mocoi_R2G

Iterative movement correction for calcium imaging. Corrects an image stack with activity dependent fluorescence (e.g., GCaMP data in green channel) based on movement correction of a simultaneously acquired, non-activity dependent fluorescent imaging stack (e.g. tdTomato or mRuby co-label in the red channel).

Required files:

moco package for ImageJ

(translation-based movement correction plugin developed by R. Yuste Lab)
Download from http://www.columbia.edu/cu/biology/faculty/yuste/methods.html.
Follow instructions in ReadMe.txt to install.

mocoi_R2G.ijm - (ImageJ macro)

1. Creates two directories: <root_dir>/registered/ and <root_dir>/moco results/ .
2. Creates "seed" reference image by applying iterative movement correction to first 1000 frames of red channel, using average of nFrames_seed frames as initial reference. Creates <root_dir>/registered/ref_img.tif , the average projection of this movement corrected stack.
3. Applies iterative movement correction to red channel and saves results of each repeat in <root_dir>/moco results/. Briefly, stitched stack is corrected with moco, using ref_img.tif as reference, then averaged to generate reference for next repeat. The last three parameters described below are used to test whether to repeat. Creates <root_dir>/registered/reg_<savFile_name(1:end-4)>red.tif , the movement corrected stack from the red channel.
4. Results are summed to calculate total translation (dx,dy) for each frame, and then applied to the green channel. Creates mocoi_Results.txt, which records in each row the total translation (dx,dy) for each frame. Creates <root_dir>/registered/reg_<savFile_name(1:end-4)>.tif, the movement corrected stack from the green channel.

Default Parameters

(use 'Plugins>>Macros>>Edit…' to change)

nFrames_seed = 100; // nFrames to AVG for for initial ref image.
max_error = 1; // threshold abs(dx)+abs(dy) per frame for repeat.
err_tolerance = 0.001; // threshold cumulative error (pixels) for repeat.
max_rep = 10; // threshold #iterations for considering only max_error for repeat.

Developed by MJ Siniscalchi (Kwan Lab, Yale University); last edited 11/4/16.