University of British Columbia Okanagan /Department of Computer Science / Unit 5

A computer vision based algorithm for counting both dead and alive bio-cells

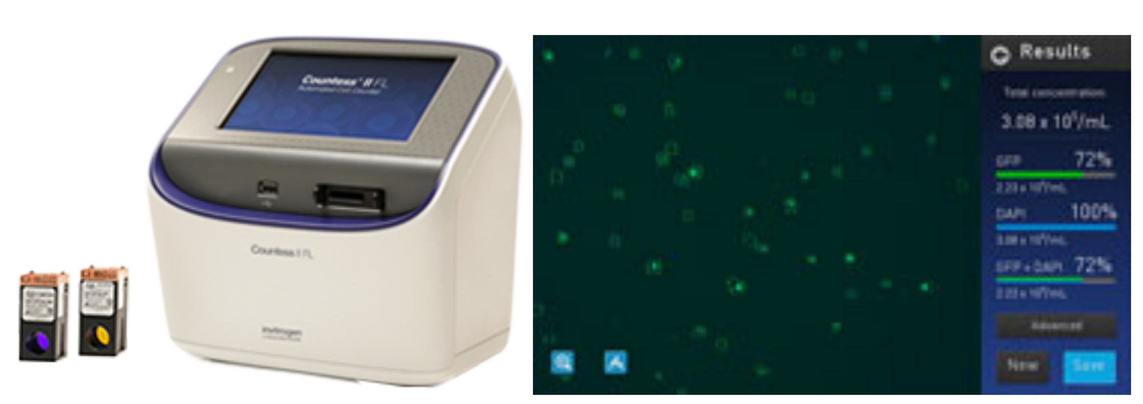
Julius Wu, Dr. Abdallah Mohamed

Abstract:

A cell-counting algorithm was created to efficiently count bio-cells images taken from a smartphone camera. The algorithm uses computer vision techniques, and it proved to be reasonably comparable to similar algorithms. The algorithm successfully resulted in counted 90% of the alive cells and 75% of the dead cells (due to lighter dead cells blending with the background color). The algorithm is different from similar ones such that it works with color images instead of just greyscale image and is able to count two different types of cells (dead and alive) using the RGB channels.

Introduction:

Traditionally, Scientists and Researchers have to manually count cells under a microscope with a manual hand tally counter. The manual cell counting method is time consuming and sometimes inaccurate due to miscount and/or double counting. According to Al-Khazraji et al.'s "An automated cell-counting algorithm for fluorescently-stained cells in migration assays", manual counting method was tested and compared with a cell counting algorithm. 47 cell images was used and it took 2.4 hours to count manually, comparing to 14.5 seconds using an automated cell counting algorithm.



Although, there are lab graded cell counting equipment allowing researchers and scientists to count cells accurately and fast (< 10 seconds) for example, Thermo Fisher's "Fluorescence counting with color options". The equipment could be very expensive and costs upward of CAD 6,000.00 dollars. The goal is to have smartphone and attract to the Ocular lens of the microscope to take photos of the cells then process them with third party software to cut down expensive equipment's cost.

Our goal is to use Matlab's computer vision and image processing toolbox to build an automated cell-counting algorithm for counting both dead (fluorescently-stained) cells and alive cells from a smartphone's camera. From the sample photos given by Dr. Isaac Li, there are flaws and issues from the photos that needs to be tackled.

Proposed Method:

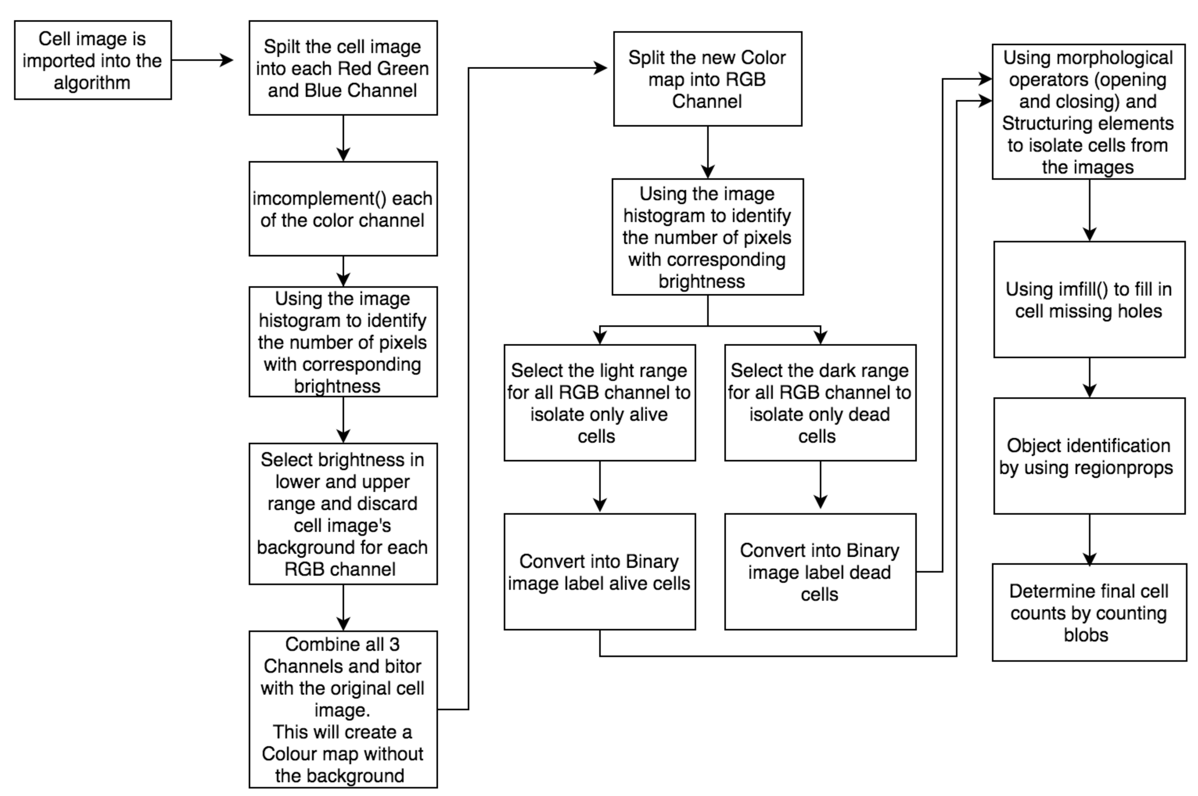


Figure 1) System Diagram outlining the main steps of the proposed cell counting method

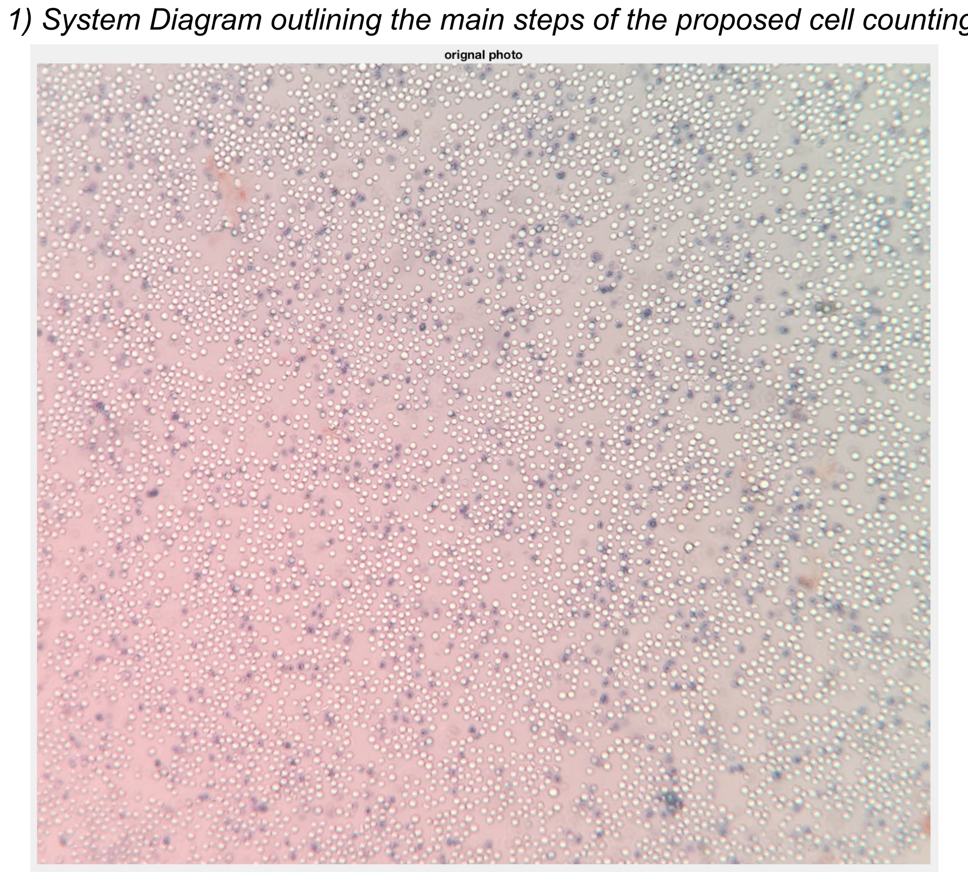


Figure 2) cropped image with adapthisteq()

1) Pre-processing the image: Raw image is imported into the Matlab program. The algorithm asked the user to crop out the selecting area for cell counting. Adapthisteq() method is used for each color channel to enhances the contrast of the image. Allowing the image to be sharper and clearer for analyzing and to fix smartphone photo's color balance.

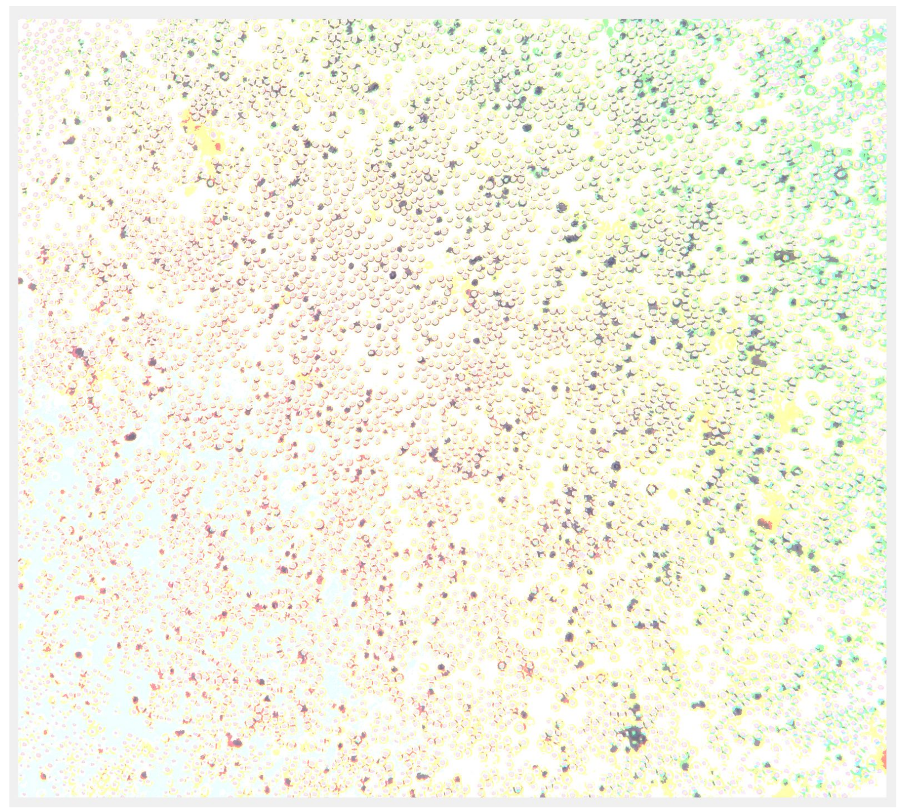


Figure 3) Color map image without the background

2) Removing image's background: Separate the cropped image into each red, green, and blue channels. First, to invert the color by using imcomplement(). Using imhist() to inspect each color channel's histogram. From each color channel, select out the upper range and the lower range of the histogram that represent the dead and alive cells and discard the background. Using bitor() operator, invert and concatenate all color channels into on image. This will create a color map image without the background.

3) Identifying dead cell color range: Once again, separate the color map image into each red, blue, green channel. Using imhist() to limit out the lower light range of the all three color channel. This will isolate only dead cells from the color map image. Create a binary image of the dead cells by using OR operators the channels.

4) Identifying alive cell color range: Using imhist() to limit out the upper light range of the all three color channel. This will isolate only the alive cells from the color map image. Create a binary image of the alvie cells by using OR operators the channels.

5) Using morphological operators: Morphological operators such as opening and closing are used for separation and identification of shapes and sizes. This allow to remove unnecessary noises and smudges. This also allow to identify overlapping cells to decrease miscounting. imfill() method is then used to fill in holes and gaps of the cells.

6) Object Identification using regionprops: Using regionprops() allowing to find the area of the alive and dead cells. This will help with identifying the cell's area and allowing for counting.

7) Determining final cell count: Using cat(num, blobs(:).Area) method to find area bigger than a set size.

Results:

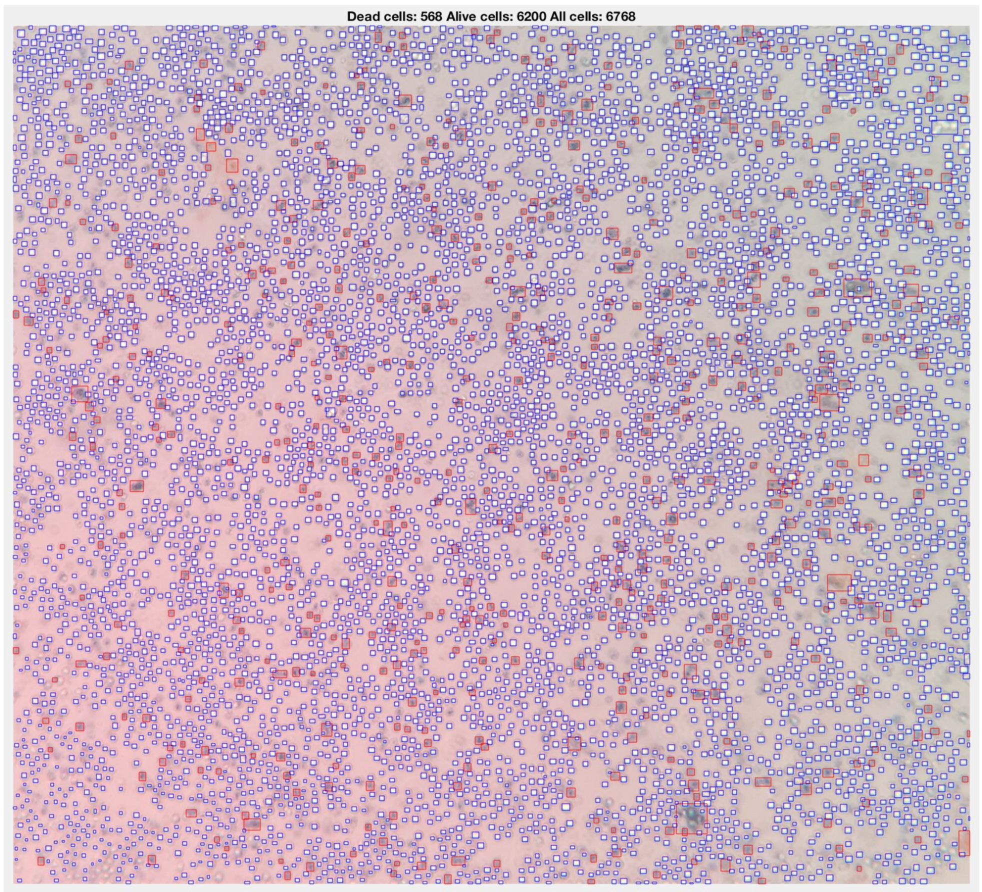


Figure 6) Final Image of the resulting cell counts "Dead cells: 568, Alive cells: 6200, All cells: 6768"

This method was tested with the sample images provided by Dr Isaac Li. Overall the algorithm works great with timing. Avenging less than 10 seconds run time per cell photo. Comparing to manual count avenging10 minute per cell photo. The algorithm was able to calculate almost all the alive cells (95%) in selected photos. We have issue, when alive cells and the background color is the same, the algorithm have trouble identifying the alive cells. When detecting dead cells, the algorithm have trouble to detecting lighter stain cells. The algorithm can detected 75% of all the dead cells. This is because some of the dead cells are very similar to the background color which the algorithm can't detect them.

Reference / Bibliography

1. D. Forsyth and J. Ponce, Computer vision, 2nd ed. Boston: Pearson, 2012.
   R. Gonzalez and R. Woods, Digital image processing, 1st ed. Will be used alongside
2. Al-Khazraji et al. Biological Procedures Online 2011, 13:9, "An automated cell-counting algorithm for fluorescently stained cells in migration assays." http//www.biologicalproduresonline.com/content/13/1/9 (19 October 2011)
3. https://www.thermofisher.com/ca/en/home/life-science/cell-analysis/cell-analysis-instruments /automated-cell-counters.html
4. Fuyong Xing, Lin Yang, "Robust Nucleus/Cell Detection and Segmentation in Digital Pathology and Microscopy Images: A Comprehensive Review", *Biomedical Engineering IEEE Reviews in* , vol. 9, pp. 234-263, 2016, ISSN 1937-3333.
5. Peter Markowsky, Svenja Reith, Tabea E. Zuber, Rainer König, Karl Rohr, Christoph Schnörr, "Segmentation of cell structures using Model-Based Set Covering with iterative reweighting", *Biomedical Imaging (ISBI 2017) 2017 IEEE 14th International Symposium on* , pp. 392-396, 2017, ISSN 1945-8452.

Acknowledgement

I would like to thank Dr. Isaac Li and Dr. Abdallah Mohammed for the opportunity to work on this cell counting project and for helpful discussion and support during the directed study.