mm3 Overview as of October 2016

This is an overview of how to use mm3. Refer to the individual guides referenced in this document for more specific information about usage and function of each script. This guide is current for October 2016, which only includes up until bookkeeping, not actually analyzing cell lives.

mm3 is a set of python scripts designed to facilitate analyzing time-lapse mother machine experiments. This can be thought of in two general tasks which are mostly independent. The first task is the bookkeeping of taking raw data (image files), identifying cell-containing growth channels, and creating TIFF image stacks that contain a single channel. The second task is to take those image stacks and actually identify cells and features to create analyzed data. Currently, there are not official polished scripts for analyzing cell lives. The workflow described below pertains to taking incoming images and creating channel specific data.

Workflow

Basic workflow is as follows:

1. Create experimental folder and choose parameters.
2. Curate input data.
3. Locate channels, create channel stacks, and return metadata (mm3_Compile.py).
4. User guided selection of empty and full channels (mm3_ChannelPicker.py).
5. Subtract phase constrast images (mm3_Subtract.py).

1. Create experimental folder and choose parameters.

mm3 python scripts are run from the terminal and point to a parameter file (.yaml file) that contains all the pertinent information about the experiment. See **Setting up .yaml file** for a guide to the parameters in the .yaml file. The most important information is arguably the paths to the images and where the analyzed images go. See **Guide to folders and files** for more information about the organization of the raw and analyzed data in the experiment folder.

2. Curate input data.

mm3 currently currently takes individual TIFF images as its input. If there are multiple color layers, then each TIFF image should be a stack of planes corresponding to a color. There is a script to convert Nikon Elements .nd2 files into TIFF images of this form. See **mm3_nd2ToTIFF guide** for usage of that script. The quality of your images is important for mm3 to work properly. See **Input images guidelines** for more information.

3. Locate channels, create channel stacks, and return metadata (mm3_Compile.py).

mm3_Compile.py is responsible for the main heavy bookkeeping. It attempts to automatically identify and crop out individual growth channels. Images corresponding to a specific channel are then stacked in time, and these "channel stacks" are the basis of further analysis. If there are multiple colors, a channel stack is made for each color for each channel.

It is also at this time that metadata is drawn from the images and saved. See **mm3_Compile guide** for usage and details.

4. User guided selection of empty and full channels (mm3_ChannelPicker.py).

mm3_Compile.py identifies all growth channels, regardless of if they contain or do not contain cells. mm3_ChannelPicker.py first attempts to guess, and then presents the user with a GUI to decide which channels should be analyzed, which channels shough be ignored, and which channels should be used during subtraction of phase contrast images. See **mm3_ChannelPicker guide** for usage and details.

5. Subtract phase contrast images (mm3_Subtract.py).

Downstream analysis of phase contrast (brightfield) images may require background subtraction to remove artifacts of the PDMS device in the images. See **mm3_Subtract guide**.

Other scripts

mm3 relies on a number of other python modules to run. Some of these are contained in the repository in the folder /external_lib. All the other modules are very standard and should already be installed with your python distribution, especially if you are using Anaconda (https://www.continuum.io/) (recommended). If you get an error and you need to install a module that you do not have, attempt to install it with pip:

```
pip install module_name
```

There is a script to make .mpg4 movies from TIFF files. See **mm3_MovieMaker guide** for usage and details.