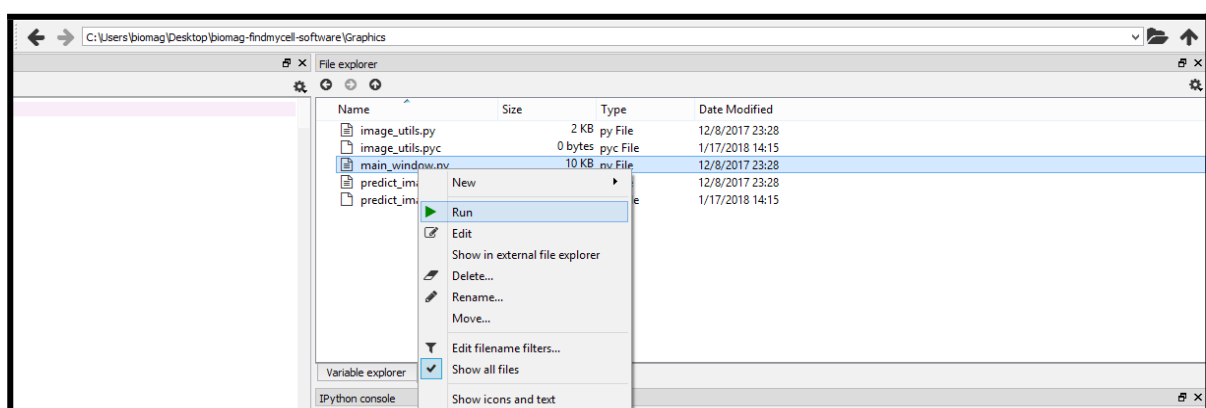

Click on the last icon.

Select 'Add path'

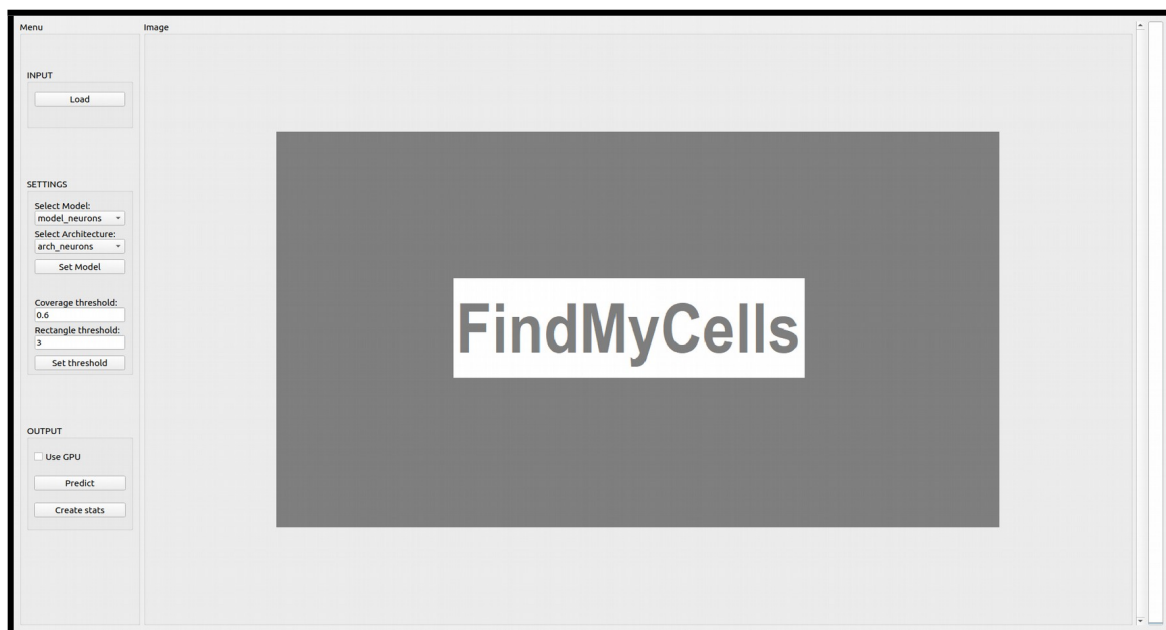Add the `path_to_the_extracted_folder\caffe\python` path.

Navigate to the FMC folder

Run the `graphics/main_window.py` file (right click - run)

User Interface:

Load button:

FindMyCells (FMC) loads in the selected images. The currently supported file formats are: jpg, jpeg, png, tif and tiff. By mouse scrolling the previous/next image will be shown.

Network scrollbars:

FMC uses *caffe*[2] based network model. In case of a new detection task, a trained *caffe* network architecture file (*prototxt*) and model/weights file (*caffemodel*) should be copied into `AI/models/`. These files can then be selected in the dropdown menus.

Coverage threshold:

A real number (between 0 and 1). It defines the threshold for recognizing an object. Larger number results more conservative detection.

Note: A negative threshold value will automatically be set back to the default value (0.6).

Rectangle threshold:

An integer number defining the threshold for the overlapping bounding boxes. Clustering grouped rectangles.

Note: If the number is lower than zero, then it will be set back to the default value (3).

Use GPU checkbox:

Check this box if You are using GPU (*caffe-gpu*), otherwise leave it unchecked. This will handle if the predictions are calculated on cpu or on gpu.

Predict button:

With the predict button the user can predict objects on the current image. Using the middle (scroll) button will reset the bounding boxes.

Output stat:

Creates a statistical report file results.csv in `graphics/` folder.

The fields in this report file contain six values, which are the following:

- Image Name: this filed will contain the full path to the predicted image

---

Caffe is a deep learning framework, made by BAIR - http://caffe.berkeleyvision.org/

- Cell ID: each identified cell has a unique number as ID (it resets at each image)
- X1 & Y1: starting X and Y coordinates of the predicted bounding box
- X2 & Y2: ending X and Y coordinates of the predicted bounding box