DETERMINATION OF BLOOD CELLS USING IMAGE PROCESSING

Submitted in partial fulfilment of the requirements for the award of

Bachelor of Technology degree in Biomedical Engineering

By

ANGELINE J (37240005)

GAYATHRI P (37240024)

DEPARTMENT OF BIOMEDICAL ENGINEERING

SCHOOL OF BIO AND CHEMICAL ENGINEERING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

MARCH-2021

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with "A" grade by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI – 600 119
www.sathyabama.ac.in

DEPARTMENT OF BIOMEDICAL ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **ANGELINE J (37240005)** and **GAYATHRI P (37240024)** who carried out the project entitled **"DETERMINATION OF BLOOD CELLS USING IMAGE PROCESSING"** under my supervision from September 2020 to March 2021.

Mr. G. UMASHANKAR, M.Tech., (Ph.D.)
Internal Guide

Dr. T. SUDHAKAR, M.sc., Ph.D.,
Head of the Department

Submitted for Viva voce Examination held on 09.04.2021

Internal Examiner **External Examiner**

DECLARATION

We**, ANGELINE J (37240005)** and **GAYATHRI P (37240024)**, hereby declare that the Project Report entitled **"DETERMINATION OF BLOOD CELLS USING IMAGE PROCESSING"** done by us under the guidance of **MR. G. UMASHANKAR,** M.Tech., (Ph.D.) is submitted in partial fulfilment of the requirements for the award of Bachelor of Technology degree in Biomedical Engineering.

Angeline.J

P. Gayathri.

DATE: 09.04.2021

PLACE: CHENNAI

SIGNATURE OFTHE CANDIDATES

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to **Board of Management of SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. J. Premkumar**, M.Sc., Ph.D., and **Dr. T. SUDHAKAR,** M.Sc., Ph.D., **Head of the Department, Dept. of Biomedical Engineering** for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide **Mr. G. UMASHANKAR,** M.Tech., (Ph.D.), **Department of Biomedical Engineering** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

We wish to express our thanks to all **Teaching and Non-teaching** staff members of the **Department of Biomedical Engineering** who were helpful in many ways for the completion of the project.

We thank our **parents** and **friends** for their utmost support and motivation throughout the completion of this project.

ABSTRACT

In medical field, blood testing is one of the most clinical examination tests. In laboratories, counting of blood cells is important for diagnosis of disease in patients. There are many diseases like bone marrow failure, anaemia, leukemia, etc. that can affect the count and type of blood cells. Therefore, understanding the counting of blood cells in our bloodstream is important. Old methods for blood count were by using Hemocytometer and microscope which are time consuming process and requires manual counting of blood cells which leads to inaccurate results (Akshaya Sahastrabuddhe 2016). There are some expensive machines like analyser which are not affordable by the patients as well as some of the clinics. In this project, we are proposing a method in which we have used image processing technique for detecting the blood cells in less time. This technique helps in counting of blood cells in blood smear image based on their morphological factors like shape and size of RBC and WBC and Platelets. In order to count the overlapped RBC cells, we have used Circular Hough Transformation (Gulpreet Kaur Chadha et.al 2019). One important factor is the waiting time for the reports. Nowadays in hospitals and laboratories the waiting time for getting blood results or the reports is most commonly 24 hours to 8 days in case of severity of disease. Patient's waiting time should be less as possible and the treatment should be started immediately. The other important factor is the cost. In some healthcare fields, the pathological tests are more expensive. These factors lead to loss of patient's life. To overcome these problems, we can use this technique to count the blood cells. We have used python open CV as the image processing software in this paper.

Keywords – Hemocytometer, Analyser, Circular Hough Transform, OpenCV, Image Processing.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

MATLAB	**MAT**rix **LAB**oratory
IDLE	**I**ntegrated **D**evelopment and **L**earning **E**nvironment
OpenCV	**Open**-source **C**omputer **V**ison library
AHE	**A**daptive **H**istogram **E**qualization
CLAHE	**C**ontrast **L**imited **A**daptive **H**istogram **E**qualization
CDF	**C**umulative **D**istributive **F**unction
COPD	**C**hronic **O**bstructive **P**ulmonary **D**isease
MDS	**M**yelo**D**ysplastic **S**yndrome
PNH	**P**aroxysmal **N**octurnal **H**aemoglobinuria
HIV	**H**uman **I**mmunodeficiency **V**irus

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 OUTLINE OF THE PROJECT

The analysis of microscopic image is extremely important in both the medical and computer fields. The complete blood cell count and analysis of blood smears, which is considered the first step in detecting the many blood related diseases like Anaemia, Leukemia, malaria. The pathology is a field of investigation of samples collected from the individual for diagnosis in the medical field. This helps from detection of the pathogens or harmful foreign particles in sample to further treatment of disease. An accurate segmentation and counting mechanism may help diagnose the abnormalities during clinical analysis. The equipment which is currently available for counting the blood cells is Hemocytometer, it is also known as Hematology analysers. This equipment is bulk and highly expensive. It consists of a thick glass microscopic slide and the chamber is engraved with a laser-etched grid of perpendicular lines. By observing a particular area of the grid, it is therefore possible to count the number of cells or particles in a specific volume of fluid, and therefore calculate the concentration of cells in the fluid. A well-used type of hemocytometer is the Neubauer counting chamber. So, this requires a lot of time and skill operators. We wanted fast and effective production of counting the blood cells to make better and affordable diagnosis. When we compared with the manual process, these automated analysers give fast and reliable results regarding the numbers, average size, variation in the size of blood cells. In this we can easily count the WBC and platelets. But we cannot count the abnormal cells, overlapped cells and do not detect the cell shape. So that we need some pre-processing techniques which is called image preparation phase. For overlapped RBC cells, we have used Circular Hough Transform (Gulpreet Kaur Chadha et.al 2019). The different algorithms were employed for the detection and counting of the blood cells. But, using python is an effective method in recognizing and counting blood ccells as a practical alternative to the manual blood cell counting and the use of analysers.

1.2 BLOOD CELLS

A blood cells, also called a hematopoietic cell, hemocyte or hematocyte. It is produced through hematopoiesis and found mainly in the blood. The main types of blood cells are red blood cells (erythrocytes), white blood cells (leuocytes), and platelets (thrombocytes). These three kinds of blood cells are totally 45% in the blood, with the remaining 55% of the volume composed of plasma. Plasma is the liquid component of blood. Our blood also contains Haemoglobin which is an iron containing protein. It gives red blood cells colour and it helps the transportation of oxygen from the lungs to tissues and carbon dioxide from tissues to the lungs to be exhaled. The RBC (erythrocytes) shapes like circular, biconcave, disk shaped and this allow them to squeeze through narrow capillaries. The normal RBC count is 4.5 to 5 million cells per microlitre for males and 4.2 to 5.4 million cells per microlitre. RBCs have a lifespan is approximately 100 to 120 days. White blood cells (leucocytes) are irregular shape and help in the immune system of the body. It involves in defending the body against infectious disease and foreign materials. They are produced from the bone marrow. Basophils, eosinophils, neutrophils, mast cells, lymphocytes and monocytes are the types of WBC cells. The normal count of WBC cells for male is 5000 – 10,000 cells/mcL and for female is 4500 – 11,000 cells/mcL. Platelets are very small, irregularly shaped clear cell fragments, 2-3 micrometer in diameter. The average lifespan is5 to 9 days. Its main work is to form blood clots. It produces thread like fibres to form these clots. They control bleeding. The normal range is 1,50,000 to 4,50,000 per cubic millimetre. If the blood cells count is increased or decreased means the individual have a disease.

1.3 PYTHON (OpenCV)

Python is an interpreted, high level and general-purpose programming language. It designs and emphasizes code readability with its notable use of significant indentation. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small- and large-scale projects. IDLE is an integrated development environment for python, which has been bundled with the default implementation of the language. Open CV is a library of cross platform programming functions aimed at real time computer vision. It is used to process images.

CHAPTER 2

LITERATURE SURVEY

Guda Sai Manisha et al. (2020) proposing a method the automated analysers. The USB compatible microscope is used in this process. The microscope image is capture; the image is stored in the system via the USB compatible microscope. So that the stored image is processed through software MATLAB to enhance the quality of the image for accurate output data, which we compare with our known values and given efficiency of the acquired output. In this MATLAB and scilab software is used. The accuracy for RBC, WBC and platelets are 90%, 99%, 86% respectively.

Varun Dyanesh et al. (2018) In this paper they find out the RBC and WBC count using the digital image processing (DIP). The blood smear images captured through a compound microscope. So, the image is done by the image processing method and followed by the edge detection and morphological operations. The overlapped, unshaped blood cells are found out the algorithms used circular Hough transformation. So, the accuracy of this project is 91% of RBC and 855 of WBC cells.

Gulpreet Kaur Chadha et al. (2019) In this paper the different types of red blood cells are counted to diagnose the disease in individual. The red blood cells are very important count in individual for diagnose a disease. In this the images are classified on the basis of colour, texture, and morphology. In this paper there are mainly three steps are followed: feature extraction using morphology, thresholding segmentation and circular Hough transformation. This algorithm achieves overall accuracy is 91.66% and it takes the time only 0.81432 seconds to count the RBC.

Vibhute et al. (2019) The blood cell count is very essential for diagnosing a disease. In this method they followed image acquisition, pre-processing, image sweetening, image segmentation, image post-processing and counting algorithm. The MATLAB software is used in this project. The accuracy of this paper is 91%.

Manasi dixit et al. (2020) There are various techniques used for counting the blood cells. In this paper gives a review on different techniques to count the blood cells. The process is image acquisition, segmentation, counting algorithm. By using these techniques, the blood cells are counted automatically using the software. The software is MATLAB watershed algorithm, K–mean algorithm, edge detecting algorithm. In this paper the main disadvantage is inaccuracy in WBC counting.

Bhong et al. (2019) In this paper the RBC and WBC are counted using image processing. So, this method is very cost effective and accuracy. The circular hough transformation is used for counting the RBC cells. The overall accuracy is 99.8%.

CHAPTER 3

AIM AND OBJECTIVE

3.1 AIM

In the medical field, blood test is one of the most essential tests. There are many types of disease like anemia, leukemia, bone marrow failure that can affect the count of the blood cells. The old method of the blood count is hemocytometer. This method is very expensive and inaccuracy because of the manual work (Akshaya Sahastrabuddhe 2016). In this project, we are proposing a method in which we have used image processing technique for detecting the blood cells.

3.2 OBJECTIVE

In this paper, counting the blood cells using the software which is the python idle using open CV method. This technique helps in counting the blood cells in blood smear image based on their morphological factors like shape and size of RBC, WBC and platelets. We have used algorithms that works on the images captured by a microscope with considerable accuracy. In image processing, we have done image acquisition, pre-processing, segmentation and counting of the blood cells (Shweta Kamble et.al 2020). The web camera attached with the electronic microscope is used in this process. The captured image is processed through some of the image processing techniques. This includes the median and Gaussian filters and the canny edge detection in image processing. The WBC and platelets are easily counted. The overlapped RBC cells are counted using the Circular Hough Transformation (Gulpreet Kaur Chadha et.al 2019). The system can be further improvised for detecting various disease related to different blood cell morphologies. This helps us to prevent death of an individual due to some sort of disease.

3.3 EXISTING METHOD

The hemocytometer is the counting chamber is used for counting the blood cells. It was invented by Louis Charles Malassez. It consists of a thick glass microscope slide with rectangular indentation that creates a precision chamber. This is covered with the laser etched grid of the perpendicular shape lines. The mainly used type of hemocytometer is the Neubauer counting chamber. This chamber is a thick crystal glass slide 30*70mm and 4mm thickness. So, the central area is for the counting the cells. This chamber grid size is 3mm*3mm. In the blood cell counting the corners are used for the WBC counting and the central place is used for the RBC and platelets because WBC concentration is lesser than the red blood cells. In this the squares are divided into 25 squares of width 0.2mm. These 25 central squares are subdivided into 16 small squares, so that the central square is made of 400 small squares.

FIG 3.1: Neubauer Chamber

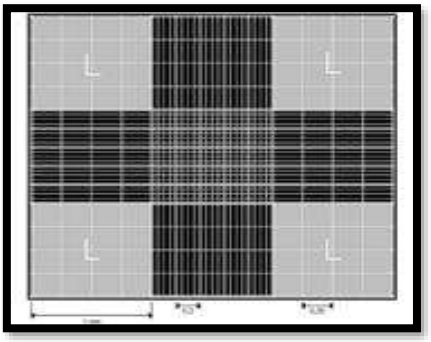

FIG 3.2: Cell count in one of the 9 big square

Place the glass cover on the Neubauer chamber centre area. The surface is should be flat for workbench place. Place a sample with the help of micropipette. The sample is must without the bubbles. If the bubbles are appeared means, the result is not accurate. The next step is to be microscopic stage. Looking for the first counting grid square for cell count will start. And then the next step is to concentration calculated. In this method there will be a maximum chance to getting error. The error has occurred, because of the pipette, mathematical calculation, volume of sample.

3.4 PROPOSED METHOD

The image processing is performed using the Python programming language with the most frequently used OpenCV library. The OpenCV is a big open-source library for the image processing. Python is a high level, general purpose language. This is an object-oriented language and the main aim is to get a clear and logical coding. The full form of IDLE is an integrated development environment for python. In image processing, image acquisition, pre-processing, segmentation are the main steps. Image acquisition is a process of retrieving the images from some sources. This is the first and main step in image processing. In pre-processing step, the filter works are done. The median and Gaussian filter are used for smoothening. Segmentation is the main process while doing the image processing. It is the process of differentiating the upload image of cells into RBC, WBC and platelets based on their morphological factors like shape and size. The last step of the image processing is the counting the blood cells. From the segmented image, the WBC and platelets are easily counted because, there is no overlapping in those cells. But the RBC cells may be overlapped, for finding the overlapped RBC cells we will use the circular Hough transformation (Gulpreet Kaur Chadha et.al 2019). Different algorithms were employed for the detection and counting of the blood cells.

CHAPTER 4

METHODOLOGY

4.1 IMAGE CAPTURING METHODOLOGY

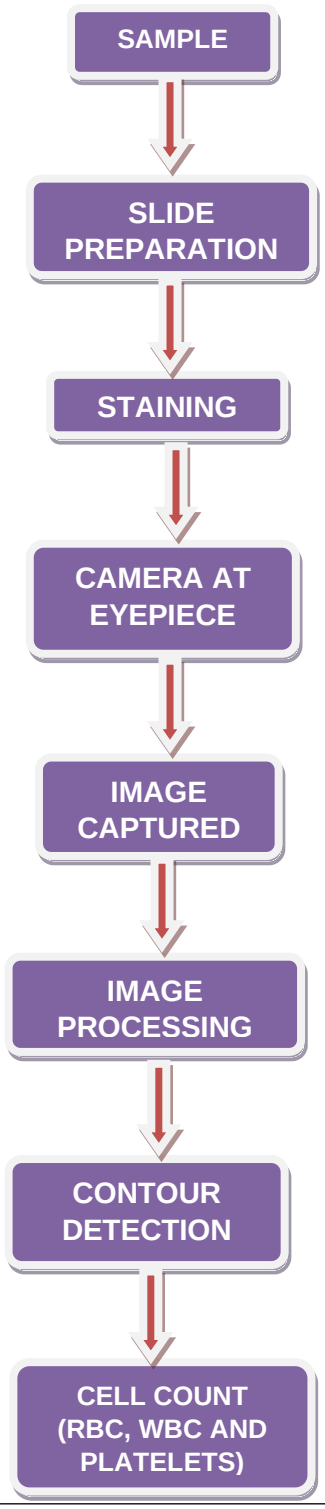

FIG 4.1: Block Diagram of whole process

4.2 BLOOD SAMPLE

Blood collection helps for the identification of metabolic, respiratory, mixed acid-based disorders. Take the clean slides and wipe it with the 70 to 90% alcohol and allow to dry. We do not touch the surface of the smear slide, where the blood smear will be made. Select the finger to puncture, usually the middle or ring figure. Clean the area to be punctured with 70% alcohol and allow to dry. Puncture the ball of the finger. Ignore the first drop of blood with clean gauze. If a blood does not well up, gently squeeze the finger. The 10 blood samples were collected from the patients. 5 male blood sample and 5 female blood sample.

FIG 4.2: 10 blood samples collected from patient

FIG 4.3: Purple – Male, Green - Female

4.3 SLIDE PREPARATION

4.3.1 Blood Smear

Blood smear is the main diagnostic method for differential count of RBC, WBC and platelets. It needs a required technique to create a monolayer of dispersed cells. That distribution reflects the concentration of cells in the blood. So that blood film must be made immediately after collecting the blood. Because the morphology of cells deteriorates rapidly after the collecting of samples. The most common technique is wedge or push technique. A small amount of blood is placed on the centre line at the end of the glass slide. Then the second slide is place over the smear slide and it forms an angle of 30 - 45º on the side of the drop.

FIG 4.4: Sample placed on the slide

Left – male, right - female

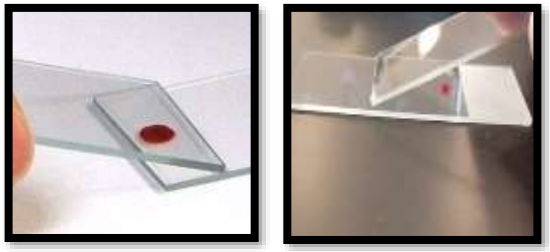

FIG 4.5: Second slide is placed on the smear slide at 45°

Left – male, right – female

The second slide is pulled back into the blood drop. So that spreads along with edges. And the push the slide in opposite direction to the end of the slide. The last push must be done with the fast movement and minimal downwards pressure. The smear must be 2/3 (approximately) of the length of the slide.

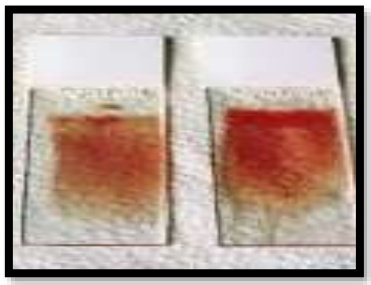

FIG 4.6: Blood Smear

Left – female, right – male

4.4 STAINING

Leishman staining is used for the staining the blood smears. It is usually used to differentiate the blood cells. It is a neutral stain for the blood smears. This stain is invented by the W. B. Leishman (1865 – 1926). This stain is based on the methanolic mixture of polychromed methylene blue and eosin. This was prepared in alcohol medium and diluted with buffer or distilled water during the staining procedure. This stain is in the form of a dry powder, then later reconstituted with the methanol.

FIG 4.7: Leishman Stain

4.4.1 PROCEDURE

Cover the well dried blood smear with the undiluted Leishman stain solution by counting the drops (6 – 10) of Leishman stain. Let it stand for 2 minutes, so that the methanol present in the stain fixes the smear onto the glass slide. After 2 minutes, add the twice amount of the distilled water or the phosphate buffer solution and mix the content by swirling or by bowling gently. Incubate the smear slides for minimum 10 minutes at 37ºC. This will stain the blood cells.

And then rinse the slides thoroughly with distilled water or phosphate buffer solution up to 2 minutes or until it acquires a purple-pinkish tinge. After a minute, flood the slide with water to remove stain. Then wash the slide under the tap water and wiping the back of slide with finger or cotton. Air dry the slides in a tilted position so that the water is easily removed out of the slides. Let it dry in air for few hours and then observe the slide under oil immersion objective lens of the compound microscope.

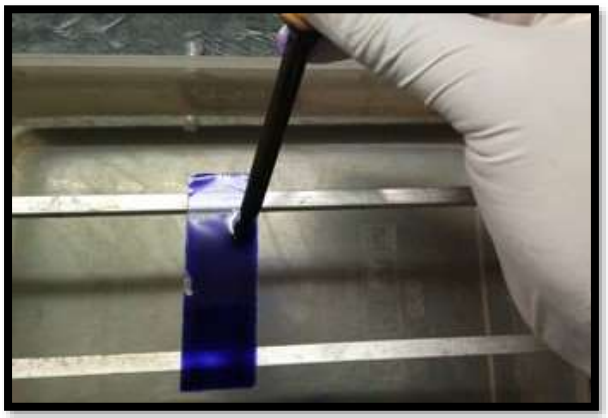

FIG 4.8: Leishman Stain in glass slide

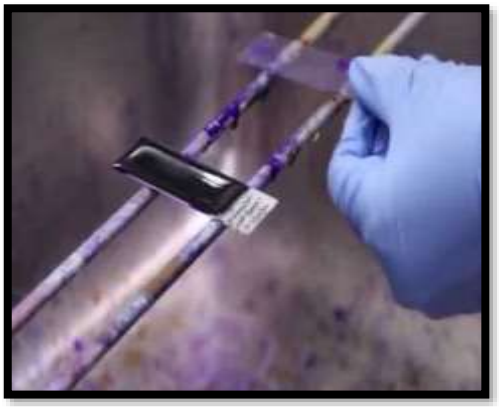

FIG 4.9: Washing the stained-glass slide

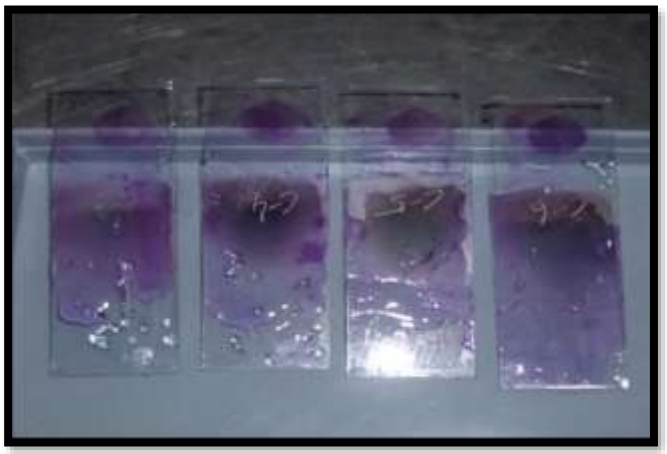

FIG 4.10: Glass slide after staining

4.5. CAMERA AT EYEPIECE (MICROSCOPIC IMAGE)

4.5.1 Web Camera

The web camera is used for capturing the image of blood cells on slide. It is a simple webcam (iballrobo K20) with resolution of 10.2 MP. The camera is connected to the laptop which acquires the image to generate the final result. The field of view of the camera is about 1mm*1mm on the actual glass slide. Slight focusing is required before capturing of image.

Focusing is done by adjusting the microscope so that the clear image will appear on the screen. It involves the qualitative as well as quantitative information about the blood cells. Also, the image processing algorithm depends on the quality of image. The captured image is then processed using image processing technique (Python).

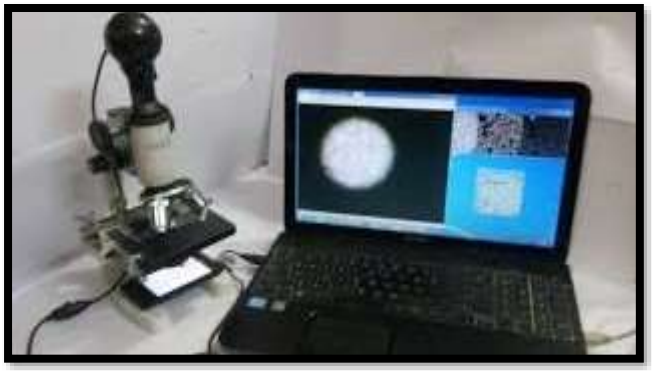

FIG 4.11: Microscope mounted with webcam interfaced with laptop

FIG 4.12: iballrobo K20 webcam

4.5.2 Microscope

The microscope is used for the appropriate magnification of the blood cells. The microscope implemented for this project is a compound microscope with 15X eyepiece and 45X objective magnification. It gives total magnification of 675X. The magnified image of RBC, WBC and Platelets is essential for application of different masks in image processing and detection of closed shapes in an image in order to obtain the number of cells present in the slide which is covered by field of view of the camera placed on the eyepiece.

An appropriate light source is used under the stage of the microscope on which slide with blood smear is placed. This illumination is necessary for observing the cells through the microscope. The microscope used in the pathology labs are digital microscope which are very costly and not easy to afford. But the compound microscope along with digital camera and processing unit together work as the digital microscope which has much less cost.

FIG 4.13: Compound Microscope

4.6. IMAGES CAPTURED

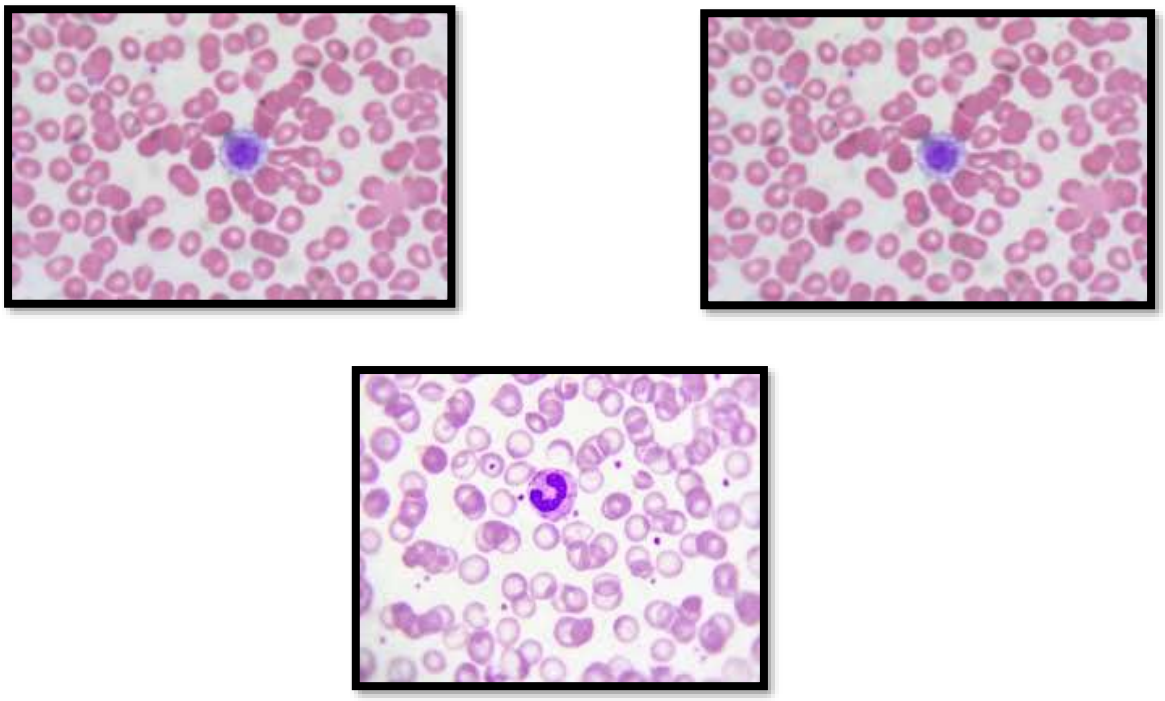

FIG 4.14: Microscopic images of blood smear from male individual

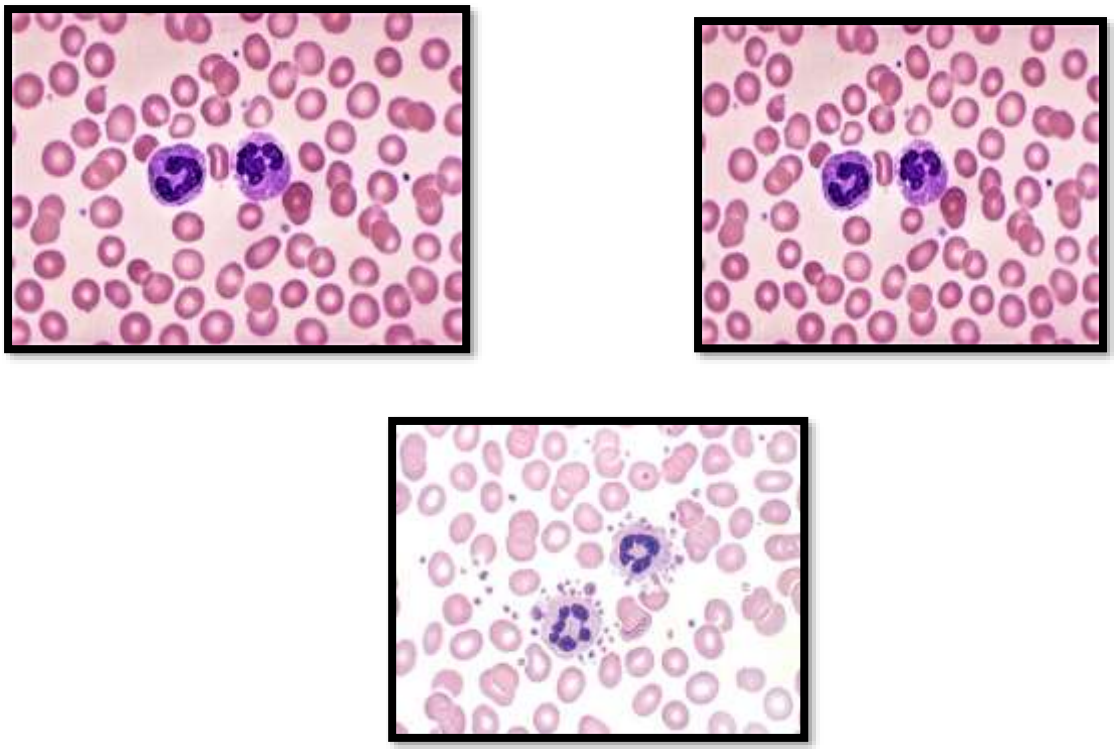

FIG 4.15: Microscopic images of blood smear from female individual

4.7 IMAGE PROCESSING METHODOLOGY

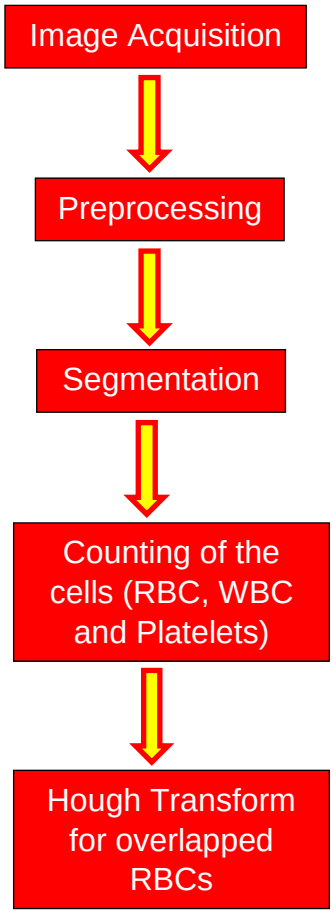

FIG 4.16: Block Diagram for Image Processing

4.7.1 IMAGE ACQUISITION

It is the first step in image processing because without an image, processing is not possible. Image acquisition is defined as the process of retrieving an image from any source especially hardware-based sources for processing. The main advantage of using digital image versus analog image such as a photograph, is the ability to make copies of the images without the loss of quality of an image. Input images are taken from the patients is used for the calculation of patient's blood cells count. These images are uploaded either in jpeg or png format and they are in RGB format. These images are captured by using electron microscope mounted with a webcam which is connected to a laptop. 10 samples (5 male samples and 5 female samples) from the patients are fed (Lokhande et.al 2019).

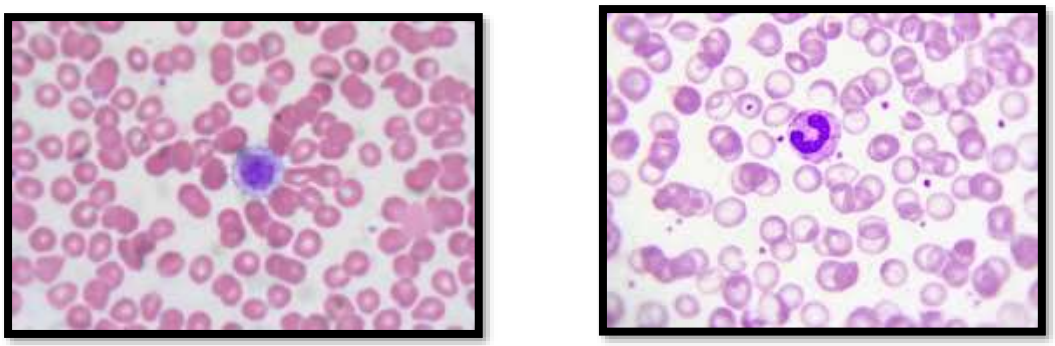

FIG 4.17: Blood smear image of male uploaded for processing

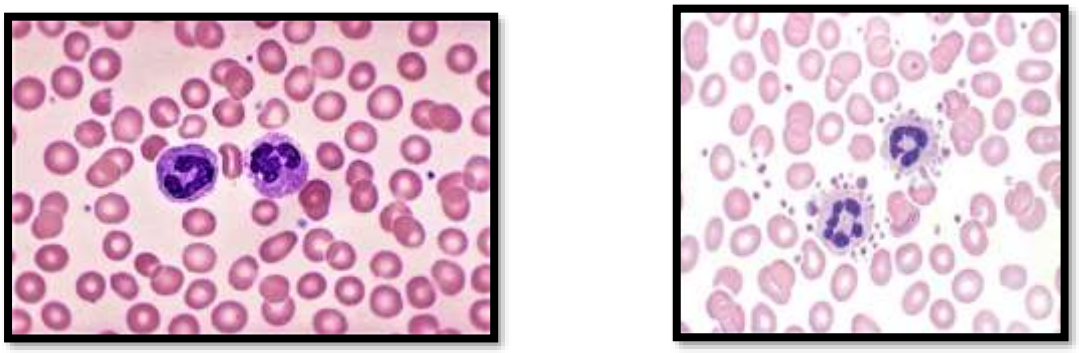

FIG 4.18: Blood smear image of female uploaded for processing

4.7.2 PRE-PROCESSING

Image Enhancement is done in this step. Image enhancement is the process of improving the quality and content of original data and removal of small objects from the image. It adjusts the digital image so that the results are suitable for display or further processing.

Common methods of Image Enhancement include:

- Conversion of RGB to Gray scale
- Filtering
- Histogram Equalization
- CLAHE
- Contrast Stretching
- Edge Detection

4.7.2.1 Conversion of RGB to Gray scale

There are many methods used for converting an RGB image into a grayscale image which includes Average method and Weighted method (Guda sai Manisha et.al 2020).

AVERAGE METHOD

The average method takes the average value of R, G and B as the grayscale value.

$$\text{Grayscale} = (R+G+B)/3$$

Theoretically, the formula is correct. But when writing code, we may encounter overflow error that is, the sum of R, G and B will be greater than 255. To avoid this R, G and B should be calculated respectively.

$$\text{Grayscale} = R/3 + G/3 + B/3$$

The average method is simple but it does not work as expected. This is because human eyeballs react differently to RGB. Our eyes are more sensitive to green light, less sensitive to red light and least sensitive to blue light. Therefore, the three colors should have different weights in their distribution.

WEIGHTED METHOD

This is also called as luminosity method. Weighted method weighs red, green and blue color according to their wavelengths. The formula used is as follows:

$$\text{Grayscale} = 0.299R + 0.58G + 0.114B$$

In grayscale, each pixel is stored as a byte and each pixel value ranges from 0 (black) to 255 (white). Therefore, every pixel has different intensity.

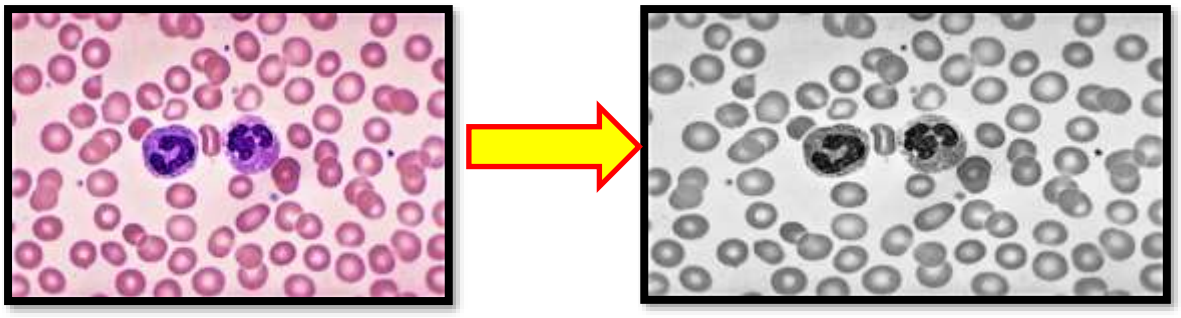

FIG 4.19: RGB image of blood smear

FIG 4.20: converted grayscale image of blood smear

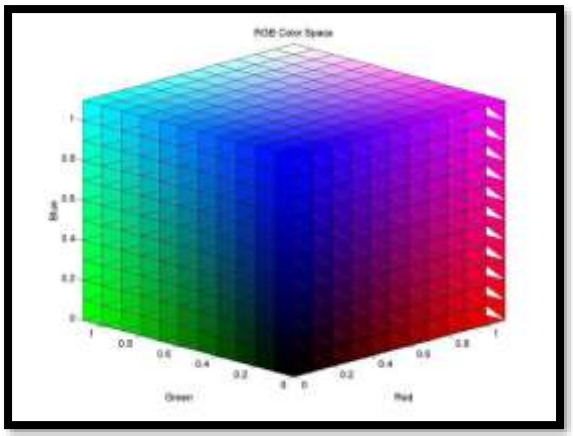

FIG 4.21: RGB Color space

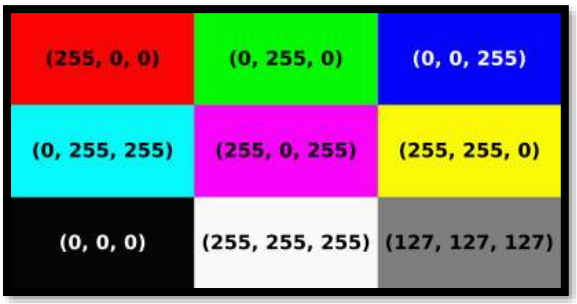

FIG 4.22: RGB Color codes

4.7.2.2 Filtering

Filters are mainly used to suppress images having either high frequencies (smoothening of an image) or low frequencies (detecting the edges in the image uploaded).

An image can be filtered by using frequency domain or spatial domain. Frequency domain is the enhancement which is obtained by applying the Fourier Transform to the spatial domain. In frequency domain, pixels are operated in groups. In Spatial domain, the image space gets divided into uniform pixels according to the coordinates and with a certain resolution.

Filters used in this project are:

- Median filter (smoothening non-linear filter)
- Gaussian filter (smoothening linear filter)

MEDIAN FILTER

The median filter is a non – linear digital filtering technique which is used to remove noise from an image. The principle of median filter is to replace the gray level of each pixel by the median of the gray levels in the neighborhood of the pixels, instead of using average operation. This is widely used in image processing because, it preserves the edges while removing the noise.

At the same time as reducing noise in a signal, it is important to preserve the edges. Edges are important for the visual appearance of the images. Median filters are better than gaussian filters at removing noise and preserving the edges. It is also effective for speckle noise and salt and pepper noise. Because of this, median filters are most widely used in image processing. The median filters work by moving through the image pixel by pixel and it replaces each value with the median value of neighboring pixels.

To perform median filtering at a point of an image, we first sort the values of the pixels in the neighborhood, determine the median and finally assign value to the corresponding pixel in the filtered image.

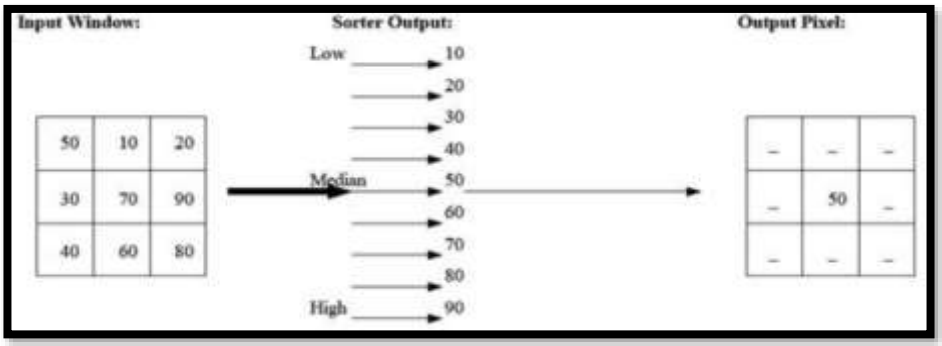

FIG 4.23: Median filter algorithm

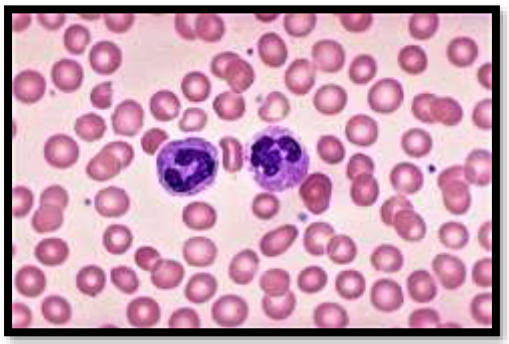

FIG 4.24: Image of Blood smear before applying median filter

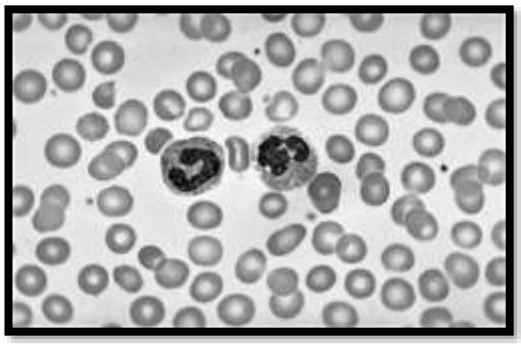

FIG 4.25: Image obtained after applying median filter

GAUSSIAN FILTER

A gaussian filter is a linear filter which is used to blur the image or to reduce the noise. This filter blurs the image with a bell shape represented by the normal distribution.

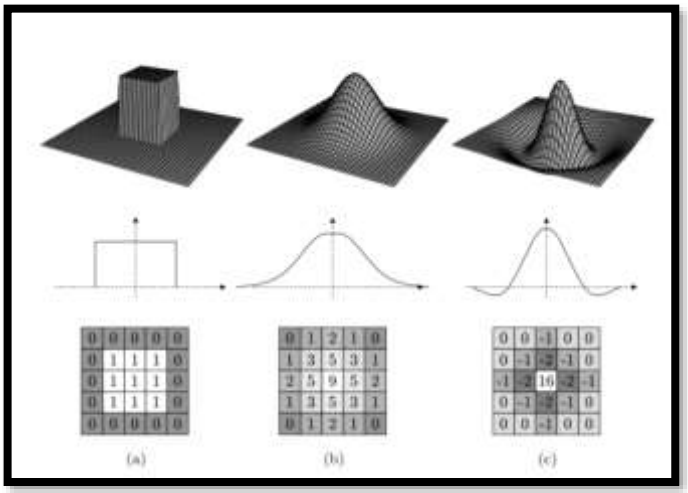

FIG 4.26: Gaussian filter distribution

The Gaussian filters are used in numerous research areas:

- These define a probability distribution for noise or data.
- Used as a smoothening operator.
- Used in mathematics.

The Gaussian filters works in the 2D distribution. This is achieved by convolution of 2D Gaussian distribution function with the image. This theoretically requires an infinitely large convolution as the distribution is non – zero. They vary from negative to positive values. Gauss function is not equal to zero as it is a symmetric function.

The Gaussian filters are a non-uniform filter. The kernel coefficients will diminish with increase in distance from the kernel's centre. The weight of central pixels is higher than those on the periphery. Increase in values of standard deviation produces a wider peak (greater blurring). Kernel size increases with the increase in standard deviation to maintain Gaussian nature of the filter. Gaussian filter coefficients depend on the value of standard deviation. At the edge, the coefficients must be equal to zero.

The first step in edge detection is applying Gaussian filter. The images below had a gaussian filter applied prior to processing. Gaussian filtering is more effective in smoothening images. It has basis in human visual perception system. It has been found that the neurons create a similar filter when processing visual images.

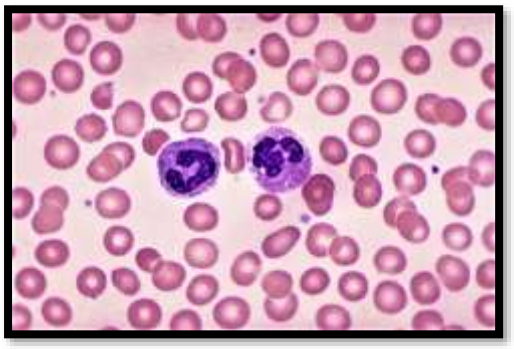

FIG 4.27: Image of Blood smear before applying Gaussian filter

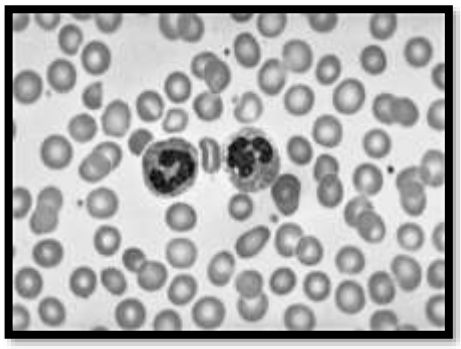

FIG 4.28: Image obtained after applying Gaussian filter

4.7.2.3 Histogram Equalization

The histogram of an image is defined as distribution of its discrete levels in the range [0, L-1].

Histogram equalization is a method to process images to adjust the contrast of an image by modifying the intensity distribution of the histogram. It is not necessary that contrast will always be increasing. There are cases where histogram equalization can be worse. In those cases, the contrast will be decreased. The objective of histogram equalization is to give linear trend to the Cumulative Distributive Function (CDF) which is associated to the image. During this process, the overall shape of the histogram changes.

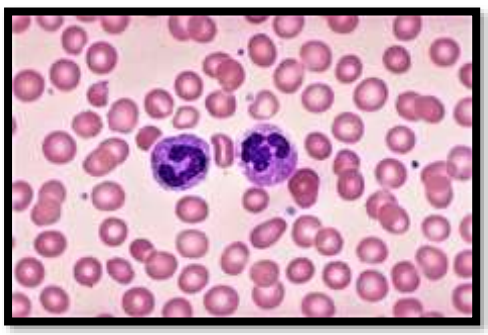

FIG 4.29: Image of Blood smear before applying histogram equalization

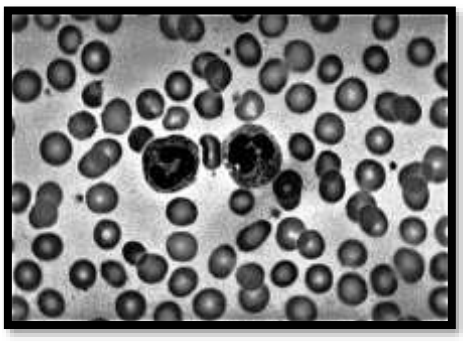

FIG 4.30: Image obtained after applying histogram equalization

4.7.2.4 CLAHE – Contrast Limited Adaptive Histogram Equalization

Adaptive Histogram Equalization (AHE) is an image processing technique which is used to improve contrast. It is suitable for improving the contrast and enhancing the definitions of edges in each region of an image. AHE has a tendency to over amplify the noise in homogenous regions of an image.

CLAHE is a variant of AHE which is responsible for over amplification of the contrast. CLAHE is performed on small regions in the image, called as tiles (Navid Shaikh et.al 2019). The neighboring tiles are then combined by using bilinear interpolation technique to remove the artificial boundaries.

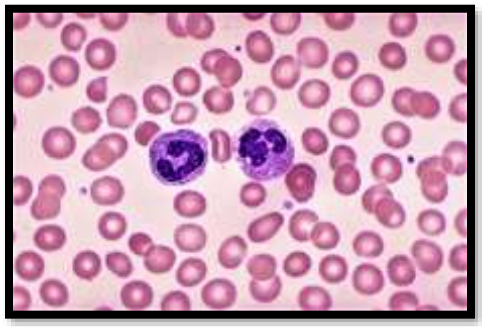

FIG 4.31: Image of Blood smear before creating CLAHE

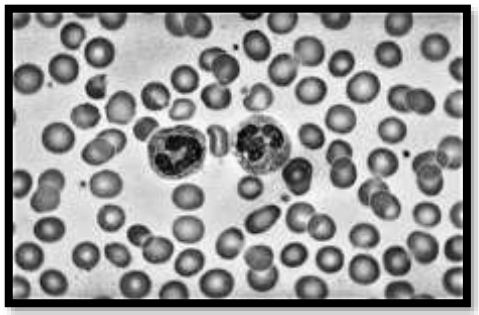

FIG 4.32: Image obtained after creating CLAHE

4.7.2.5 Contrast Stretching

Contrast stretching is a technique to improve the contrast in an image by stretching the range of intensity values. This can only apply a linear scaling function to the image pixel values.

Before stretching it is important to specify upper- and lower-pixel value limits over which the image has to be normalized. For example, for an 8-bit gray level images the lower and upper limits are 0 and 255.

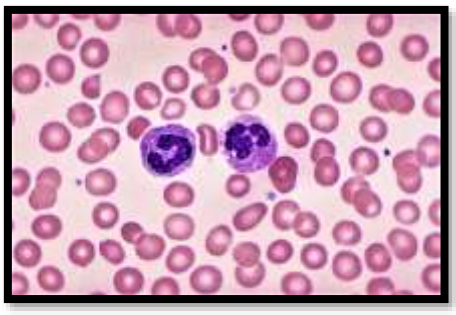

FIG 4.33: Image of Blood smear before contrast stretching

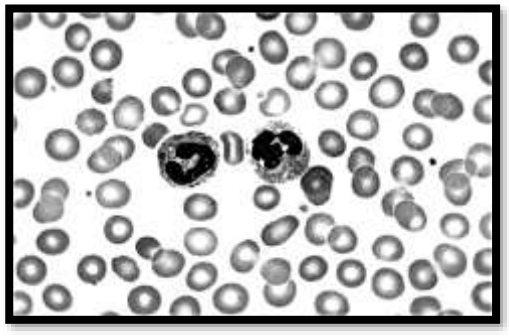

FIG 4.34: Image obtained after contrast stretching

4.7.2.6 Edge Detection

Edge detection aims at identifying points in an image at which the brightness of the image changes sharply. The points at which brightness changes sharply are organized into set of curved line segments termed as edges. The purpose is to capture important events and changes in properties.

Applying an edge detector algorithm to an image significantly reduce the amount of data to be processed and therefore filter the information that are considered as less relevant. Edges extracted from non-trivial images are hampered by fragmentation, that is, edge curves are not connected. Edge detection is one of the fundamental steps in image processing.

CANNY EDGE DETECTOR

The Canny edge detector is an operator which uses multi stage algorithm to detect a wide range of edges in images. It was developed and named after John F Canny in 1986.

The general criteria for edge detection are:

- Edge detection with low error rate that is the detection should accurately note as many edges shown in the image.
- The edge point detected should accurately localize in the center of the edge.
- A given edge in the image should be marked once.
- Noises in image should not create false edges.

FIG 4.35: Image of Blood smear before edge detection (canny edge detector)

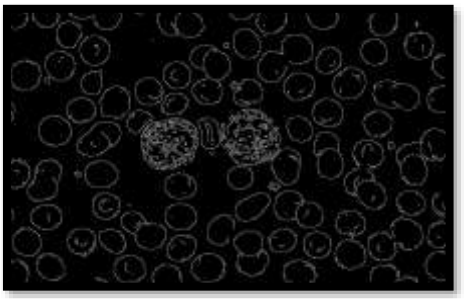

FIG 4.36: Image obtained after edge detection (canny edge detector)

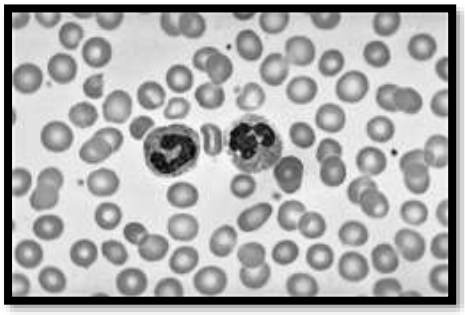

FIG 4.37: Image of Blood smear after applying median filter

FIG 4.38: Image obtained for median filter applied images after edge detection

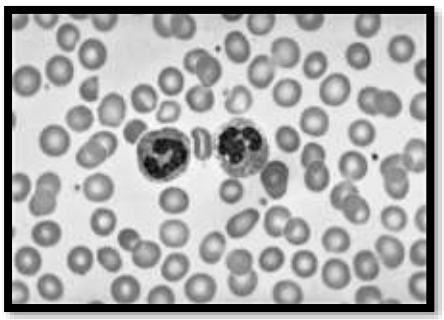

FIG 4.39: Image of Blood smear after applying Gaussian filter

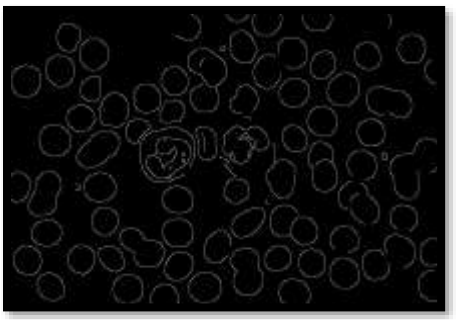

FIG 4.40: Image obtained for Gaussian filter applied images after edge detection

4.7.3 SEGMENTATION

Segmentation is the process of partitioning an image into a multiple pixel. The main aim of the segmentation is to simplify the presentation of an image to analyze easily. It is the process of labelling to every pixel in an image. In this project, we have differentiated the blood cells into RBC, WBC and Platelets from the uploaded images. The easy way of image segmentation is by thresholding. There are many methods of thresholding and we have used Otsu's thresholding in this project. In this method, image uploaded is converted from gray scale to a binary image. The blood cells are segmented based on their morphological factors like shape and size.

TABLE 4.1: Morphology of Blood cells

Morphology	RBC (Erythrocytes)	WBC (Leucocytes)	Platelets (Thrombocytes)
Shape	Bi-concave	Irregular	Irregular
Size	7-8 µm in diameter	12-17 µm in diameter	2-3 µm in diameter

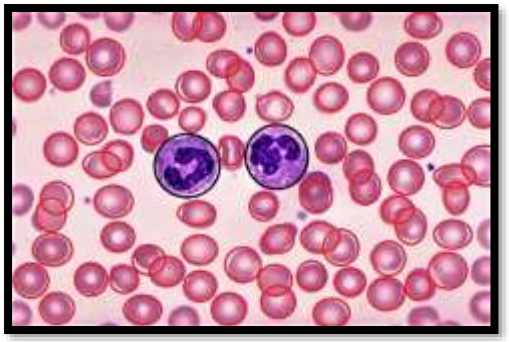

FIG 4.41: Segmented RBC & WBC in female blood smear.

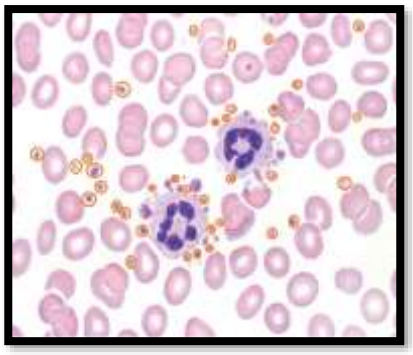

FIG 4.42: Segmented Platelets in female blood smear.

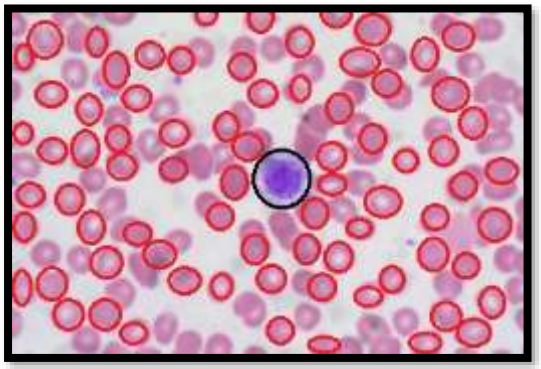

FIG 4.43: Segmented RBC & WBC in male blood smear.

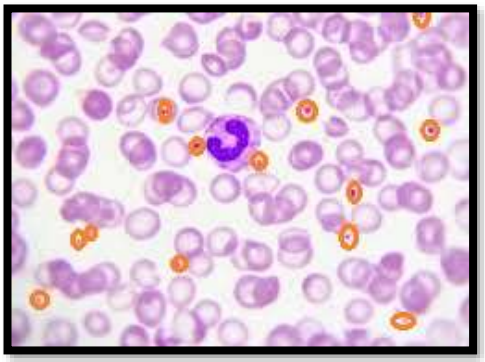

FIG 4.44: Segmented Platelets in male blood smear.

4.7.4 COUNTING OF THE CELLS

The last process of the image processing is the counting of blood cells. In this, we can calculate the average size and count of the erythrocytes, leucocytes and thrombocytes. From the segmented image, the leucocyte and platelets can be counted easily whereas erythrocytes cannot be counted due to their overlapping. In this case we have used special algorithm for counting the overlapped erythrocytes.

4.7.5 HOUGH TRANSFORM FOR OVERLAPPED RBCs

This is the special algorithm which is used for counting of overlapped RBCs. The Hough transform is a special technique which is used to isolate features of a particular shape within an image. The classical Hough transform is commonly used for the detection of regular curves like lines, circles, ellipses, etc. In this project, we have used Hough transform for circular parameters. The classical Hough transform retains many applications as most of the manufactured parts contains featured boundaries which is described by the regular curves.

The main advantage of Hough transform is that it is tolerant of gaps in featured boundary and is unaffected by image noise. This can be used to identify parameters of a curve which fits a set of given edge points. This edge description is commonly obtained from detecting operator such as canny edge detector. It may contain multiple edge fragments corresponding to a single feature.

Further, as the output of an edge detector defines only where features are present in an image. The work of the Hough transform is to determine both what the features are and how many of them exist in the image.

CHAPTER 5

RESULT AND DISCUSSION

5.1 RESULT

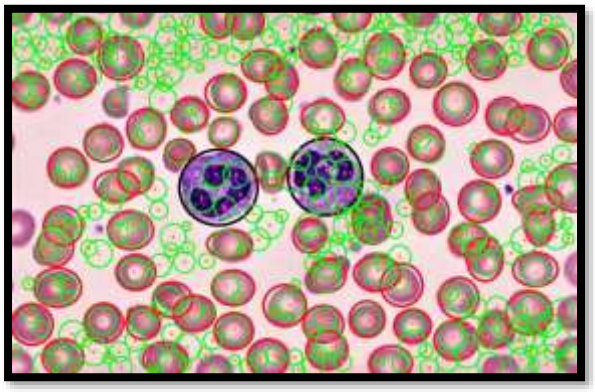

FIG 5.1: Output obtained after processing for Female RBC and WBC

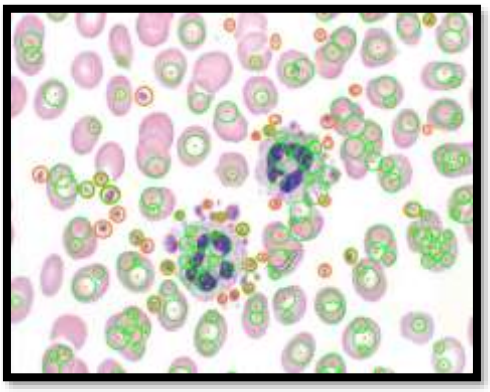

FIG 5.2: Output obtained after processing for Female Platelets

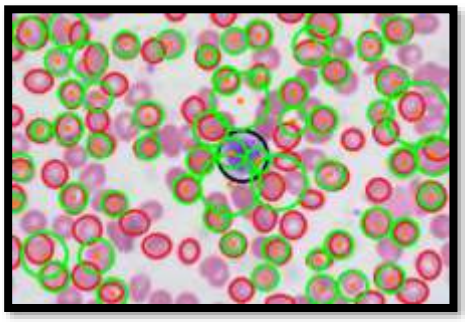

FIG 5.3: Output obtained after processing for male RBC and WBC

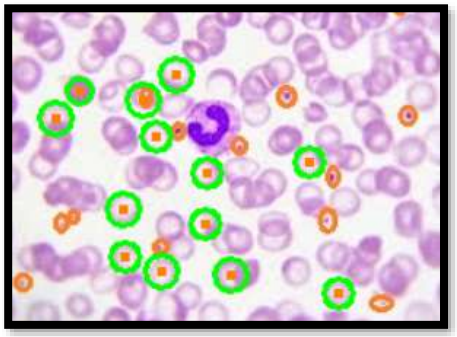

FIG 5.4: Output obtained after processing for male Platelets

TABLE 5.1: Normal range of blood cells

Gender	RBC (Erythrocytes)	WBC (Leucocytes)	Platelets (Thrombocytes)
Male	4.7 to 6.1 million cells per microlitre (cells/mcL)	5000 – 10,000 cells/mcL	1,35,000 – 3,17,000 cells/mcL
Female	4.2 to 5.4 million cells/mcL	4500 – 11000 cells/mcL	1,57,000 – 3,71,000 cells/mcL

TABLE 5.2: Blood cell count obtained after processing

Patient	Gender	RBC (million cells per microlitre cells/mcL)	WBC (cells/mcL)	Platelets (cells/mcL)	Disease
1	Male	6.00 (N)	943 (D)	1,95,000 (N)	Viral Infection
2	Female	3.18 (D)	5000 (N)	2,40,000 (N)	Anemia
3	Male	5.20 (N)	6700 (N)	2,00,000 (N)	-
4	Female	4.52 (N)	4600 (N)	50,000 (D)	Heavy Menstrual bleeding
5	Male	4.40 (N)	5300 (N)	3,00,000 (N)	-
6	Female	4.20 (N)	7200 (N)	3,60,000 (N)	-
7	Male	5.80 (N)	8550 (N)	2,30,500 (N)	-
8	Female	5.00 (N)	9600 (N)	3,80,000 (I)	Iron Deficiency
9	Male	4.75 (N)	6840 (N)	1,55,000 (N)	-
10	Female	5.16 (N)	9100 (N)	2,50,000 (N)	-

N – Normal I – Increased D - Decreased

5.2 DISCUSSION

We can conclude that erythrocytes, leucocytes and thrombocytes are counted successfully by using python Idle software. From the table, we can see that there are some individuals who have normal blood cell count whereas some individuals have increase or decrease in blood count. In a male individual, we can see that the leucocyte count is decreased. This is because the individual is exposed to viral infection. When we take the case of female, there are 3 female individuals whose RBC count is decreased because the patient is anemic. For the other female individual, platelet count has been decreased because she has heavy menstrual flow. Finally, for the other female individual platelet count has been increased because of iron deficiency.

5.3 FORMULA

5.3.1 RBC

Volume = 1/50 mm

Dilution employed = 100 (Diluting Fluid)/0.5 (Blood) = 200

Number of cells = cells counted*volume*dilution = cells counted*50*200 = cells counted*10000

5.3.2 WBC

RBC per 100 WBCs

Multiply WBC count by 100 = Cells counted*100 ------ 1

Add 100 to RBC count = RBC count+100 ------ 2

Divide both the result = ---1/---2

5.3.3 Platelets

Multiply the cells counted with 15,000

5.4 DISEASES

5.4.1 RBC Count Increases

- Polycythemia vera – type of Blood Cancer. causes your bone marrow to make too many red blood cells
- Kidney tumors – Kidney Cancer
- Lung disease – Emphysema (shortness of breath), COPD - Chronic Obstructive Pulmonary Disease (block airflow and make difficult in breathing), Pulmonary Fibrosis (lung becomes scarred and damaged).

5.4.2 RBC count decreases

- Anemia - lack enough healthy red blood cells to carry adequate oxygen to your body's tissues.
- Bone Marrow Failure - body cannot supply itself with the blood it needs
- Bone Marrow Failure Diseases - Aplastic anemia (prone to infections and uncontrolled bleeding), MDS - MyeloDysplastic Syndrome (preleukemia) and PNH - Paroxysmal Nocturnal Haemoglobinuria (red blood cells to break apart).

5.4.3 WBC count increases

- Leukemia – Blood Cancer
- Stress
- Infection
- Inflammation
- Trauma (deeply distressing)
- Allergy

5.4.4 WBC count decreases

- Leukopenia
- Auto immune disorders
- Viral infections

5.4.5 Platelets count increases

- Acute bleeding and blood loss
- Cancer
- Iron deficiency
- Removal of your spleen

- Hemolytic anemia – type of anemia in which your body destroys red blood cells faster than it produces them.
- Inflammatory disorders – Rheumatoid Arthritis (autoimmune disease that can cause joint pain and damage throughout your body), Inflammatory bowel disease (group of intestinal disorders that cause prolonged inflammation of the digestive tract.) or Sarcoidosis (disease involving abnormal collections of inflammatory cells that form lumps known as granulomata. This disease usually begins in the lungs, skin, or lymph nodes).

5.4.6 Platelets count decreases

- Viral infections – Chicken Pox, hepatitis C, HIV
- Sepsis, a severe bacterial infection in blood
- Heavy menstrual bleeding
- Nose bleeding

5.5 APPLICATION AND FUTURE SCOPE

The main area where this project would be applicable is in the medical field for research in change of cell structure during vaccination and when different infections affect the human body. This can also be used for identifying different types of leucocytes. It can also be used for comparing the counts with the manual counts so that we will have a better understanding of the cell structure. It can also be incorporated for counting animal blood cells as well as the cell characteristic changes for different type of animals like arthropods and molluscs as a future scope.

CHAPTER 6

SUMMARY AND CONCLUSION

6.1 SUMMARY

The blood cell count is a blood test which is used to evaluate our health and detect a range of disorders including anemia, viral infections and leukemia. Blood cell count measures several components and features in blood which included – Red Blood Cells or the erythrocytes which carry oxygen, White blood cells or the leucocytes which fight infection and the platelets or thrombocytes which help with blood clotting. Abnormal increase or decrease in cell count will lead to medical condition that calls for further evaluation.

6.2 CONCLUSION

As a conclusion, this project is successfully completed using image processing technique for estimation of RBC, WBC and Platelets. This project proposes a system which uses python Idle software using OpenCV for the blood cell count. We can see that different algorithm are used for detection and counting of the cells. From this we can also note that, the overlapped RBCs are counted by using Hough Transform. It enables the study of morphological features like the shape and size of the RBC, WBC and Platelets. The pathologist can also determine whether the person has normal range of blood count. There is a need for fast and cost-effective production of the reports. This system includes an effective method when compared to manual blood cells counting or by using analyzers and also in producing the blood count reports as soon as possible. This also helps us to prevent death of an individual due to lack of treatment or due to late treatment. This system can be further improvised if possible, for detecting diseases related to blood cell morphologies.

REFRENCES

1) Akshaya P Sahastrabuddhe, “Counting of WBC and RBC using Image Processing: A Review” published in International Journal of Research in Engineering and Technology (IJRET) (2016) eISSN: 2319-1163 | pISSN: 2321-7308

2) Amruta Pandit, Shrikrishna Kolhar, Pragati Patil, “Survey on Automatic RBC Detection and Counting”, published in International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering (IJAREEIE) (2015) Vol. 04 (Issue: 01).

3) Dibyendu Ghoshal, Soumen Biswas, “Blood Cell Detection using Thresholding Estimation Based Watershed Transformation with Sobel Filter in Frequency Domain” published in 12th International Multi-Conference on Information Processing-2016 (IMCIP) (2016).

4) Guda sai manisha, Rohith R Bharadwaj, Nayani Lathasree, Chinnareddyvari Nikhil, Rashmi K, “Complete Blood Checkup using Image Processing” published in International Journal of Advanced Science and Technology, (2020) Vol. 29 (Issue 05).

5) Gulpreet Kaur Chadha, Aakarsh Srivastava, Abhilasha Singh, Ritu Gupta, Deepanshi Singla, “An Automated Method for Counting Red Blood Cells using Image Processing” published in International Conference on Computational Intelligence and Data Science (ICCIDS) (2019).

6) Lokhande T P, Salunke P B, Shinde P T, Chaugule J D, Prof. Bhong V S, “Counting of RBC’s and WBC’s using Image Processing Technique” published in International Journal of Engineering Science and Computing (IJESC) (2019) Vol. 09 (Issue 04).

7) Navid Shaikh, Karansing Rajaput, Rahul Pawar, Dr. A S Vibhute, “Counting of RBCs and WBCs using Image Processing Technique” published in International Research Journal of Engineering and Technology (IRJET), (2019) Vol. 06 (Issue 04).

8) Nicoleta Safca, Dan Popescu, Loretta Ichim, Hassan Elkhatib, Oana Chenaru, “Image Processing Techniques to Identify Red Blood Cells” published in 22nd International Conference on System Theory, Control and Computing (ICSTCC) (2019) pp. 93-98.

9) Padma Mali, Vineeta P Gejji, Vishwesh K Aithal, “A survey for automatic identification and counting of blood cells” published in International Research Journal of Engineering and Technology (IRJET) (2020) Vol. 07 (Issue 05).

10) Pawan Agrawal, Pradipti Verma, “Automated detection and counting of red blood cell using image processing techniques” published in International Research of Scientific Research and Management (IJSRM) (2015) Vol. 03 (Issue 04).

11) P Najiya Nasreen, A Chempak Kumar, P Asjad Nabeel, “Counting of RBCs using circular hough transform with median filtering” published in the third National conference on Emerging Trends in engineering (NET) (2015).

12) Pradipta Maji, Ankita Mandal, Madhura Ganguly, Sanjoy Saha, “An Automated Method for Counting and Characterizing Red Blood Cells using Mathematical Morphology” published in Eighth International Conference on Advances in Pattern Recognition (ICAPR) (2015).

13) Razali Tomari, Wan Nurshzwani Wan Zakaria, Rafidah Ngadengon, Mohd Helmy Abd Wahab, “Red blood cell analysis by considering an overlapping constraint” published in Asian Research Publishing Network (ARPN) journal of engineering and applied sciences (2015) Vol.10 (Issue 03).

14) Shiroq Al-Megren, Heba Kurdi Fatimah Al-Hafiz, "Red blood cell segmentation by thresholding and Canny detector" published in the 8th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH) (2018).

15) Shweta Kamble, Prof. Dr. Manasi Dixit, “A Review on Blood Cells counting using Image Processing” published in International Journal of Creative Research Thoughts (IJCRT) (2020) Vol. 08 (Issue 02).

16) S S Savkare, S P Narote, “Blood Cell Segmentation from Microscopic Blood Images” published in International conference on Information Processing (ICIP) (2015) pp. 502-505.

17) Thejashwini M, M C Padma, “Counting of RBC’s and WBC’s Using Image Processing Technique” published in International Journal on Recent and Innovation Trends in Computing and Communication (IJRITCC) (2015) Vol. 03 (Issue 05).

18) Valiente, Leonardo C Castor, Celine Margaret T Mendoza, Arvin Jay B, Song Cherry Jane L, Torres Bailey Brian P, Dela Cruz, Jennifer C, “Determination of blood components (WBCs, RBCs and platelets) Count in Microscopic Images using Image Processing and Analysis” published in IEEE 9th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM) (2017).

19) Varghese J. Thiruvinal, Sangeetha Prasanna Ram, “Automated Blood Cell Counting and Classification Using Image Processing” published in International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering (IJAREEIE) (2017) Vol. 06 (Issue 01).

20) Varun D Dyanesh, P S Lakshmi, K Reddy, Abirami S Vasavi, “Blood Cell Count using Digital Image Processing” published in International Conference on Current Trends towards Converging Technologies (ICCTCT) (2018).

APPENDIX

```
import cv2
import numpy as np

# read original image
image = cv2.imread(“female rbc and wbc count.jpg")
# convert to gray scale image
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
cv2.imwrite('gray.png', gray)
# apply median filter for smoothening
blurM = cv2.medianBlur(gray, 5)
cv2.imwrite('blurM.png', blurM)
# apply gaussian filter for smoothening
blurG = cv2.GaussianBlur(gray, (9, 9), 0)
cv2.imwrite('blurG.png', blurG)
# histogram equalization
histoNorm = cv2.equalizeHist(gray)
cv2.imwrite('histoNorm.png', histoNorm)
# create a CLAHE object for
# Contrast Limited Adaptive Histogram Equalization (CLAHE)
clahe = cv2.createCLAHE(clipLimit=2.0, tileGridSize=(8, 8))
claheNorm = clahe.apply(gray)
cv2.imwrite('claheNorm.png', claheNorm)

# contrast stretching
# Function to map each intensity level to output intensity level.
def pixelVal(pix, t, s, R, S):
  if 0 <= pix <= t:
      return (s / t) * pix
   elif t < pix <= R:
      return ((S - s) / (R - t)) * (pix - t) + s
   else:
return ((255 - S) / (255 - R)) * (pix - R) + S
```

```
# Define parameters
r1 = 70
s1 = 0
r2 = 200
s2 = 255
# Vectorize the function to apply it to each value in the Numpy array.
pixelVal_vec = np.vectorize(pixelVal)
# Apply contrast stretching
contrast_stretched = pixelVal_vec(gray, r1, s1, r2, s2)
contrast_stretched_blurM = pixelVal_vec(blurM, r1, s1, r2, s2)
cv2.imwrite('contrast_stretch.png', contrast_stretched)
cv2.imwrite('contrast_stretch_blurM.png', contrast_stretched_blurM)

# edge detection using canny edge detector
edge = cv2.Canny(gray, 100, 200)
cv2.imwrite('edge.png', edge)
edgeG = cv2.Canny(blurG, 100, 200)
cv2.imwrite('edgeG.png', edgeG)
edgeM = cv2.Canny(blurM, 100, 200)
cv2.imwrite('edgeM.png', edgeM)
src_path = "D:/BME/Project/female rbc and wbc count.jpg “

# read enhanced image
img = cv2.imread('female rbc and wbc count', 0)
# morphological operations
kernel = np.ones((5, 5), np.uint8)
dilation = cv2.dilate(img, kernel, iterations=1)
closing = cv2.morphologyEx(img, cv2.MORPH_CLOSE, kernel)
# Adaptive thresholding on mean and gaussian filter
th2 = cv2.adaptiveThreshold(img, 255, cv2.ADAPTIVE_THRESH_MEAN_C,
cv2.THRESH_BINARY, 11, 2)
th3 = cv2.adaptiveThreshold(img, 255, cv2.ADAPTIVE_THRESH_GAUSSIAN_C,
cv2.THRESH_BINARY, 11, 2)
```

```
# Otsu's thresholding
ret4, th4 = cv2.threshold(img, 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)
# Initialize the list
Cell_count, x_count, y_count = [], [], []

# read original image, to display the circle and center detection
display = cv2.imread("D:/BME/Project/female rbc and wbc count.jpg")

# hough transform with modified circular parameters
circles = cv2.HoughCircles(img, cv2.HOUGH_GRADIENT, 1.2, 20,
                           param1=50, param2=30, minRadius=0,   maxRadius=25)
# circle detection and labeling using hough transformation
if circles is not None:
    # convert the (x, y) coordinates and radius of the circles to    integers
    circles = np.round(circles[0, :]).astype("int")
    # loop over the (x, y) coordinates and radius of the circles
    for (x, y, r) in circles:
        cv2.circle(display, (x, y), r, (0, 255, 0), 2)
        cv2.rectangle(display, (x - 2, y - 2),
                      (x + 2, y + 2), (0, 128, 255), -1)
        Cell_count.append(r)
        x_count.append(x)
        y_count.append(y)
    # show the output image
    cv2.imshow(“count", display)
    cv2.waitKey(0)
# display the count of cells
print(len(Cell_count))
# Total number of cells
print(Cell_count)
# X co-ordinate of circle
print(x_count)
# Y co-ordinate of circle
print(y_count)
```

OUTPUT

318

[24, 24, 24, 24, 24, 24, 24, 24, 24, 24, 24, 24, 24, 24, 23, 24, 24, 24, 24, 24, 24, 24, 24, 23, 23, 22, 24, 24, 24, 24, 24, 24, 24, 24, 24, 24, 24, 24, 23, 24, 24, 24, 23, 24, 24, 24, 24, 24, 24, 24, 24, 22, 20, 24, 23, 23, 24, 23, 22, 24, 24, 22, 22, 24, 24, 22, 22, 22, 19, 24, 23, 22, 24, 24, 24, 22, 20, 20, 24, 24, 24, 23, 22, 24, 24, 23, 23, 22, 20, 24, 23, 23, 20, 23, 22, 22, 22, 19, 18, 24, 23, 20, 20, 19, 19, 24, 22, 20, 19, 19, 24, 22, 19, 22, 20, 20, 19, 19, 15, 23, 20, 20, 18, 18, 16, 15, 22, 18, 18, 16, 20, 19, 19, 18, 15, 24, 16, 20, 16, 19, 19, 19, 19, 16, 16, 15, 19, 15, 15, 14, 16, 16, 14, 18, 15, 14, 19, 16, 15, 15, 14, 21, 17, 16, 16, 16, 15, 14, 14, 14, 23, 18, 16, 16, 16, 16, 15, 14, 14, 14, 18, 15, 14, 12, 15, 15, 14, 14, 13, 18, 16, 15, 15, 15, 14, 14, 14, 14, 20, 16, 16, 14, 14, 12, 12, 12, 15, 15, 15, 11, 16, 15, 15, 15, 14, 14, 14, 12, 12, 14, 14, 14, 14, 12, 12, 11, 16, 15, 15, 14, 14, 12, 12, 12, 12, 12, 10, 15, 14, 14, 12, 12, 12, 11, 10, 10, 14, 12, 12, 11, 10, 9, 15, 14, 14, 12, 12, 12, 11, 11, 10, 10, 10, 15, 14, 14, 12, 12, 11, 11, 11, 10, 9, 12, 11, 10, 10, 10, 10, 10, 12, 10, 10, 8, 12, 12, 11, 11, 11, 11, 10, 10, 10, 10, 10, 10, 10, 10, 8, 8, 8, 14, 11, 11, 10, 10, 8, 8, 8, 8, 7, 11, 10, 10, 10, 10, 10, 10, 8]

[673, 469, 955, 202, 620, 728, 272, 76, 28, 967, 675, 551, 755, 805, 46, 460, 832, 809, 218, 393, 704, 796, 992, 305, 499, 741, 43, 262, 403, 571, 746, 862, 946, 185, 809, 979, 991, 741, 254, 500, 608, 58, 109, 164, 166, 365, 601, 760, 818, 952, 93, 400, 837, 245, 487, 659, 889, 494, 301, 880, 272, 385, 101, 549, 215, 223, 305, 646, 803, 763, 867, 349, 410, 650, 958, 554, 667, 562, 668, 896, 38, 470, 581, 763, 293, 614, 788, 781, 470, 235, 406, 880, 121, 257, 559, 871, 323, 22, 589, 503, 231, 386, 332, 787, 367, 705, 934, 439, 855, 374, 239, 943, 464, 86, 157, 83, 316, 466, 329, 847, 850, 385, 758, 358, 789, 997, 201, 971, 340, 731, 560, 584, 701, 945, 688, 284, 137, 797, 43, 281, 890, 937, 553, 578, 911, 59, 285, 587, 314, 200, 718, 602, 422, 839, 89, 922, 925, 197, 955, 589, 757, 549, 705, 956, 961, 242, 134, 286, 674, 295, 989, 147, 538, 635, 977, 151, 347, 659, 683, 605, 885, 998, 183, 371, 961, 190, 484, 22, 833, 176, 188, 721, 739, 343, 361, 862, 476, 241, 539, 549, 106, 1009, 73, 610, 950, 410, 671, 740, 68, 941, 20, 931, 998, 183, 440, 649, 71, 484, 19, 818, 908, 197, 417, 913, 932, 524, 362, 869, 574, 621, 89, 196, 231, 362, 608, 33, 904, 652, 716, 290, 788, 259, 478, 64, 569, 976, 905, 523, 665, 866, 813, 814, 20, 583, 634, 928, 887, 482, 650, 218, 697, 256, 867, 625, 665, 131, 725, 280, 1003, 118, 248, 453, 561, 385, 98, 338, 688, 149, 551, 59, 530, 225, 497, 388, 11, 439, 853, 435, 795, 302, 932, 241, 435, 472, 592, 724, 998, 824, 842, 861, 781, 665, 289, 385, 518, 382, 383, 105, 317, 778, 242, 80, 826, 165, 339, 469, 548, 253]

[641, 293, 475, 97, 128, 575, 127, 505, 29, 91, 290, 74, 118, 55, 112, 641, 350, 597, 478, 154, 496, 29, 337, 50, 545, 320, 152, 367, 583, 487, 244, 278, 400, 325, 517, 25, 73, 340, 231, 525, 368, 74, 296, 250, 565, 302, 397, 370, 421, 380, 628, 505, 442, 644, 145, 478, 587, 33, 266, 388, 632, 229, 507, 333, 415, 179, 25, 572, 170, 89, 28, 347, 27, 383, 45, 580, 109, 207, 363, 34, 556, 394, 28, 608, 416, 55, 104, 69, 314, 398, 305, 193, 97, 38, 463, 93, 77, 86, 362, 403, 314, 363, 545, 593, 556, 75, 476, 110, 464, 626, 104, 544, 213, 419, 436, 395, 191, 58, 427, 649, 364, 431, 639, 76, 517, 437, 608, 621, 200, 421, 602, 274, 641, 23, 445, 525, 38, 86, 593, 469, 645, 319, 649, 632, 205, 634, 59, 427, 95, 38, 368, 28, 622, 109, 82, 425, 28, 655, 607, 604, 465, 273, 31, 649, 309, 567, 436, 93, 171, 337, 268, 67, 469, 320, 651, 637, 562, 409, 76, 219, 431, 567, 634, 57, 579, 175, 377, 155, 567, 584, 440, 623, 136, 28, 650, 575, 627, 535, 49, 20, 131, 305, 449, 94, 346, 437, 59, 388, 112, 226, 573, 269, 628, 88, 33, 549, 43, 77, 403, 188, 629, 393, 92, 571, 609, 310, 605, 643, 83, 410, 307, 71, 614, 38, 429, 62, 386, 236, 327, 447, 189, 340, 548, 424, 567, 250, 99, 287, 19, 346, 15, 307, 507, 203, 358, 290, 59, 646, 93, 16, 173, 141, 523, 14, 458, 135, 487, 33, 605, 153, 68, 93, 301, 17, 59, 62, 341, 581, 625, 568, 646, 212, 53, 127, 644, 197, 83, 374, 536, 640, 202, 11, 406, 189, 465, 21, 405, 44, 326, 597, 286, 323, 391, 572, 21, 83, 275, 393, 397, 41, 292, 602, 626, 656, 278, 361, 411, 211]

Process finished with exit code 0