Use MERFISH pipeline

Tuesday, 26 March 2019 14:10

All the code is located in the Isogailab shared Dropbox

→ in the folder called CODE

→ subfolder merfish_analysis

Steps

1. <u>Split channels & create maximum intensity projections</u>
    - ○ Using MATLAB: splitChannels_saveMaxIntProjections
    - ○ Define directory where the raw data lives:

        → dataDir, line 23

        If reading from winstor, make sure you're logged in, and ideally run the script from a computer that has a 10GB connection, otherwise this is going to be very slow
    - ○ Define experiment name (folder name where your experiment data lives
    - ○ *If you wish, define a results directory - if you change this, at the moment you will also need to then change subsequent scripts, as the current assumption is that the results directory is fixed: the code creates a subfolder with the name of the experiment you want to analyse within that results directory*

2. <u>Run storm analysis for single molecule localisation</u>
    - ○ Using Python:
    - ○ Open anaconda navigator
    - ○ In top left corner, change base (root) to stormanls, that way you are in the environment that's set up to run the analysis
    - ○ Launch Spyder
    - ○ Open EJ_stormScript_testParams_andLoop_v4.py

        → This script lets you pick an example stack and test the parameters for all the bits and (one) fiducial

        → To save time, there is an option to crop each image & test the parameters on the cropped image

        → After having picked the parameters for a given bit, it is possible to

    - After having picked the parameters for a given bit, it is possible to test those parameters on other stacks and change them if desired
- There are only 2 things that need to be provided for this script before running it
    - Line 27: define the experimentName (this should be the same as in step 1)
    - Line 29: define the resultsFolderName
        - This creates a subfolder into which the results from the current run are saved, name can be anything, but be careful to adjust this is running a new analysis, else previous results will be overwritten
- *If you changed the results directory in step 1, then you need to make sure windowsDataPath (line 35) is the same*
- The rest of the script is interactive
    - If it looks like nothing is happening - check that a window hasn't popped up behind your spyder console & you just can't see it

- There are a few other scripts that allow you to
    - Just loop through the stacks if the parameters already exist
    - Look at a single stack
    - Only look at fiducials

3. Decode the single molecule localisations
    - Using MATLAB:
      AnalyzeMERFISH_EJ
    - Lines 23-25 again define folders where the data lives;
        - Provide the experiment name and resultsFolderName (ie the subfolder you created with this round's storm results)
    - The script loops through the stacks and saves the results for each stack with a corresponding filename into the experiment subfolder
    - Within that subfolder, there is another subfolder with the results of the drift correction in FiducialReports
    - Then I created a function that extracts the XY coordinates for all the genes in a given tile and saves them in a subfolder called geneLocs, in which there are subfolders for all the different genes that were in this experiments library
        - Inside each gene folder, there are the XY coordinates per tile for that given gene

4. Create tiled images and/or overlay several genes with DAPI
    - Using MATLAB
        - All functions should be documented & have explanations for

        - All functions should be documented & have explanations for required inputs
    - createTiledImage
        - Creates a tiled dapi image if it doesn't exist yet
        - Creates a tiled image of the gene(s) of interest
        - Overlays the gene dots over dapi
        - *Current drawbacks:*
            - → *Right now I am discarding all the dots that are outside the frame of the first round - because of drift, the frames of the different rounds don't match exactly and therefore later rounds might have dots outside what was in the frame in the first round*
            - → *We might want to consider doing "more intelligent stitching" where we don't discard that info*
            - → *Because I only consider what is inside the frame of the first round, I do the dapi background using the dapi image that was captured in the first round. However, we have dapi images for all rounds, which may help to create a more intelligent stitching.*
    - multiGeneOverlays
        - Creates a tiled image with dapi in the background and displays the dots of the genes provided
            - → *At the moment, the maximum number of genes that can be overlaid at the same time is 9 - I just haven't defined more colours than that, but it's an easy thing to fix if we want to add more genes*

5. <u>Cell segmentation</u>
    - Using MATLAB
        - Under development - I've got some code but haven't written a user-friendly function yet