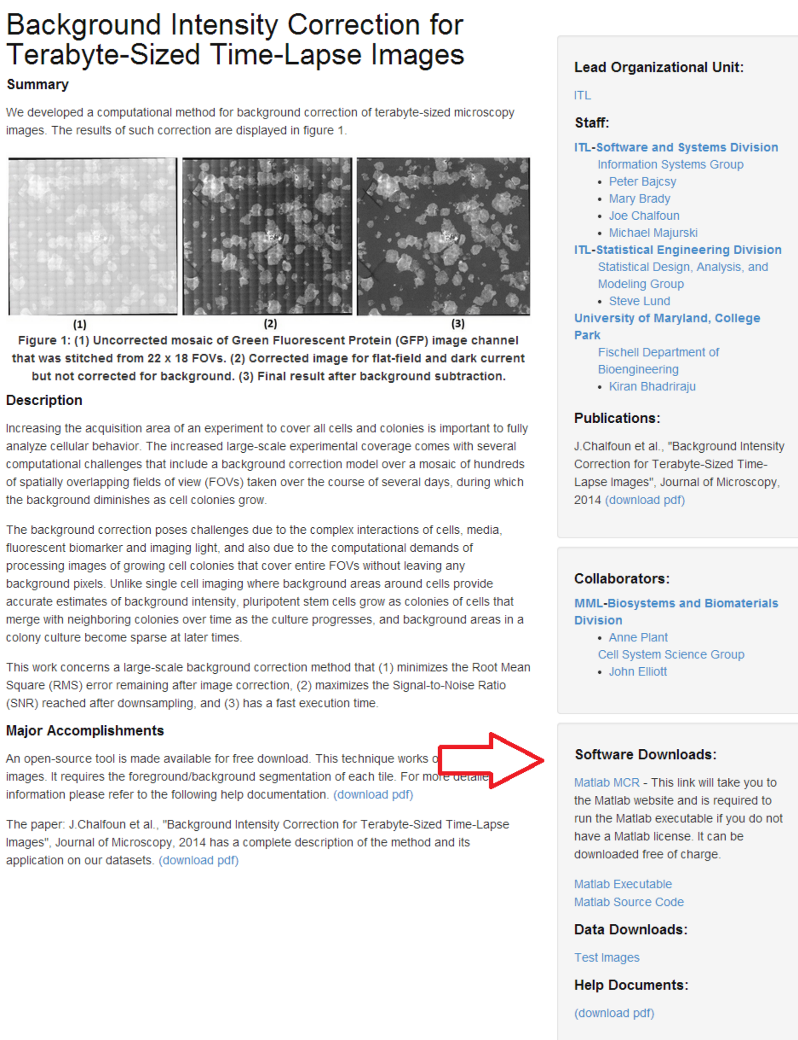
Background Intensity Correction for Terabyte-Size Tile-Lapse Images User Guide

Version 1.1

This background correction method is developed to correct a mosaic microscopy images stitched together from a grid of individual image tiles. It requires a grid of individual raw images, a grid of corresponding segmented masks (foreground background), a dark image and a flatfield correction image. For more details about this background correction technique and the dataset it was developed for, please see J.Chalfoun et al., "Background Intensity Correction for Terabyte-Sized Time-Lapse Images", Journal of Microscopy, 2014.

1 Getting Started: Downloading Source Code

This section describes how to download the Matlab source code for the background intensity correction and set it up within the Matlab environment or as independent application. Both the source code as well as executables to perform the background correction are available on the project webpage (Figure 1): http://isg.nist.gov



Background Intensity Correction for Terabyte-Sized Time-Lapse Images

Summary

We developed a computational method for background correction of terabyte-sized microscopy images. The results of such correction are displayed in figure 1.

(1) (2) (3)

Figure 1: (1) Uncorrected mosaic of Green Fluorescent Protein (GFP) image channel that was stitched from 22 x 18 FOVs. (2) Corrected image for flat-field and dark current but not corrected for background. (3) Final result after background subtraction.

Description

Increasing the acquisition area of an experiment to cover all cells and colonies is important to fully analyze cellular behavior. The increased large-scale experimental coverage comes with several computational challenges that include a background correction model over a mosaic of hundreds of spatially overlapping fields of view (FOVs) taken over the course of several days, during which the background diminishes as cell colonies grow.

The background correction poses challenges due to the complex interactions of cells, media, fluorescent biomarker and imaging light, and also due to the computational demands of processing images of growing cell colonies that cover entire FOVs without leaving any background pixels. Unlike single cell imaging where background areas around cells provide accurate estimates of background intensity, pluripotent stem cells grow as colonies of cells that merge with neighboring colonies over time as the culture progresses, and background areas in a colony culture become sparse at later times.

This work concerns a large-scale background correction method that (1) minimizes the Root Mean Square (RMS) error remaining after image correction, (2) maximizes the Signal-to-Noise Ratio (SNR) reached after downsampling, and (3) has a fast execution time.

Major Accomplishments

An open-source tool is made available for free download. This technique works o[...] images. It requires the foreground/background segmentation of each tile. For more detaile[...] information please refer to the following help documentation. (download pdf)

The paper: J.Chalfoun et al., "Background Intensity Correction for Terabyte-Sized Time-Lapse Images", Journal of Microscopy, 2014 has a complete description of the method and its application on our datasets. (download pdf)

Lead Organizational Unit:

ITL

Staff:

ITL-Software and Systems Division
Information Systems Group
- Peter Bajcsy
- Mary Brady
- Joe Chalfoun
- Michael Majurski

ITL-Statistical Engineering Division
Statistical Design, Analysis, and Modeling Group
- Steve Lund

University of Maryland, College Park
Fischell Department of Bioengineering
- Kiran Bhadriraju

Publications:

J.Chalfoun et al., "Background Intensity Correction for Terabyte-Sized Time-Lapse Images", Journal of Microscopy, 2014 (download pdf)

Collaborators:

MML-Biosystems and Biomaterials Division
- Anne Plant

Cell System Science Group
- John Elliott

Software Downloads:

Matlab MCR - This link will take you to the Matlab website and is required to run the Matlab executable if you do not have a Matlab license. It can be downloaded free of charge.

Matlab Executable
Matlab Source Code

Data Downloads:

Test Images

Help Documents:

(download pdf)

Figure 1: Project Webpage Highlighting the Source Code Download

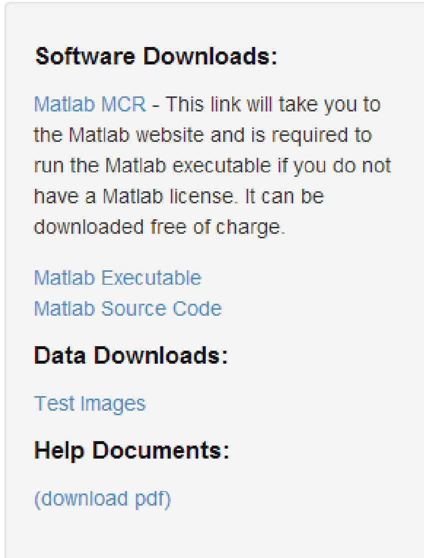


Figure 2: Source Code Download WebPage Element

The link "Matlab MCR" (Figure 2) will take you to the Matlab web page where you can download the free MCR (Matlab Compiler Runtime). The MCR is required to run any compiled Matlab code without requiring a Matlab license. However, you need to match the version of the MCR to the version of Matlab used to compile the executable. This version information is available in the readme.txt file that accompanies the executable. In our case, please make sure to download the MCR version 8.2.

The link "Matlab Source Code" will download "background_correction_source_code.zip" file containing the source code in Matlab that will perform the background correction. The link "Matlab Executable" will download a compiled version of the source code.

2 Test Data Organization

The test data contained within "background_correction_test_images.zip" has everything required to perform a small background correction test to validate that the code is setup properly and runnable. It also provides a means to explain the organization and structure that is required for this correction method to work.

The data download labeled "Test Images" will download a "background_correction_test_images.zip" file containing a 5x5 grid of test images, and all associated metadata, to demonstrate how the background correction works.
"background_correction_test_images.zip" manifest:

- full_mosaic_test (folder)
    - cor_imgs (empty folder for results)
    - raw_imgs (folder)
        - raw_stitched_mosaic.tif
    - seg_imgs (folder)
        - seg_stitched_mosaic.tif

        - raw_img_name_grid.csv
        - seg_img_name_grid.csv
- sub_mosaic_test (folder)
    - cor_imgs (empty folder for results)
    - raw_imgs (folder)
        - img_r01_c01.tif
        - ...
        - img_r05_c05.tif
    - seg_imgs (folder)
        - seg_img_r01_c01.tif
        - ...
        - seg_img_r05_c05.tif
    - dark_image.tif
    - flatfield_image.tif
- raw_img_name_grid.csv
- seg_img_name_grid.csv

There are two types of background correction methods presented here. The first we denote as the "sub-mosaic" technique where each image tile in the grid is corrected individually. The second type we denote as the "full-mosaic" technique where the stitched mosaic image is corrected as a whole. Therefore, this demonstration test dataset contains the same image data in two formats. The first format is a 5x5 grid of image tiles, each 1040x1392 pixels in size (Figure 3), that form a composite mosaic when stitched together. The second is the composite stitched mosaic image which is 6454x5826 pixels.

Due to the fact that knowledge of which pixels are background is required to perform the correction, segmented mask images are included as well. The raw fluorescent images each have a corresponding segmented mask image which contains the information on which pixels are background. These segmented mask images have pixel values of zero for background and pixel values of one for foreground.

Within the file "background_correction_test_images.zip", the sub-mosaic test data is contained in the folder "sub_mosaic_test" and the full-mosaic test data is contained in the folder "full_mosiac_test".

For the background correction sub-mosaic technique to work, the program needs to know the grid location of each individual image tile. To accomplish this two csv files, "raw_img_name_grid.csv" and “seg_img_name_grid.csv” have been included with test data to document the organization of the image tile grid.

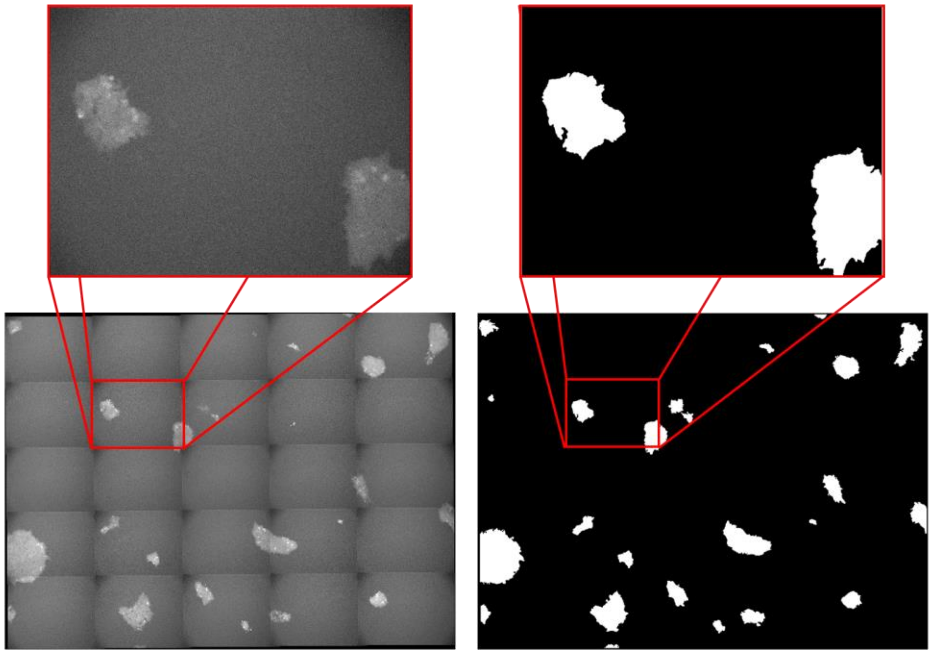


Figure 3: Example Un-Corrected Single Tile Fluorescent Image (top left); Example Segmented Single Tile Image (top right); and Stitched 5x5 Mosaic Un-Corrected (bottom left); Stitched 5x5 Segmented Mosaic (bottom right)

The file "background_correction_test_images.zip" also contains two other images of interest contained within the sub-mosaic folder. "dark_image.tif" is the dark current image taken on the microscope with the shutter closed. It represents the minimum noise level of the CCD camera when no light hits the photo-detector. "flatfield_image.tif" is an image that exemplifies the vignetting effects of the microscope imaging system. While the dark current effect is an additive effect, the flat-field is a multiplicative effect. These two images can optionally be used when performing sub-mosaic background correction on the test data.

3 Code Organization

The source code download is contained within "background_correction_source_code.zip".

"background_correction_source_code.zip" manifest:

- Background_Correction_GUI.m
  This is the main entry point for running the background correction code. This will present the user with a GUI to control the background correction parameters.
- doc (folder)
  - Help.pdf
- img (folder)
  - NIST_Logo.tif
- src (folder)
  - bin_image.m
    This is a function to bin an input image to produce a smaller resolution copy of the same image. The *I1* parameter is the input image to be binned. The *bin_size* parameter is the size of the bin to be applied to the input image. A *bin_size* of 2 would produce an output image pixel by averaging a 2x2 block of input image

pixels. A *bin_size* of 4 would produce an output image pixel by averaging a 4x4 block of input image pixels.

- correct_fullmosaic.m
  This is the background correction function for the full-mosaic techniques.
- correct_submosaic.m
  This is the background correction function for the sub-mosaic techniques.
- save_32_tiff.m
  This is a function to save a 32bit floating point TIFF image.

4 Background Correction Methods

The full-mosaic background correction technique operates by fitting a model to the known background pixels and using the model to interpolate the background values under the foreground regions. The sub-mosaic background correction technique operates on individual image tiles, relying on the premise that the background is constant through the whole imaged region (Figure 4). This technique will not work for imaged regions that do not have a uniform background. Use full-mosaic based corrections for those cases (Figure 5). For details about how the two techniques work see the paper J.Chalfoun et al., "Background Intensity Correction for Terabyte-Sized Time-Lapse Images", Journal of Microscopy, 2014.

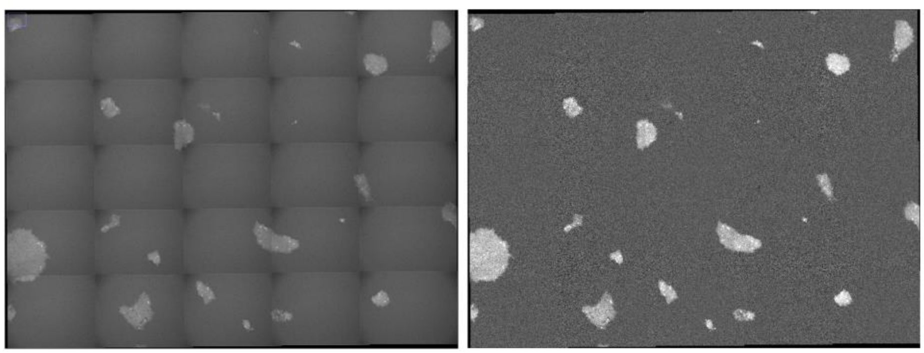


Figure 4: Un-Corrected mosaic (left); Sub-Mosaic corrected using poly33 model(right)

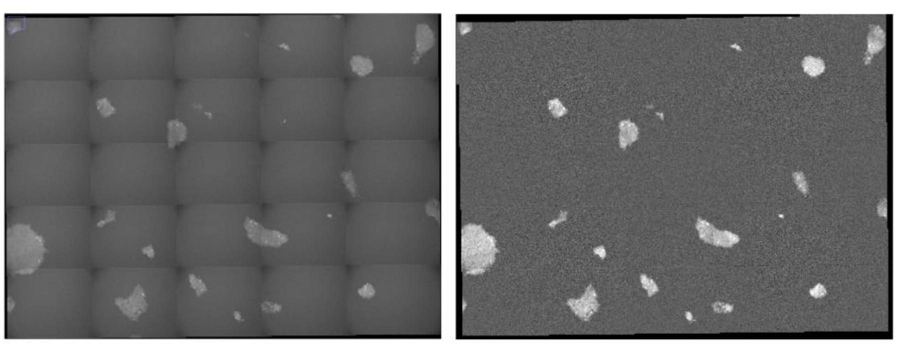


Figure 5: Un-Corrected mosaic (left); Full-Mosaic corrected using linearinterp model(right)

5 Background Correction GUI help

With the exception of the Dark and Flat-Field images, all of the parameters are required.

- Raw Image Source Directory
  This is the file path to the images that are going to be corrected. Within this directory should be the set of tiff images who's file names are found in the Image Name CSV Grid file that was loaded by selecting the Load Raw button.
- Segmented Image Source Directory
  This is the file path to the segmented mask images of the images to be corrected. Within the directory should be the set of tiff images who's file names are found in the Image Name CSV Grid file that was loaded by selecting the Load Seg button.
- Output Image Directory
  This is the file path to the folder where the output images will be saved. If using a full-mosaic method the single tiff image "corrected.tif" will be saved here. If using a sub-mosaic method one corrected image tile will be saved here for each input raw image tile. The output images will have the same names as the input raw images.
- Use Dark Image (Select Button)
  Select this if you want to use a dark image in the background correction. The dark image is subtracted from each raw input image during the correction. Use the select button to navigate to the dark image and select it.
- Use Flat-Field Image (Select Button)
  Select this if you want to use a flat-field image in the background correction. The raw input image is divided by the flat-field image during the correction. Use the select button to navigate to the flat-field image and select it.
- Correction Model
  This is the model that Matlab's fit function will use to background correct the images.
- Bin Size
  To reduce the computatin time required for background correction the images are binned to reduce their size. Valid values consist of any integer that is greater than or equal to 1 and less than half the smallest dimension of the images being corrected if using the sub-mosaic method. There is no practical maximum value when using the full-mosaic. However, the larger the value the less accurate the results are. For the sub-mosaic technique bin sizes of 8 or 16 have almost no effect on the accuracy of the results. It is recommended to have a bin size of at least 8.
- Image Name CSV Grid
  This is a pair of csv files containing the image file names organized to reflect their location in the 2D grid of images that make up the mosaic. This file is required only for the sub-mosaic technique. There is a load button for the raw image name grid and the segmented mask image name grid.

One advantage of the sub-mosaic technique (Figure 6,7) is that it takes roughly the same amount of time to correct a 5x5 grid as a 10x10 grid because the number of times Matlab's fit function gets called scales with the bin size; the smaller the bin size the more calls to fit. Therefore there are the same number of calls to fit for a 5x5 grid using a bin size of 8, as there is for a 10x10 grid using a bin size of 8. Each call to fit would have 4x the number of data points, fitting a surface to a 10x10 pixel surface as opposed to a 5x5 pixel surface. As the image grid size increases the size of the surface that Matlab's fit function operates on increases, but for small image grid sizes the number of calls to the fit function is the dominant factor in runtime.

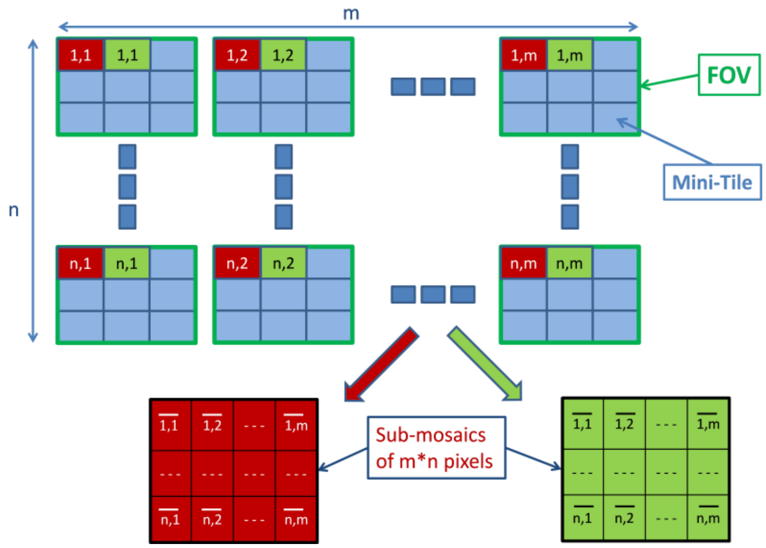


Figure 6: Average values of the red pixels from the same location of all FOV are assembled into the red sub mosaic. The same procedure is performed on the pixels at the following location (the green one) etc. until all locations on the FOV have been assembled into sub mosaics. Background estimation by surface fitting is performed to the constructed sub mosaics. There are m = 18 times n = 22 equal to 396 FOVs used for background estimation. The red and green colors denote the original location of the pixels in each FOV and their new locations in a sub mosaic images.

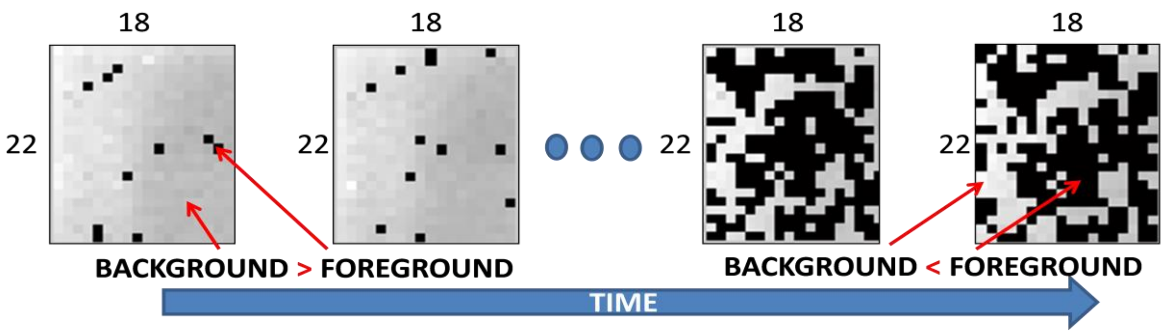


Figure 7: The resulting sub mosaic images over time with the decreasing number of background pixels due to the growth of stem cell colonies. The sub mosaic images on the left are at the early time points and on the right at the later time points of the experiment. Non-black pixels in 18 x 22 sub mosaic images are at those FOV locations that contain only background pixels within a FOV. The black pixels in 18 x 22 sub mosaic images represent the average intensity value of the foreground pixels in that particular FOV.