Transmitted light timelapse images correction
ImageJ macro
User guide
Marcel Boeglin June-July 2020

ImageJ macro "DivideStackByRunningAverage_29.ijm"

About the macro name: should be changed because current version is not limited to a division of input frames by running sub-stack Average T-projections.

Designed to correct transmitted-light time-lapse images in cases such as:

- Dirt on the camera or optical surfaces;
- Illumination unevenness due to:
  1. imperfections in illumination system;
  2. distortion of light beam by sample (e. g. imaging near border of wells).

Two working modes:

- Active image processing (**single image processing**);
- Processing of a time-lapse images folder (**batch processing**).

Accepted image dimensions: XYT, XYZT, XYCZT.
Processed dimensions: XYT.
Multi-channel images are reduced to their transmitted-light channel.
Z-stacks are reduced to their Average Z-projection.

If an acceptable image is selected, the macro displays a suite of user interfaces and processes it after eventual reduction to its XYT dimensions.
If no images are open, the macro switches to batch-processing mode and asks for input-output folders and for including and excluding file filters and processing parameters.

The macro was first designed to work without any empty-field correction-image to process images close to the border of well, in which case illumination and images are distorted by the surface of the culture medium.
If no such problems occurred during acquisition, the macro can be run using a correction-image consisting in an empty-field image (the same for all positions in case of a multi-position time-lapse).

Algorithms:

Algorithm 1a: division of frames by running sub-stack average-projection
For each frame of active transmitted-light time-lapse image:
- creates a running sub-stack of constant, adjustable size (timeAverageRange):
  first slice = current input image frame - timeAveragingRange/2
  last slice = current input image frame + timeAveragingRange/2-1
- does average T-projection of running sub-stack;
- divides current input slice by projection and multiplies the result
  by input stack mean value;
  in order to remove jumps of slice brightness.
¤ Divides corrected stack by its running average projection

- removes optionally dark and bright zones in projection
- does optional gaussian blur of projection

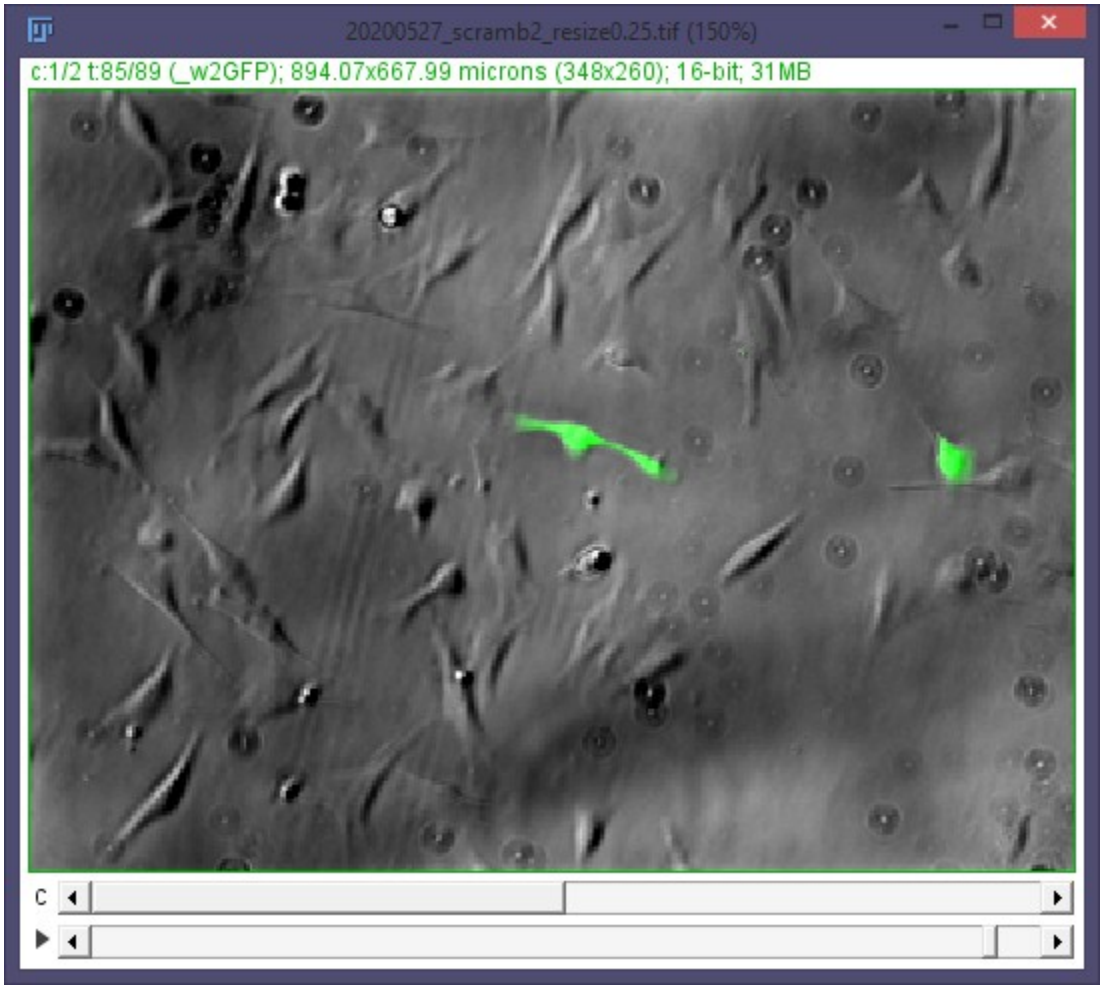
Algorithm 1b: division of each frame by empty-field image (correction-image).

Algorithm 2: division of each frame by an image obtained from the frame by:
- removal of brightest and darkest regions smaller than size of objects to be preserved;
- gaussian blur with a sigma calculated from size of objects.

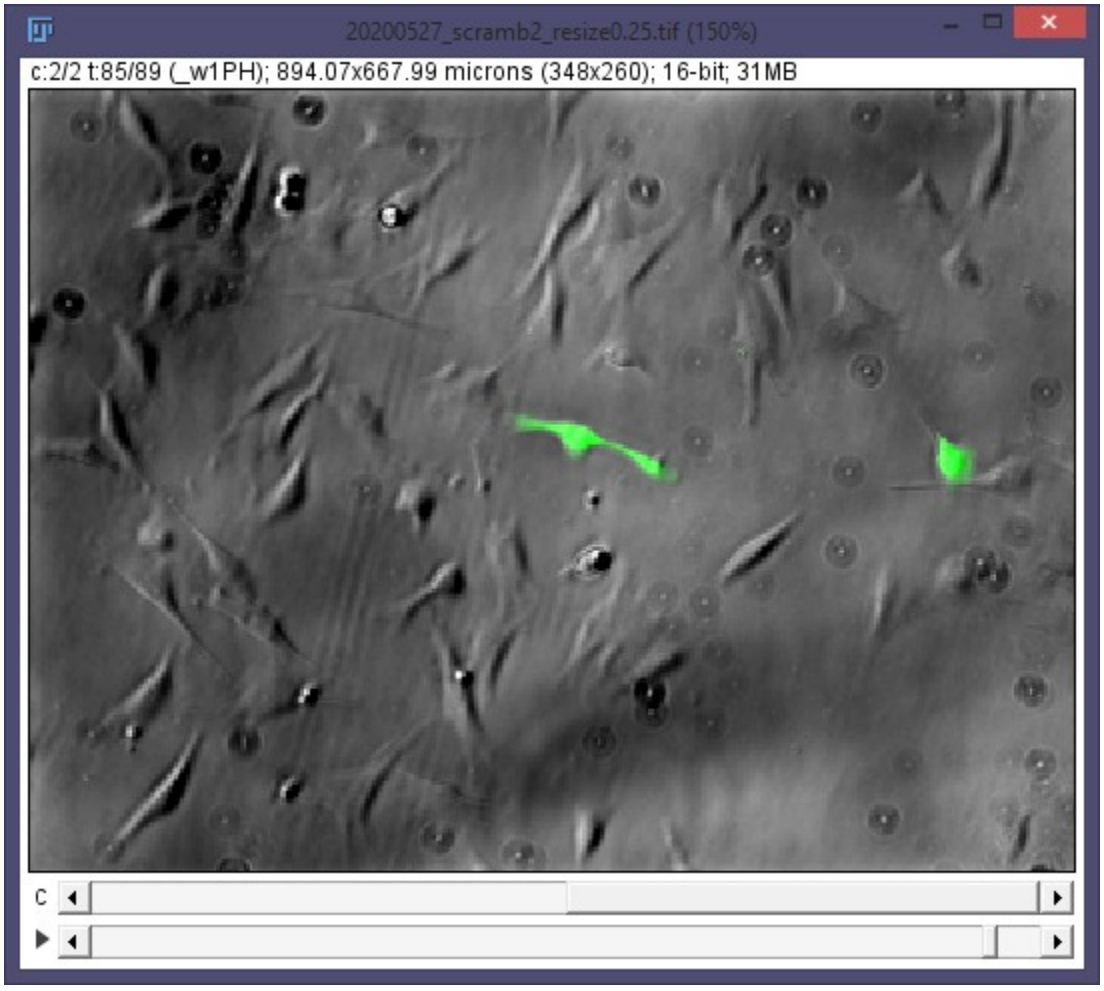
Algorithm 2 is similar to FFT Bandpass but faster for small objects and provides results of better quality.

Algorithms 1 and 2 are independent.
Processing can be limited to 1a or 1b.
It can be followed by algorithm 2.
It can also be limited to Algorithm 2.

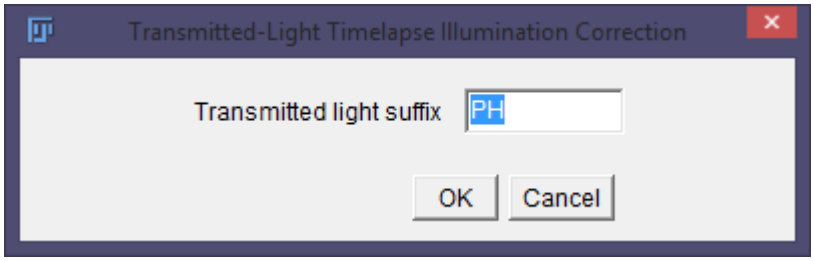
Example of accepted input image:



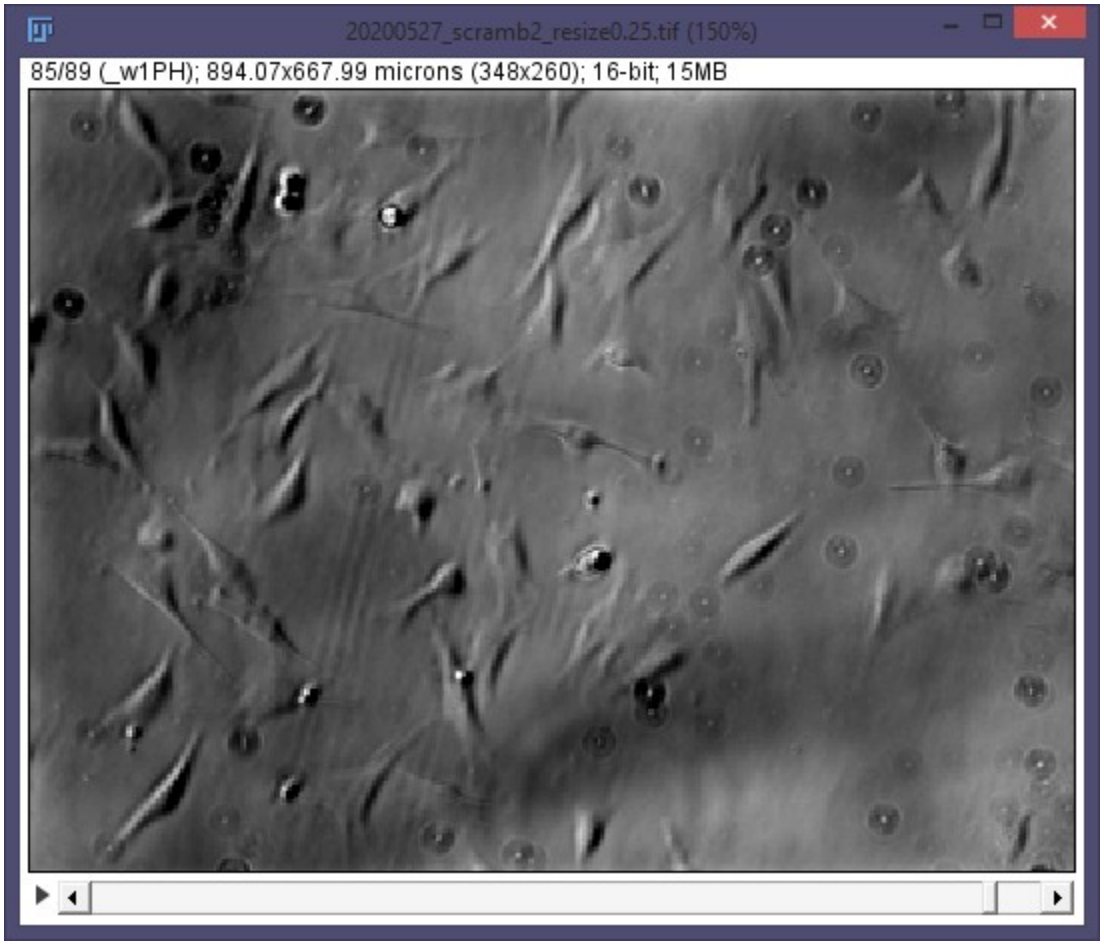
Same input image with channels slider on phase contrast (_w1PH):



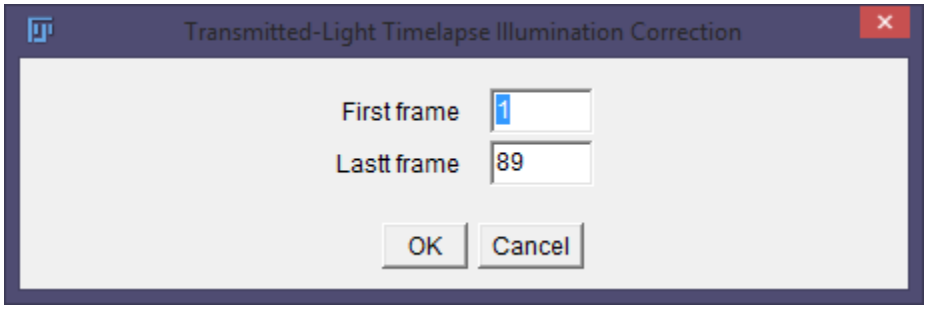
Choose label of transmitted-light channel:



This results in reduction of input image to phase contrast channel:

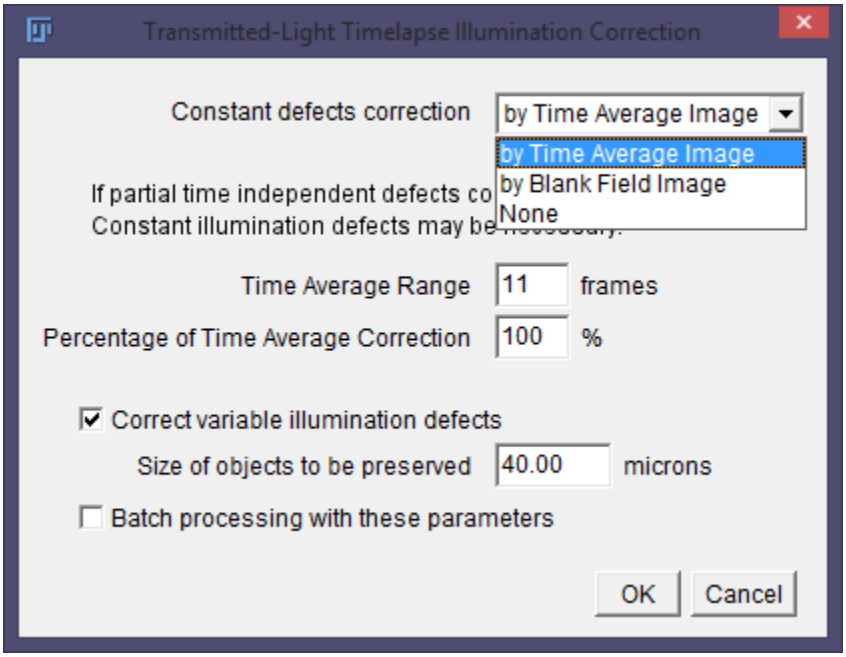


Time-range to be processed:



Defaults: 1st frame = 1, last frame = number of time-points of input image.
Allows to limit processing to a part of time-lapse to be analyzed.

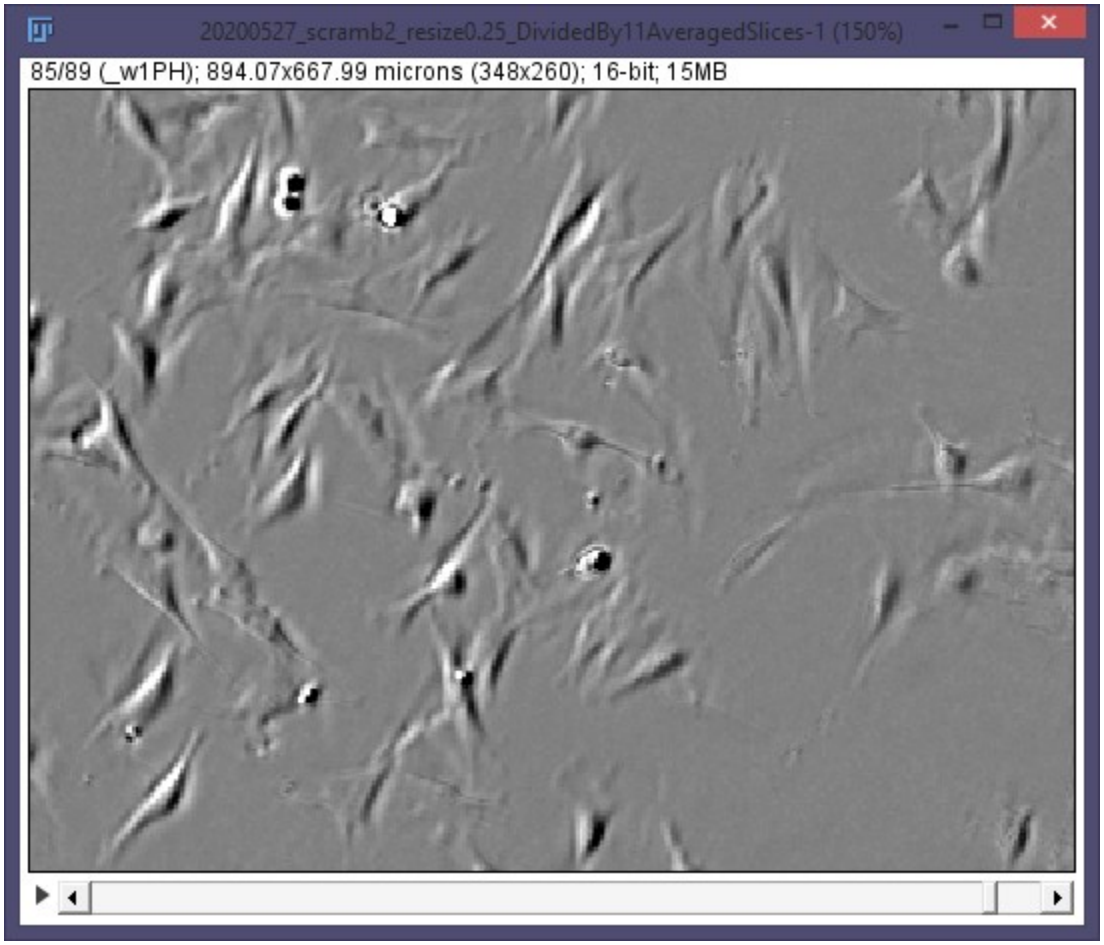
Processing parameters:



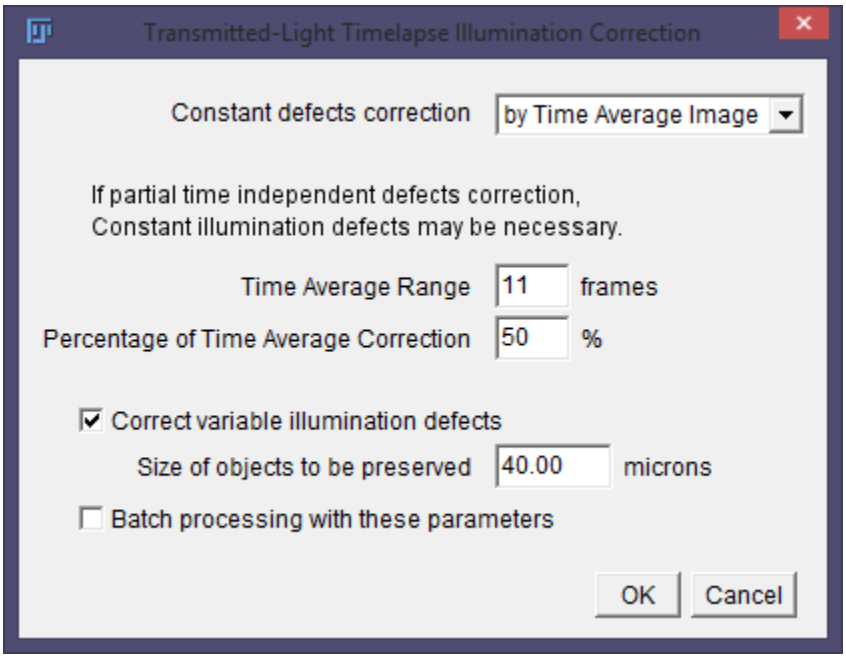
Quality of the result strongly depends on:

- nature and severity of image defects;
- processing algorithm(s);
- processing parameters.

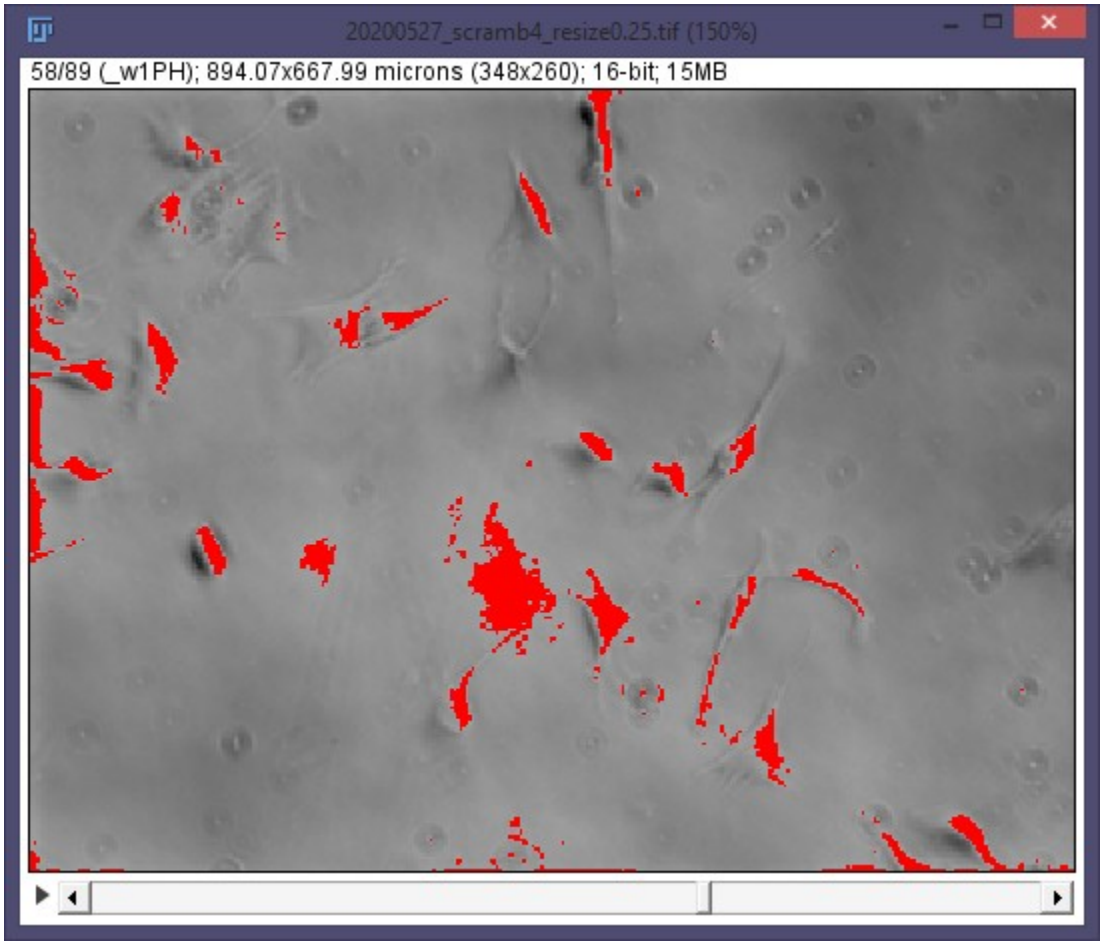
Result:



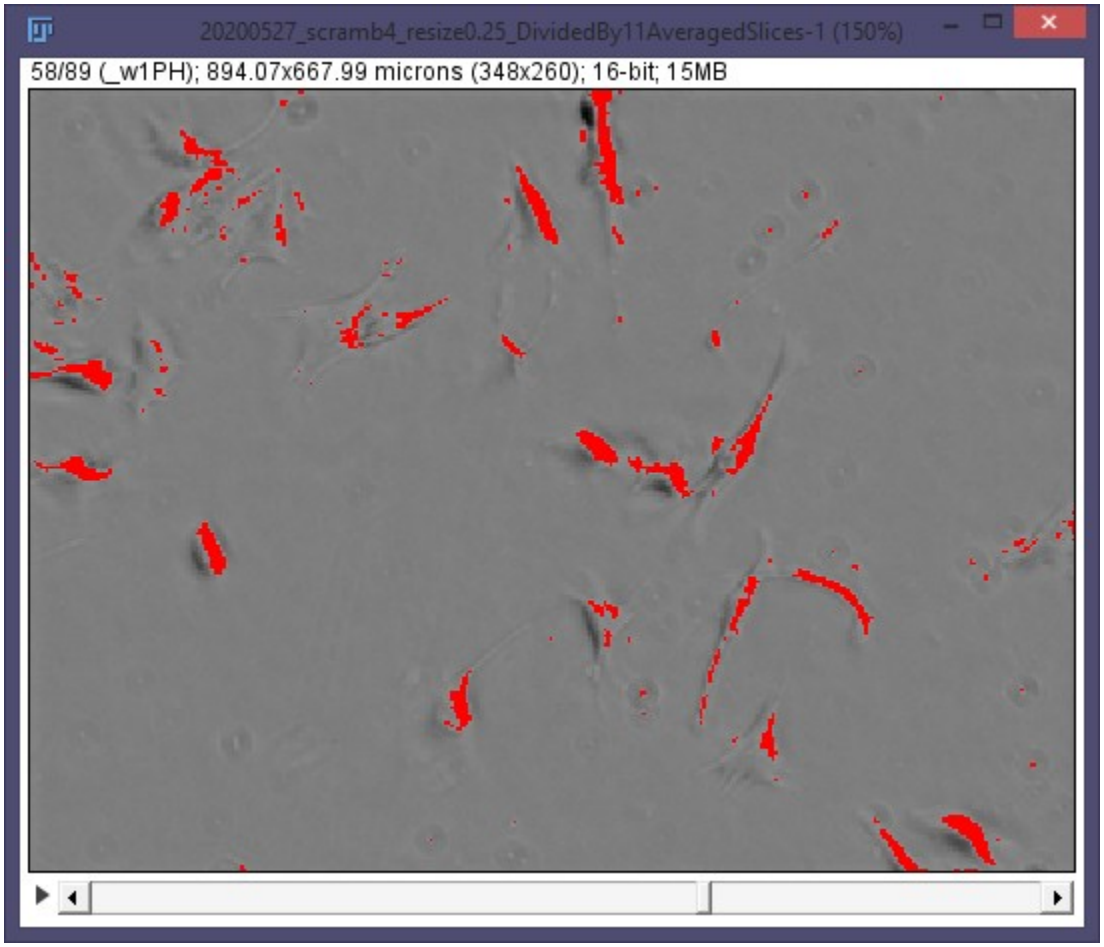
Other example:
Processing parameters:



Thresholded unprocessed image:



Thresholded Processed image:



* Author: Marcel Boeglin - June - July 2020
* E-mail: boeglin@igbmc.fr
*/