CELL DETECTION AND COUNTING THROUGH DIFFERENT APPROACHES TO DEVELOP A RELIABLE ALGORITHM

Shivang Dogra(18BCE0200), Lakshit Dhanuka(18BCE0217), Tushar Takshak(18BCE0203)

School of Computer Science and Engineering
Vellore Institute of Technology, Tamil Nadu

Abstract **- The objective of this project is to develop an algorithm that can be conveniently used as safe, fast and reliable method for human cell detection. We are trying to develop an automated cell image segmentation algorithm to get improved cell image segmentation with respect to cell boundary detection and segmentation of the clustered cells for all cells in the field of view in negative phase contrast images. Study all these previous algorithms and researches and compile them along with a more efficient algorithm to detect and count the red and white blood cells so that medical professionals are able to detect the abnormalities in one's body through any discrepancies found in the amount of cells in one's body. We will be counting all the cells irrespective of their type except for the boundary cells i.e. incomplete cell from the image.**

Keywords **— blood cell count, thresholding, boundary cell detection, Mahotas Module.**

I. INTRODUCTION

In recent years the IP mechanisms are used widely in several medical areas for improving earlier detection and treatment stages, in which the time factor is very important to discover the disease in the patient as possible as fast, especially in various cancer tumours such as the lung cancer. If these tests take long it can lead to consequences. Many of such tests are based on number of cells present in the given sample and hence cell counting must not be a slow process. Cell counting is fundamental and critical to numerous biological experiments. the development of cell image segmentation algorithms with high robustness and accuracy is attracting more and more attention. In this study, an automated cell image segmentation algorithm is developed to get improved cell image segmentation with respect to cell boundary detection and segmentation of the clustered cells for all cells in the field of view in negative phase contrast images. A new method which combines the thresholding method and distance transformations was proposed to optimize cell boundary detection. In order to segment clustered cells, the geographic peaks of cell light intensity were utilized to detect numbers and locations of the clustered cells. Automated imaging can greatly speed up the cell counting process while reducing manual labour and human errors. The system can count cells automatically, allowing users to have walk-away time. Today many technologies are

there to count number of cells fast and accurately, but we are discussing a method based on IP using python. The objective of this project is to develop an algorithm that can be conveniently used as safe, fast and reliable method for human cell detection. Our efforts would be heavily based on making the algorithm easy to use and ready for the end-user. We will be counting all the cells irrespective of their type except for the boundary cells i.e. incomplete cell from the image. The successful implementation of automated techniques relies in the adjustment of cell staining, image display parameters and cell morphology. We used several of the IP method on the sample image in order to obtain the result i.e. number of cells present in the sample image.

II. LITERATURE REVIEW

A. INITIAL REVIEW

Firstly, our group started finding research papers related to our topic. We analyzed all the methods and ways to find the blood cells count. We decided objective of a project to be is to develop an algorithm that can be conveniently used as safe, fast and reliable method for human cell detection. Our efforts where be heavily based on making the algorithm easy to use and ready for the end-user. We counted all the cells irrespective of their type except for the boundary cells i.e. incomplete cell from the image. A new method which combines the thresholding method and distance transformations was proposed to optimize cell boundary detection.

B. LITERATURE SUMMARY

For final review we have composed all the methods and started to implement it. We have used python as our base programming language. And we collected samples related to blood cell counting. We used spyder as our IDE. We utilized modules like Mahotas, Numpy, Pyplot which is a part of matplotlib, Interact and fixed which are a part of Ipython. We tried to doing the most basic thing, reading an image and printing it: (mahotas ships with an image of nuclei for our analysis). Initially the sample image is loaded into a variable named “dna”. We will be doing pre-processing using various image processing techniques. Once we are done with pre-processing we’ll be removing boundary incomplete cells present in our image as mentioned in objective. Now the only thing left to be done is to remove small cells because they are not significant enough to be counted. We can now count number of cells present in the image and our project is over.

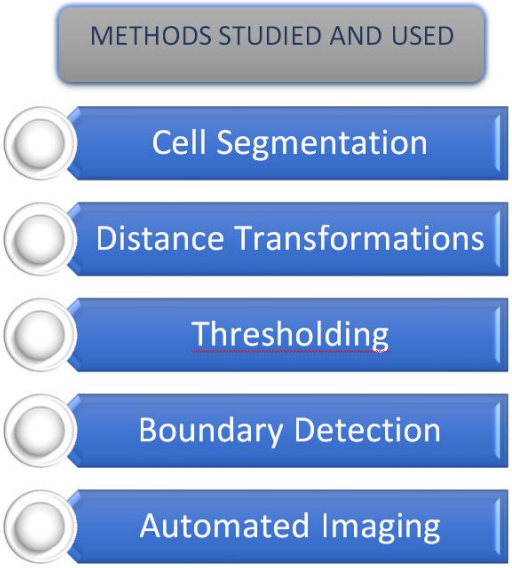
We are trying to develop an automated cell image segmentation algorithm to get improved cell image segmentation with respect to cell boundary detection and segmentation of the clustered cells for all cells in the field of view in negative phase contrast images.

Study all these previous algorithms and researches and compile them along with a

III. PROPOSED METHOD

The objective of this project is to develop an algorithm that can be conveniently used as safe, fast and reliable method for human cell detection.

more efficient algorithm to detect and count the red and white blood cells so that medical professionals are able to detect the abnormalities in one's body through any discrepancies found in the amount of cells in one's body. We will be counting all the cells irrespective of their type except for the boundary cells i.e. incomplete cell from the image.



A. LIBRARIES USED

The tools needed for IP in python are not given under any one specific module, rather they distributed among several modules. Fortunately, they all work on the same data representation, the numpy array. So, first of, we start by installing and importing the modules that are required.

Modules needed:

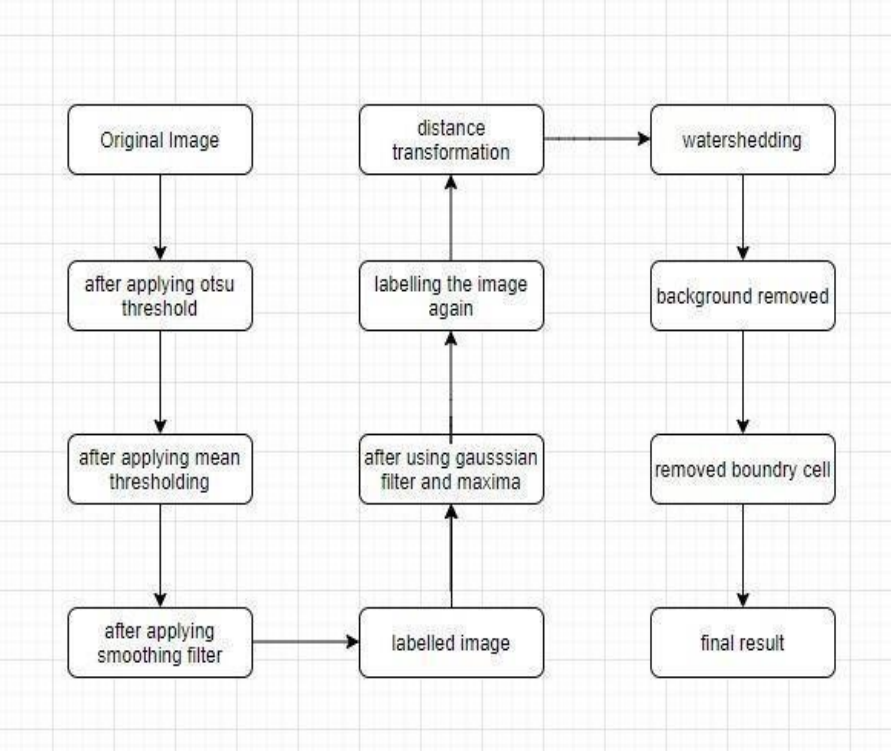
- Mahotas
- Numpy
- Pyplot which is a part of matplotlib
- Interact and fixed which are a part of Ipython

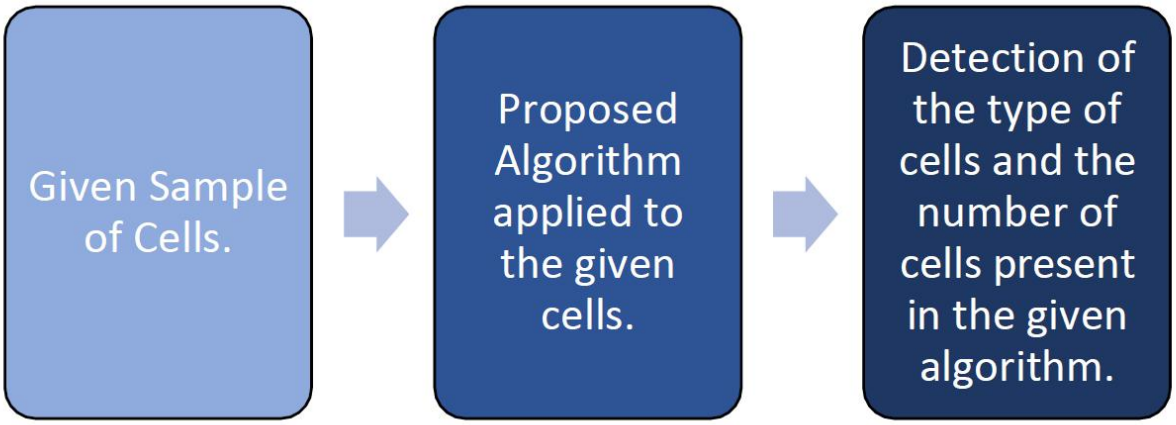
B. PROPOSED ALGORITHM

1. Initially the sample image is loaded into a variable named "dna". The image is resized to our requirements using figure. figure size function of Pyplot.
2. As the standards go, the image is converted to greyscale using **plt.gray()** function. Now we limited the number of axis to 2, this means we converted the image to a 2D matrix. Then we printed all the results.
3. Now that we have our image and applied resizing to fit the viewing window, we can go forward to do some processing to achieve out final goal. We try using the standard **"otsu" thresholding**. But it does not work so well on this data. What does work well is thresholding by the mean.
4. The image obtained after average thresholding is not so clean and is very noisy around the edges of the nucleus. We tried resolving this problem by applying a simple gaussian blurring filter (smoothening filter).
5. Now we have a binary image, but it is useful to label the image in order to assign a different index to each component. We do this by using the **label ()** function under mahotas module.

6. We applied **gaussian filter** again with sigma value equal to 12 and dilating the contrast stretched region max image.
7. We applied **distance Transform** in order to obtain the derived representation of a binary image, where the value of each pixel is replaced by its distance to the nearest background pixel.
8. After distance transformation we applied **water shading** in order isolate objects in the image from the background. Once we are done with water shading we'll be removing boundary incomplete cells present in our image as mentioned in objective.
9. Now the only thing left to be done is to remove small cells because they are not significant enough to be counted. We are now done with all the pre-processing required and our image is now ready to be labelled for one last time.
10. We can mow count the number of cells present in the image and our project is over.

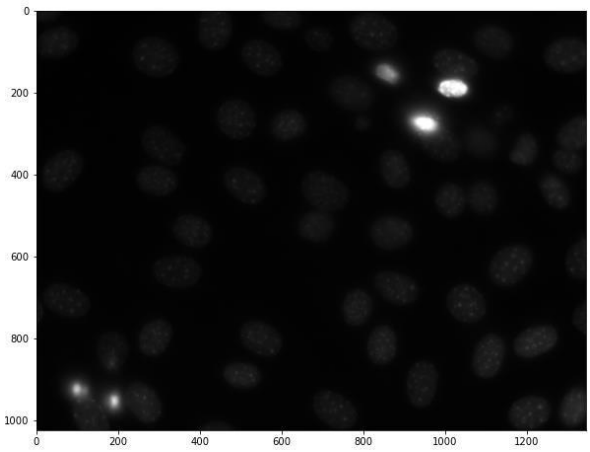
D. ARCHITECTURE

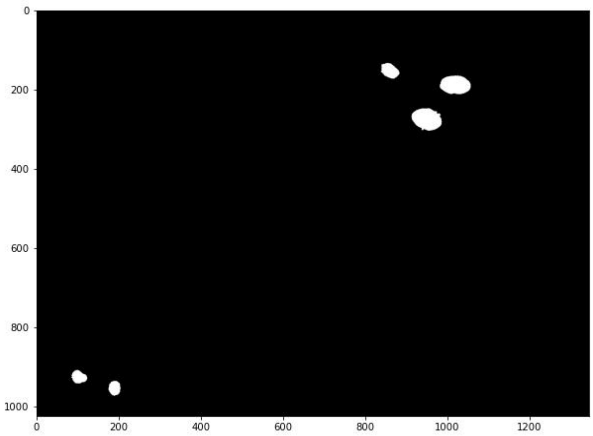


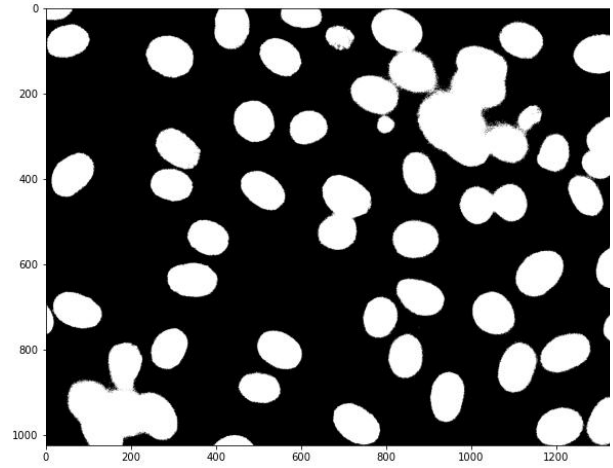


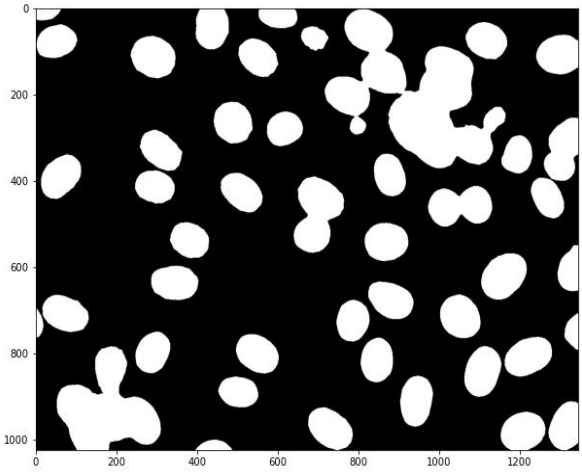
IV. RESULT AND ANALYSIS

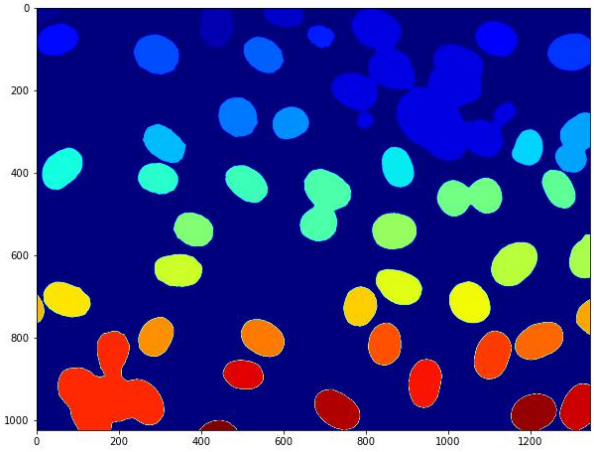
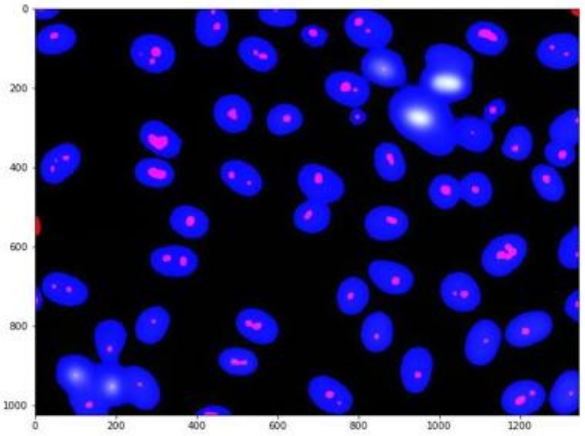
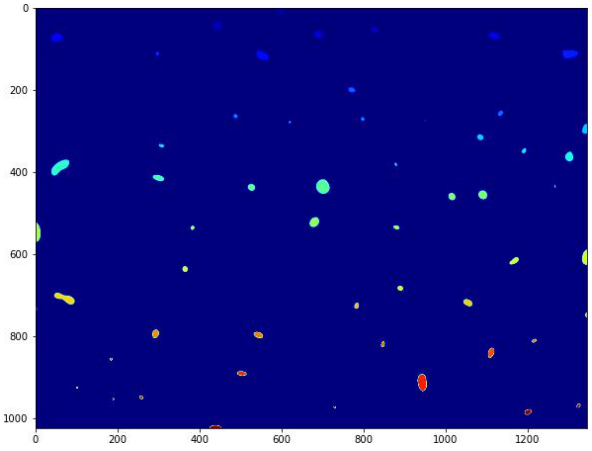
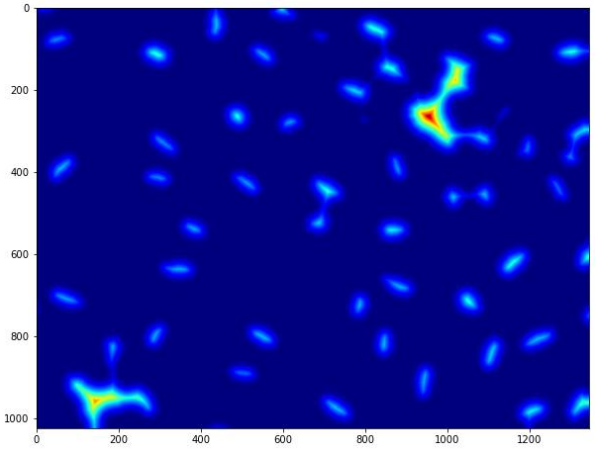
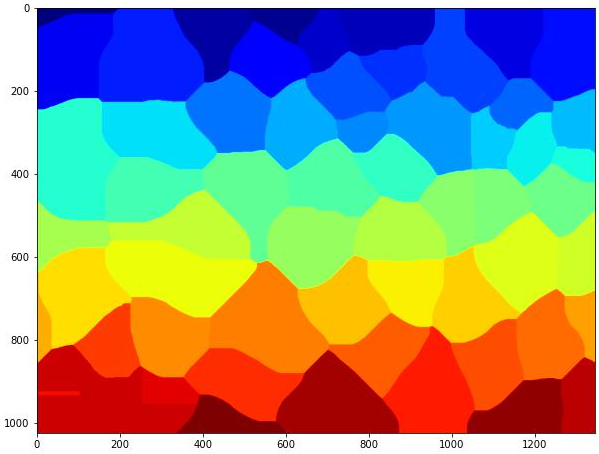
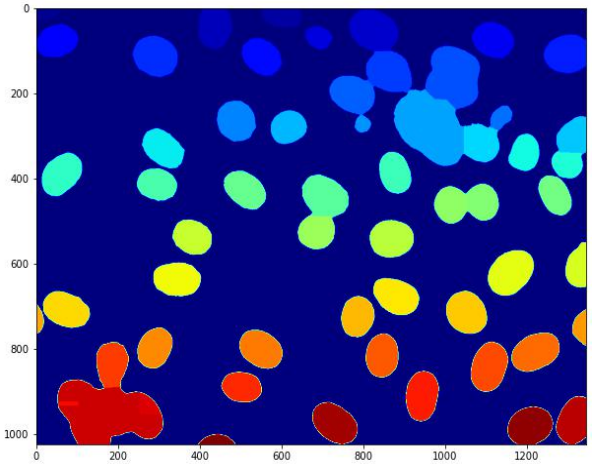
A. EVALUATION AND OUTCOMES

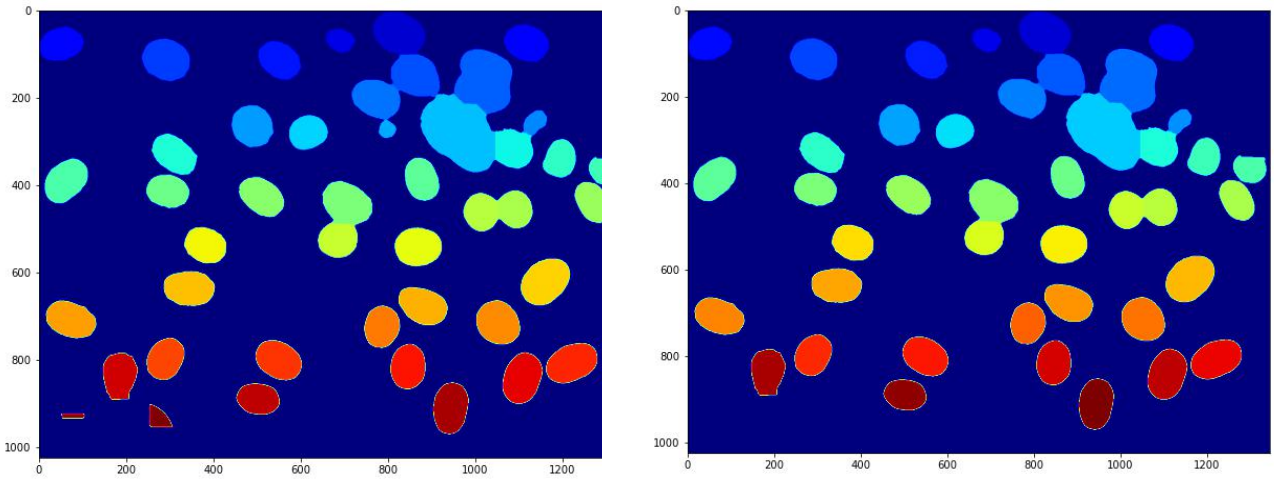






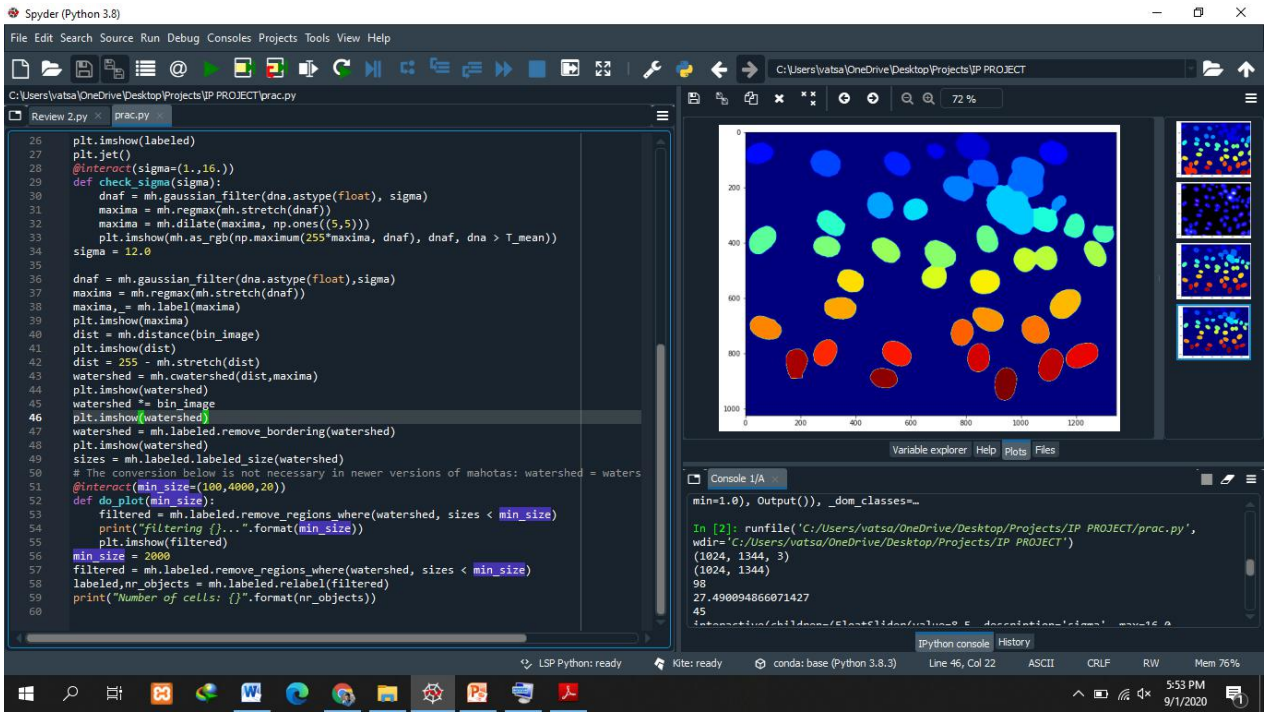


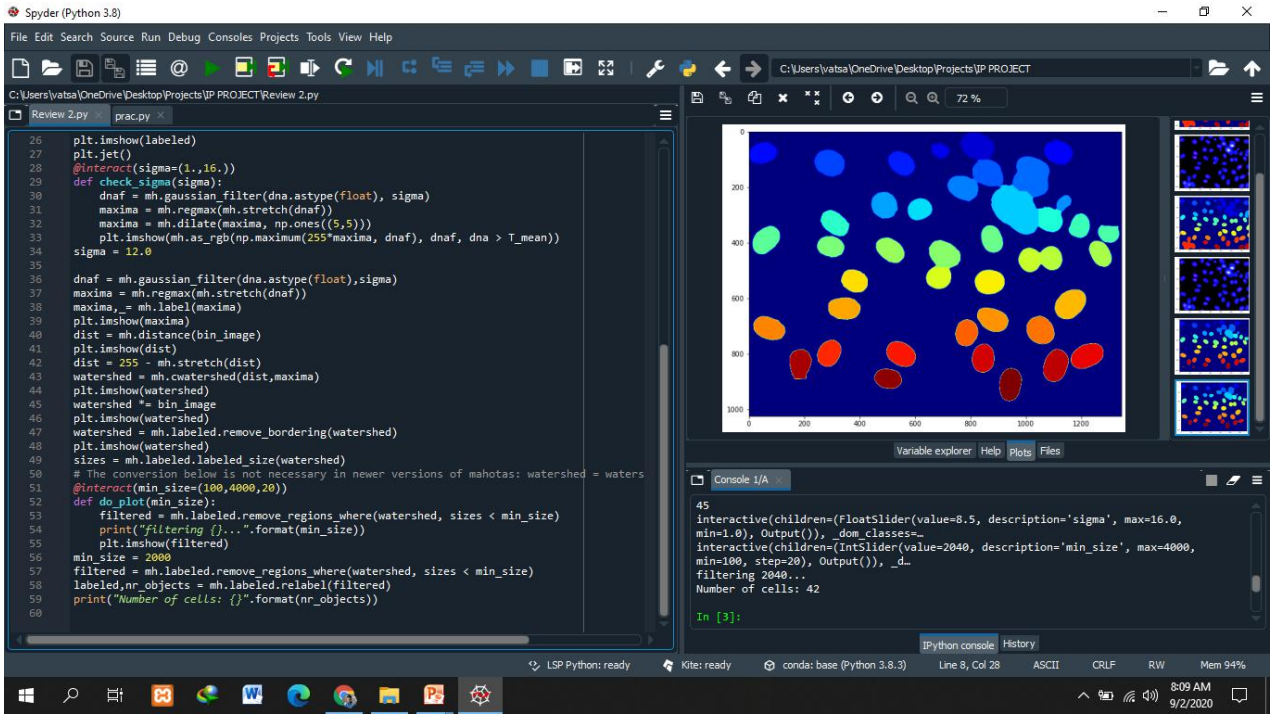









B. OUTPUT SCREEN

Link for code: https://rb.gy/p3iji7





C. TESTING RESULTS

Manual Count without Image processing techniques: **45 Cells**

```
(1024, 1344, 3)
(1024, 1344)
98
27.490094866071427
45
```

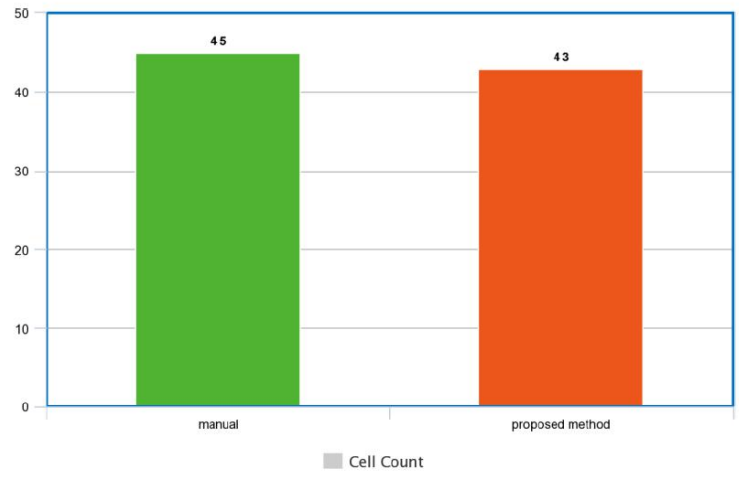
Automated Count after Image Processing: **42 Cells**

```
filtering 2040...
Number of cells: 42
```

D. ANALYSIS AND PLOT

Table: Comparison of the Result between the proposed method and the manual method

Image Samples	Cells counted manually	Cells counted using our proposed method
a	45	43



Graphical Representation of the results

Expected Output:

The output is expected to be in the range of 40-45 cells rather than 45 (manual count).

Test Accuracy: 95%-98%

Output Received:

The output received is 42 cells using our algorithm.

Test Accuracy: 96%

It is observed that the results obtained by the proposed method offer a good conformity with the manual counting method. In our method, we have left out the cells that are not totally in the image field. However, in a real blood test where blood count is done manually, the practice is to count the cells on two adjacent edges of the image field and take each cell as one irrespective of how much of it is in the image field. It is assumed that two opposite sides have same number of such cells. As this edge correction has not been considered, in each of the samples the count values by the proposed method are slightly less than the count values obtained manually. The accuracy of our method is 96%.

V. CONCLUSION

This project presents a methodology to achieve an automated detection and counting of red blood cells in microscopic images. Results indicate that the counting of red blood cell in microscopic images offer remarkable accuracy. Laser-based citometers are available to count blood cells, but they are not image based and destroy the blood samples during the analysis. Also, installation of such system is very costly. Our proposed method is very cost-effective and can be easily implemented in medical facilities anywhere with minimal investment in infrastructure. It can also identify overlapping blood cells and count them separately. It is also very time effective as manual counting is a very tedious job and time consuming. However, the software must be modified to count the effective number of RBC's which are partly in the image fields to obtain more accurate result. We were successful in developing a reliable algorithm for counting the number of cells in the image.

VI. FUTURE SCOPE

The above implemented algorithm based on image processing techniques like segmentation and thresholding has currently been implemented with Mahotas Library. This algorithm process can further be implemented on various other image datasets like Kaggle dataset.
This algorithm proves to be very effective in our current project and will prove the same for other dataset

VII. REFERENCES

[1] An Automatic Nuclei Cells Counting Approach Using Effective Image Processing Methods
Mogeeb A. A. Mosleh; Abdul Aziz AL-Yamni; Abdu Gumaei

[2] Blood Cell Count using Digital Image Processing Varun D Dvanesh; Priya S. Lakshmi; Kandluri Reddy; Abirami S Vasavi

[3] Blood Cell Counting and Segmentation Using Image Processing Techniques