Manual

Histogram-based background subtractor for ImageJ

Janick Cardinale, janick@inf.ethz.ch,
MOSAIC Group, ETH Zürich, www.mosaic.ethz.ch

January 2008 (rev. November 2010)

1 Introduction

To process signals from light microscopy images, it is often useful to correct for inconstant background illumination and artifacts from autofluorescence. The implemented algorithm is based on the assumption that, compared to the background region, object (foreground) regions are small. The plugin builds local histograms and assumes the most occuring intensity to be part of the background. The plugin is is an alternative to deletion of low frequencies in Fourier space or the rolling ball algorithm (ImageJ standard implementation) proposed by S.Steinberg (1983). An example of the plugins functionality is shown in figure 1.

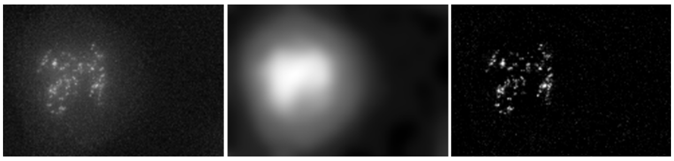

Figure 1: Left: The original image A. Kinetochores are marked with a fluorescent protein. Middle: The generated background image B. Right: The final image with removed background illumination: C = A - B.

2 Installation

Copy the `mosaic_plugins.jar` file to the ImageJs `plugins` directory. Then, restart ImageJ. The plugin can be launched from the `Mosaic` submenu in ImageJs plugin menu.

The plugin handles 8-bit, 16-bit and 32-bit grayscale images or stacks.

Prerequisits: At least Java 5.0 and ImageJ 1.36.

3 How to use the plugin

Open a grayscale image. Before starting the plugin, one might set a lower threshold using the `Ctrl-t` shortcut. This threshold can later be used to propose the plugins parameter. The lower intensity threshold should be set such that objects are above the threshold (and will be labelled red).

Start the plugin from the `Plugin` menu. Select the length parameter for your image: the parameter should be as small as possible but larger (in pixel) than any line that one can fit within the largest object in your image.

If the autothreshold-algorithm (EM-algorithm) was successful or if you had manually selected a threshold before you launched the plugin, one might use the `propose parameter` button to get a automatically proposal for the histogram side length parameter.

If you would like to get the background illumination image (the image that is subtracted from the original one), you can select the checkbox `show background image`. The background image (or image stack) will be shown after processing all the slices in a separate window.

There is no undo for this operation.

4 Algorithm

4.1 Main algorithm

A sliding window with side length L is moved along the image with a speed of $L/2$ per iteration. At every iteration, the maximum of the local histogram (the histogram of the pixel intensities within the sliding window) is supposed to be background at the sliding windows position (center). For pixel where the background is not explicitly calculated, the background intensity is approximated by bi-linear interpolation of it's four nearest explicitly calculated pixel. A schematic overview of the algorithm is shown in figure 2.

At the boundary of the image, the image is mirrored to generate the local histogram (Neumann boundary conditions).

In a second step, the created background image is convolved with a Gaussian kernel with a radius of $2 \cdot L$ to filter for noise due to a too small L or due to very high concentrations of objects.

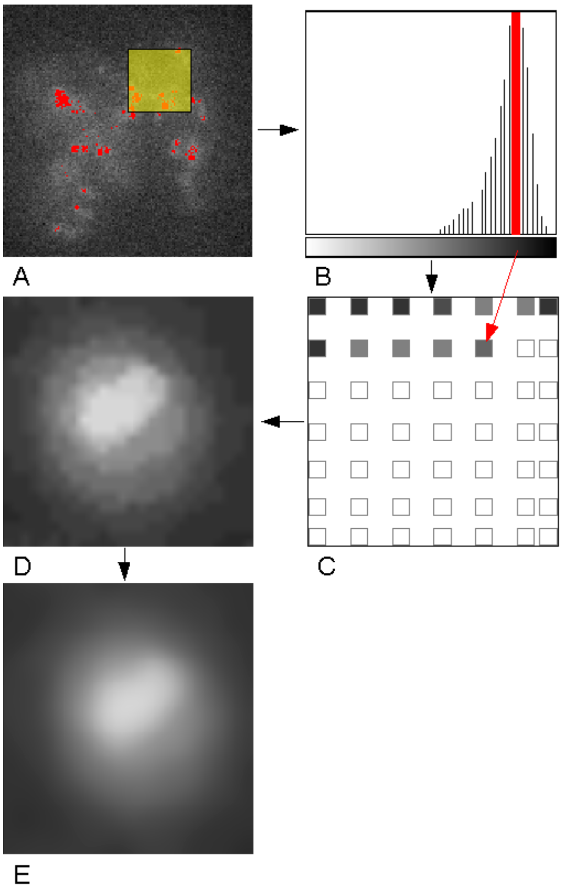

Figure 2: A: The original image and the sliding window. B: The histogram of the pixel values within the sliding window. C: The generation of the background image. The pixel value at position of the middle of the sliding window is equal to the mode of the histogram. D: Not all pixel are calculated explicitly, this would be too expensive. All other pixel are interpolated bi-linearly. E: The background image after convolution with a Gaussian kernel.

4.2 Proposal of the length parameter L

A set of connected components $\mathcal{C}$ is generated using basic intensity based thresholding. Next, the largest diameter $d_{\max}$ that fit within any of the connected components in $\mathcal{C}$ is determined. The suggested parameter L for the window size is double the size of $d_{\max}$.

5 Disclaimer

IN NO EVENT SHALL THE ETH BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE ETH HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. THE ETH SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN ”AS IS” BASIS, AND THE ETH HAS NO OBLIGATIONS TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.