Classification of Parasitized Cells for Malaria Detection with Help of Deep Learning

Wasem Qassab Bashi

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Science
in
Electrical and Electronic Engineering

Eastern Mediterranean University
July 2023
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Ali Hakan Ulusoy
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science in Electrical and Electronic Engineering.

Assoc. Prof. Dr. Rasime Uyguroğlu
Chair, Department of Electrical and Electronic Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Electrical and Electronic Engineering.

Prof. Dr. Erhan İnce
Supervisor

Examining Committee

1. Prof. Dr. Erhan İnce

2. Prof. Dr. Önsen Toygar

3. Assoc. Prof. Dr. Kamil Yurtkan

ABSTRACT

Millions of people worldwide suffer from malaria, a potentially fatal disease. Early and precise diagnosis is essential for the medical condition to be successfully treated and managed. This thesis employs three computer-aided methods to determine percentages of red blood cells that are either parasitic or uninfected given test set(s) obtained from the National Institutes of Health (NIH) dataset. The three methods employed are traditional image processing, Support Vector Machine (SVM), and Convolutional Neural Networks based Deep Learning (CNN-DL). The simulations are performed using a dataset that has 27,558 images of red blood cells. The traditional image processing method achieves an accuracy of 91.97%. SVM classifier using Histogram of Oriented Gradients (HOG) features has an accuracy of 88.6% and with features extracted using Local Binary Patterns (LBP) accuracy has improved to 92.5% using a smaller subset of 6,040 images. The two previous methods proved to be inferior when compared with the CNN- DL classification which gave an accuracy of 95.7% using AlexNet, and 96.32% using GoogLeNet. The accuracy of each of the three computer-aided methods was based on performance metrics calculated using confusion matrices and the Receiver Operating Characteristic (ROC) curves.

Keywords: Support Vector Machine; Local Binary Patterns; Histogram of Oriented Gradients; Convolutional Neural Networks Based Deep Learning.

ÖZ

Dünya çapında milyonlarca insan, potansiyel olarak ölümcül bir hastalık olan sıtmadan muzdariptir. Tıbbi durumun başarılı bir şekilde tedavi edilmesi ve yönetilmesi için erken ve kesin teşhis şarttır. Bu tezde, Ulusal Sağlık Enstitüleri (USE) veri setinden rastgele seçilerek oluşturulmuş deneme kümeleri içindeki kırmızı kan hücrelerinin parazitik olma veya enfekte olmama yüzdelerini belirlemek adına üç bilgisayar destekli yöntem kıyaslanmaktadır. Kullanılan yöntemler sırası ile geleneksel görüntü işleme yöntemleri, destek vektör makinesi (DVM) ve evrişimsel sinir ağları tabanlı derin öğrenme (ESA-DÖ) olmuştur. Benzetimler için 27,558 kırmızı kan hücresi içeren bir veri seti kullanılmıştır. Geleneksel görüntü işleme yöntemi %91,97'lik bir doğruluk ile temiz ve parazitli hücreleri ayırabilmiştir. Yönlendirilmiş Gradyanlar Histogramı (YGH) özniteliklerini kullanan DVM sınıflandırıcının doğruluğu %88.6 iken yerel ikili Örüntü (YİÖ) kullanılarak çıkarılan özniteliklerle sınıflandırma sonrası doğruluk %92,5'e yükselmiştir. AlexNet ve GoogleNet derin öğrenme modelleri kullanıldığında elde edilen sınıflandırma sonuçları ilk iki bilgisayar tabanlı yöneme göre daha iyi çıkmıştır. Bütün veri tabanı kullanıldığında sınıflandırma başarısı AlexNet için %95.7 iken GoogLeNet %96.32 ile biraz daha iyi sonuç vermiştir. Başarı değerleri her üç yöntem için elde edilen karışıklık matris değerleri kullanılarak ve Alıcı Çalışma Karakteristik eğrilerinden faydalanılarak değerlendirilmiştir.

Anahtar Kelimeler: Destek Vektör Makinesi; Yerel İkili Örüntü; Yönlendirilmiş Gradyanlar Histogramı; Evrişimsel Sinir Ağları Tabanlı Derin Öğrenme.

Dedicated to

My parents and siblings

ACKNOWLEDGMENT

First and foremost, I would like to thank my supervisor Prof. Dr. Erhan A. İnce for guiding and helping me through my Master's Thesis, for his patience, and for kindly sharing his knowledge with me.

I would like to express my gratitude to my friends who supported and motivated me during this journey.

I also wish to thank all the faculty members at the Department of Electrical and Electronics Engineering, and especially the head of the department, Assoc. Prof. Dr. Rasime Uyguroğlu.

Last but not least, I would like to express my appreciation to my parents and my siblings for all the constant love and support throughout my life so I can get to this point.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

$f(x)$	Radial Basis Function
σ	Free Variable
$K(x, y)$	Kernel Function
AUC	Area Under The Curve
CNN	Convolutional Neural Networks
DL	Deep Learning
FP	False Positive
FN	False Negative
HOG	Histogram of Oriented Gradients
LBP	Local Binary Patterns
NIH	National Institute of Health
NLM	National Library of Medicine
ROI	Region of Interest
ROC	Receiver Operating Characteristic
RBF	Radial Basis Function
ReLU	Rectified Linear Unit
SVM	Support Vector Machine
TP	True Positive
TF	True Negative
VG	Vanishing Gradients

Chapter 1

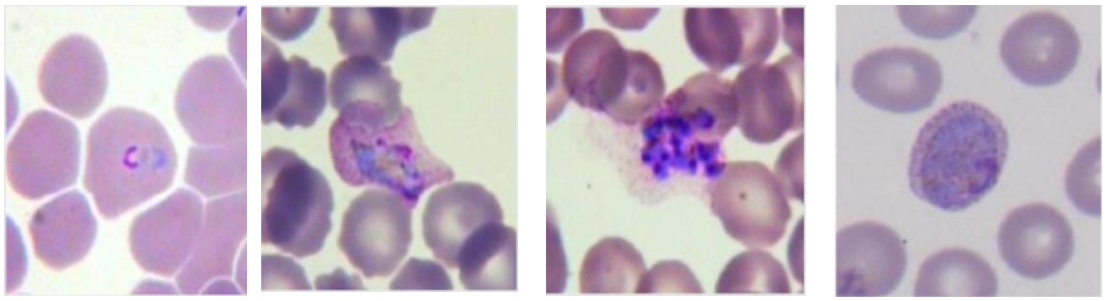
INTRODUCTION

In this chapter the five different species of malaria are introduced as well as general information about the illness and how risky it is. This chapter present some of the work and results obtained by other researchers regarding the usage of computer-based methods to classify the malaria cells. The research conducted in this thesis has highlighted that computer-based techniques such as Support Vector Machines (SVM) and Convolutional Neural Networks in Deep Learning (CNN-DL) have demonstrated substantially higher accuracy and time efficiency compared to the conventional manual microscopy-based diagnosis.

1.1 Overview

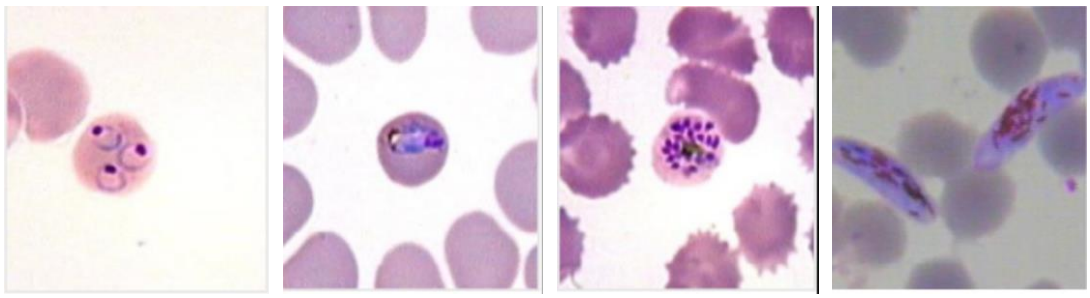
Plasmodium is a class of parasitic protozoan that causes the illness referred to as malaria. The red blood cells are infected by this parasite, which is spread by the bites of infected female Anopheles mosquitoes. Malaria is a primary cause of juvenile neuro-disabilities and is particularly harmful to children in Africa, where one dies from it virtually every minute. According to the World Malaria Report 2016 [1], 3.2 billion people in 95 nations and territories are at risk of contracting malaria with 1.2 billion of those people living in sub-Saharan Africa. There are five species of malaria, P. falciparum, P. malariae, P. vivax, P. ovale, and P. knowlesi. The dataset obtained from NIH has images of both P. vivax and P. falciparum so the focus will be on these two types. Outside of the sub-Saharan region P. vivax is the most common type of malaria, one of the reasons for that is the capability to create dormant forms called hypnozoites

in the liver. These hypnozoites can stay inactive for extended periods, ranging from several months to years, before becoming active again and triggering a fresh episode of malaria, so it appears to be the specie of malaria to be seen by doctors more often than the other species. It can cause symptoms like fever, headache, shaking chills, and vomiting. P. falciparum however, is considered to be the deadliest due to how rapidly in reproduces which means it multiply incredibly fast. It also has the ability to change the proteins on its surface, which allows the parasite to evade the immune response and continue to cause infection. The symptoms are similar to P. vivax but more intense that in most cases it is fatal. There are different stages for malaria, it starts with ring and the last stage in the cell is called gametocyte. Figures 1.1 and 1.2 show P. vivax and P. falciparum and their stages respectively. Every year, skilled microscopists manually count parasites and infected red blood cells in digitized copies of hundreds of millions of blood slides to diagnose malaria. However, microscopy-based diagnostics are not standardized and heavily rely on the microscopist's expertise [1]. In low-resource settings, microscopists often work in isolation without proper training or a system in place to improve their skills, leading to incorrect diagnostic decisions in the field [1]. False-positive instances can result in the unnecessary use of anti-malarial medications and their possible adverse effects, whereas false-negative cases can result in the needless use of antibiotics, a second consultation, missed workdays, and the development of severe malaria.



(a) Ring (b) Trophozoite (c) Schizont (d) Gametocyte

Figure 1.1: P. vivax and Its Four Stages



(a) Ring (b) Trophozoite (c) Schizont (d) Gametocyte

Figure 1.2: P. falciparum and Its Four Stages

The work done in this thesis has pointed out that the computer based methods like SVM and Convolutional Neural Networks based Deep Learning (CNN-DL) models proved to be significantly more accurate and more time efficient then the manual microscopy-based diagnosis. CNN-DL architectures such as GoogLeNet and AlexNet would provide classification accuracies up to 96% for the full set and 98% for the smaller subset of images (6,040 images).

1.2 Literature Review

Malaria is an illness that affects the red blood cells in the human body. The female mosquitoes carrying the parasite bite humans and transmit the disease. Malaria has a high risk of being deadly. It is regarded as a major health issue all around the world because of how fatal it could be. There are 5 different species of malaria, two of which could be dangerous. In [2], many different computer based methods were used to

classify malaria, among those are, Cubic SVM, Linear SVM, and Cosine KNN. However, Cubic SVM got the highest accuracy out of those methods.

Pre-trained deep learning models however proved to deliver higher classification performances. According to Shah D. et al [3], a custom convolutional neural network is primarily designed to distinguish between healthy and infected blood samples. The proposed model consists of three convolutional layers and fully connected layers. The neural network presented is a cascade of several convolutional layers having multiple filters present in layers, which yields fairly good accuracy as per the available resources [3]. The custom CNN classifier that uses a sigmoid activation function managed to reach an accuracy of 95%. In [4], a method that uses a combination of VGG19-SVM was proposed. The proposed transfer learning approach can be achieved by unifying existing Visual Geometry Group (VGG) network and Support Vector Machine (SVM). Implementation of this unification is carried out by using "Train top layers and freeze out rest of the layers" strategy [4]. Initial '*k*' layers of the pre-trained VGG are retained and (*n*-*k*) layers are replaced with SVM [4]. This method gave an accuracy of 93.1%. Reddy, A. S. B. et al in [5], classified the malaria image set from NIH using DL model ResNet-50. The images will enter ResNet50 layer with the pre-trained weights and the last layer is a classic fully connected dense layer with sigmoid activation. The proposed model consists of 2 layers, Pre- trained ResNet layer, and a dense layer. Pre-Trained Weights for the ResNet50 model are imported. The input data will be trained with the pre-trained weights and the only layer which is learning with back propagation is the dense layer [5]. This gave an accuracy of 95.91%. According to Diker in [6], GoogLeNet had an accuracy of 96.31%, AlexNet 95.77% and ShuffleNet 96.44% when using a dataset which is same as the one used in this thesis.

In [7], AlexNet, VGG-16, ResNet-50, and a customized machine learning model were used to classify the images. The custom CNN has three convolutional layers and two fully connected layers. The input to the model constitutes segmented cells of 100× 100×3 pixel resolution. The convolutional layers use 3×3 filters with 2 pixel strides. The first and second convolutional layers have 32 filters and the third convolutional layer has 64 filters. The sandwich design of convolutional/rectified linear units (ReLU) and proper weight initialization enhances the learning process. Max-pooling layers with a pooling window of 2×2 and 2 pixel strides follow the convolutional layers for summarizing the outputs of neighbouring neuronal groups in the feature maps. The pooled output of the third convolutional layer is fed to the first fully-connected layer that has 64 neurons, and the second fully connected layer feeds into the Softmax layer [7]. Dropout regularization with a dropout ratio of 0.5 is applied to outputs of the first fully connected layer. These models gave accuracies between 93% - 95.7% where ResNet-50 led to the highest accuracy. To further improve classification performance features were extracted from optimal layers [7], and an accuracy of 95.9% was attained using ResNet-50.

1.3 Outline

In this work, we will employ three computer-aided methods to determine what percent of the tested red blood cells from the National Institutes of Health (NIH) [8] dataset are parasitic or uninfected. Traditional image processing, Support Vector Machine (SVM) based classification [9-10], and classification based on deep learning architectures known as AlexNet [11], and GoogLeNet, are the three methods studied. Our goal is to determine the most efficient computer-based approach for replacing and/or assisting the decisions of microscopists.

Traditional image processing is a method of analysing visual data which takes a select interest in certain areas within images instead of merely observing the entire visual environment at one time. This is achieved by dividing up said images according to different criteria such as texture attributions or particular colour characteristics. Once these individualized segments are observed that meet this set criterion, further specific digital processes can be undertaken for each section individually. This selective analysis provides a higher degree of accuracy when it comes to manipulating content making the technique particularly useful in fields like medical imaging, natural resource management, digital arts, and object detection. In this report, the area and region of interest will be the features used.

SVM is a machine learning algorithm that helps classify data into different groups. It finds the best decision boundary represented by a line that separates the data points most effectively. This boundary is determined by the support vectors, which are the closest points from each class. If the data points are not easily separable, SVM can transform them into a higher-dimensional space where they become more distinguishable. SVM can then classify new data points by determining which side of the boundary they fall on. It is a powerful algorithm that generalizes well to unseen data, making accurate predictions based on previous examples. The extraction of certain features like Local Binary Patterns (LBP) and Histogram of Oriented Gradients (HOG) and that will be done in this work.

AlexNet is an 8-layer deep learning convolutional neural network. Five of these layers are dedicated to extracting important features from input images through the application of filters or kernels, enabling the detection of patterns, edges, corners, and textures. These filters slide across the image and generate feature maps. The remaining

three layers are Max Pooling Layers, responsible for reducing the spatial dimensions of the feature maps while preserving essential information. Max pooling involves selecting the maximum value within each pooling region. AlexNet was developed by Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton and gained significant recognition and widespread usage following its participation in the 2012 ImageNet competition. This network has undergone pre-training using an extensive dataset containing over a million images [11].

In 2014, Google researchers introduced GoogLeNet in the research article "Going Deeper with Convolutions" with the assistance of several universities. When it came to the 2014 ILSVRC image classification competition, this architecture was the winner. In comparison to previous winners AlexNet (ILSVRC 2012 Winner) and ZF-Net (ILSVRC 2013 Winner), as well as VGG (ILSVRC 2014 Runner Up), it has offered a considerable decrease in error rate. This architecture employs methods like global average pooling and 1–1 convolutions in the middle of the architecture [12]. There are 22 layers in the architecture as a whole. The architecture was created with consideration for computational effectiveness. The rationale behind the architecture's ability to function on individual devices even with limited computational resources [12].

1.4 Thesis Contributions

In this thesis three different computer aided methods were simulated to determine the most appropriate one that could replace the manual microscopy-based diagnostics of malaria. As a result of the work done, a conference paper was submitted to SIU2023 and the paper was accepted for publication. A copy of the four page paper could be found in the Appendix.

1.5 Thesis Organisation

Chapter 2 introduces the image processing steps utilised by the traditional image processing approach and provides an example for classifying test images either as parasitic or uninfected. Chapter 3 discusses what a Support Vector Machine (SVM) is and also explains the two-feature extraction methods used with it: namely LBP and HOG. Chapter 4 provides architectural details for each deep learning model. Simulation results are presented and compared in Chapter 5 for full dataset and for a smaller version of the full dataset (subset). Finally, conclusions are made and ideas for future work are provided in Chapter 6.

Chapter 2

TRADITIONAL IMAGE PROCESSING

Chapter 2 explains traditional image processing and shows how edge detection and thresholding can be used to detect the parasite(s) in the red blood cells and how the method classifies these tested cells. The chapter also describes how the filter known as Sobel filter could be used as an edge detector to detect the edges of the parasitic part(s) in a cell.

2.1 Introduction

Image Processing is the study and technology of analysing, manipulating, and enhancing digital images. It includes various techniques and algorithms for extracting important and significant information from images, enhancing their quality, performing image restoration, enhancement, classification, detection, and more. This thesis focuses more on the classification and detection parts of red blood cells that may be infected by the malaria parasite.

2.2 Methodology of Traditional Image Processing

Traditional image processing uses a gradient-based edge detector and morphological operations on the cells images. The data set is from the National Library of Medicine (NLM) [8] and contains 27,558 images separated into two classes: parasitic and uninfected. Each class has a total of 13,779 red blood cell images. As a pre-processing step images in the dataset are converted into grayscale. The region of interest (ROI) in each image is determined by edge detection. A second edge detection is then employed in the ROI using a Sobel filter with a threshold of 0.05 to obtain both the perimeter of

the ROI and the edges of the infection (if it exists). The difference between the two edge images would then be dilated to create the mask that shows the position of the parasites in the analysed cell. In the final step, the area of the mask is compared with an empirical threshold (100 herein). If the area is larger than the set threshold, the cell is classified and marked as infected, otherwise it is considered to be uninfected. In Figure 2.1 the block diagram for the process is shown. Figure 2.2 depicts the images obtained after each step.

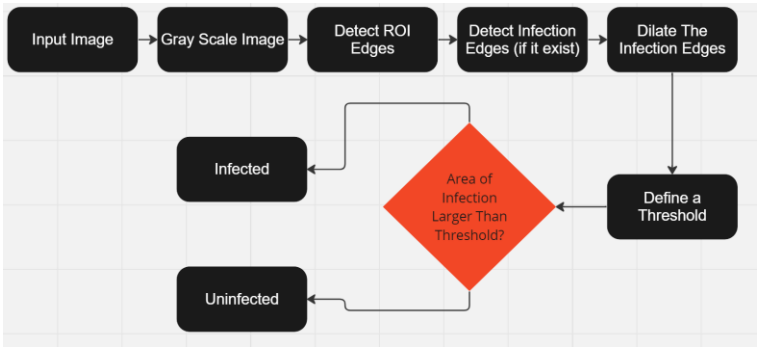


Figure 2.1: Methodology for Traditional Image Processing

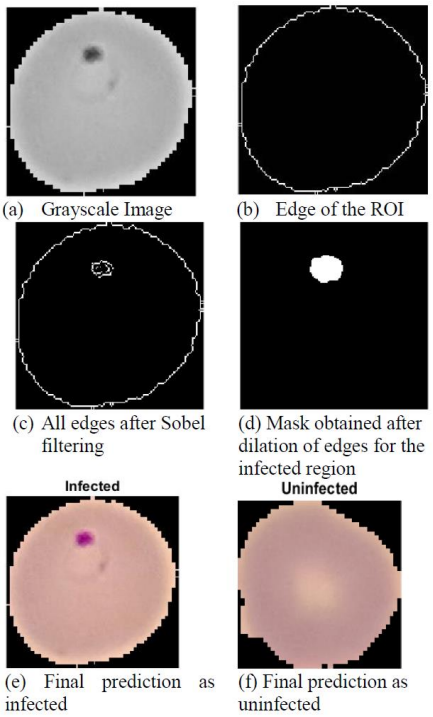


Figure 2.2: Classification of Red Blood Cells Using Traditional Image Processing

2.2.1 Sobel Filter

The Sobel filter, also known as the Sobel operator, is commonly used in for edge detection in images. Named after its inventor, Irwin Sobel. The filter is based on calculating the gradient of the image, highlighting areas of prompt intensity change and identifying edges. Sobel filter is a simple and efficient method that calculates the gradient of an image with two separate filters: one for vertical edge detection and the other for horizontal edge detection. These filters are usually a 3×3 matrix. The vertical edge filter, commonly known as the Sobel-Y, highlights the vertical variations in intensity, while the horizontal edge filter, known as the Sobel-X, highlights the horizontal variations [13-16].

The filter matrices of the Sobel filter can be summarised as follows:

$$\text{Sobel-X}=\begin{bmatrix}-1 & 0 & 1\\ -2 & 0 & 2\\ -1 & 0 & 1\end{bmatrix}$$

$$\text{Sobel-Y}=\begin{bmatrix}-1 & -2 & -1\\ 0 & 0 & 0\\ 1 & 2 & 1\end{bmatrix}$$

To use a Sobel filter, each filter matrix convolved with the corresponding pixels in the image. The filters calculate the magnitude and direction of the gradient at each pixel. The magnitude of the gradient represents the strength of the edge, while the gradient direction indicates the direction of the edge. Once the Sobel filter is applied, the gradients can be further processed, such as thresholding to obtain binary edge maps or combining horizontal and vertical gradients to calculate the overall magnitude and direction of the gradients.

Chapter 3

SUPPORT VECTOR MACHINE

In this chapter Support Vector Machine (SVM) is introduced as a classifier. It will be used in combination with two feature extraction methods: Local Binary Patterns (LBP) and Histogram of Oriented Gradients (HOG). For data points that are not linearly separable, a kernel function known as Radial Basis Function (RBF) was used to project the data to higher dimensions to separate the points and then to classify the images. A subset of 6,040 images was used in the SVM classifier combined with both feature extraction techniques.

3.1 Introduction

A Support Vector Machine (SVM) is a machine learning algorithm that is widely used for classification tasks. This algorithm works by analysing labelled data to recognize patterns and relationships between features. The aim is to find the optimal decision boundary that separates different classes in the feature space. In binary classification tasks, the SVM identifies a hyper-plane that divides the two classes, and it chooses the hyper-plane that maximizes the distance between the hyper-plane and the closest points from each class, known as support vectors. SVMs are a popular choice for classification tasks because they are effective at handling high-dimensional data and can handle both linearly and non-linearly separable data. They are also dubious to overfitting when compared with other machine learning algorithms [9-10].

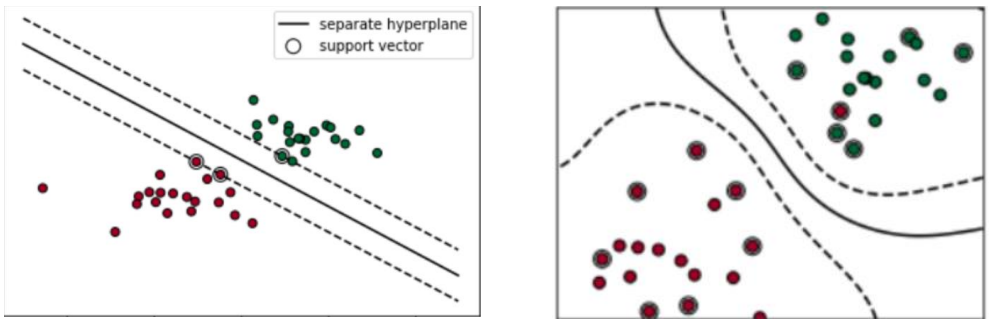
3.2 Methodology of SVM

In this thesis, SVM with two feature extraction methods were used, the first is Local Binary Patterns (LBP), and the second is the Histogram of Oriented Gradients (HOG). The images in the dataset will be separated into a split of 80%-20% where 80% is for the training images and 20% is for the testing. A kernel function Radial Basis Function Kernel (RBF) is used as well, this will allow for more accurate results since the data is not linear, unlike the standard SVM that is linear. The radial basis function can be written as follows:

$$K(x, y) = \exp\left(-\frac{||x-y||^2}{2\sigma^2}\right) \tag{3.1}$$

where $||x - y||$ represents the Euclidean distance between x and y, and σ is a free variable to tune the equation.

Figure 3.1 (a) and (b) show the difference between the linear SVM and the RBF kernel function:



(a) SVM Distribution Plot (b) RBF Distribution Plot

Figure 3.1: The Distribution Plots for the Linear SVM and the Kernel Function RBF[29]

A block diagram of the SVM methodology is shown in Figure 3.2 below:

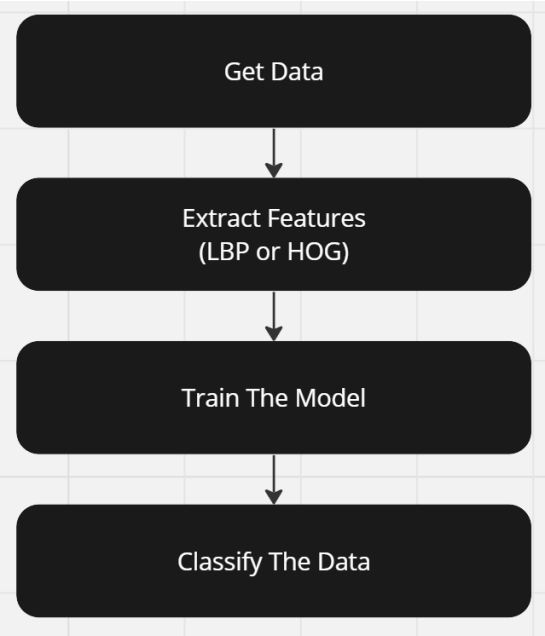


Figure 3.2: SVM Classifier Methodology

3.2.1 Local Binary Patterns

LBP is a technique that helps in identifying patterns in images by analysing the neighbouring pixels and comparing them to the center pixel [17]. The local binary pattern operator is an operator that describes the surrounding of a pixel by generating a bit-code from the binary derivatives of a pixel. In its simplest form the LBP operator takes the 3×3 surrounding of a pixel and generates a binary 1 if the neighbour of the center pixel has larger value than the center pixel and 0 when neighbour is less than the center pixel. The eight neighbours of the center can then be represented with an 8-bit number (an unsigned 80bit integer). It is used to describe the texture of grayscale images and can efficiently capture spatial patterns and contrast in the images. The function 'extractLBPFeatures' under MATLAB returns extracted uniform binary patterns from a grayscale image. LBP sample bins represent the particular order of adjoining pixels in Local Binary Patterns (LBP). Each bin corresponds to a specific

binary pattern generated by comparing the gray level of the current pixel with that of its surrounding pixels [14]. The bins can capture the possible shapes in an image. The number of LBP bins is determined by the radius chosen and the number of neighbours used in the calculations. The combination of radius and neighbours helps project the results in different patterns. Figure 3.3 depicts the placement of neighbouring pixels around a center pixel at different radii. For 8 neighbours ($P=8$) there would be 256 different combinations of two histogram features, resulting in 256 LBP sample bins [18-19]. In summary, the LBP sample bin represents the different shapes that arise in an image based on the comparison of gray levels between the central pixel and its surrounding neighbours.

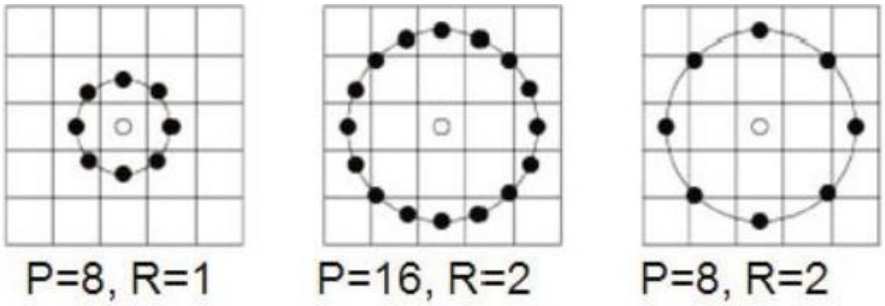


Figure 3.3: LBP with Different Radii and Neighbouring Pixel Values

3.2.2 Histogram of Oriented Gradients

HOG is a feature descriptor used in computer vision for object recognition and detection tasks. It was introduced in 2005 by Dalal and Triggs et.al. It has been shown that HOG is effective in a wide range of applications, including pedestrian detection [20], face detection [19], and object recognition [14]. It has also been used as a feature descriptor in machine learning algorithms, such as Support Vector Machines [21]. One of the strengths of HOG is its ability to capture the shape and structure of an object, regardless of its colour or texture. However, HOG is computationally intensive and requires careful parameter tuning to achieve optimal performance. There have been

many extensions and variations of HOG proposed over the years, such as multi-scale HOG [22] and dense HOG [23], which aim to address some of these limitations.

In the context of Histogram of Oriented Gradients (HOG) features, the terms “Magnitude” and “HOG Bins” are used. Magnitude refers to the intensity or strength of the gradient at each pixel, indicating the variation in pixel intensity between adjacent pixels in different directions. The magnitude value indicates the intensity of the local edge or texture information. Higher values correspond to more pronounced edges or textures, while lower values indicate smoother areas or less pronounced edges.

A HOG Bin refers to histogram bins used to quantify gradient orientations. The Histogram of Oriented Gradient (HOG) method divides gradient orientations into multiple bins to capture the distribution of gradient orientations in a specific image or region of interest. Binning allows summarizing the distribution of gradient orientations at a concise representation that holds information that is essential.

The gradient at each pixel refers to the spatial rate of change of intensity or colour values in an image. It measures how rapidly the pixel values change in different directions.

Figure 3.4 shows an example of how the orientation of the gradient in the image and the magnitude of each orientation or bin are generally represented:

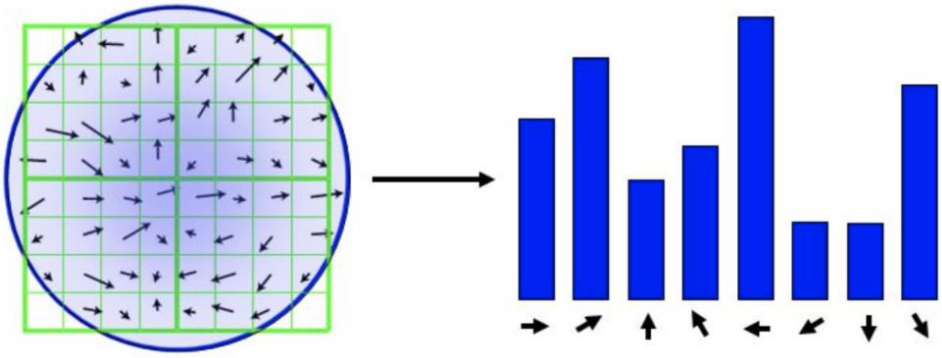


Figure 3.4: General Representation of HOG Magnitudes and Bins

The arrows below each bar in the histogram indicate the bins in the image, where each blue bar represents the strength of the gradient.

Chapter 4

DEEP LEARNING USING TRANSFER LEARNING

Chapter 4 introduces two machine learning models that rely on convolutional neural networks (CNN). These models that make use of many CNN layers are referred to as the deep learning convolutional neural networks (DL-CNNs). The first model which is 8 layers deep is known as AlexNet and the slightly more complex 22 layers deep second model is referred to as the GoogLeNet. The chapter explains the structures for both models in details in addition to the ReLU activation function that is used in both models to introduce non-linearity and allow the models to capture more complex features.

4.1 Introduction

A Convolutional Neural Network (CNN) is a deep learning algorithm primarily employed for image recognition and computer vision tasks. Drawing inspiration from the human brain's visual processing, CNNs are designed to automatically learn and extract meaningful features from images. Comprising multiple layers, including convolutional, pooling, and fully connected layers, CNNs employ filters (kernels) in the convolutional layers to detect patterns and features by sliding over the input image. Additionally, pooling layers reduce spatial dimensions, allowing the network to focus on crucial information. CNNs have brought about a revolution in the field of computer vision, exhibiting impressive accuracy in tasks such as object detection, image classification, and facial recognition. Their hierarchical architecture and ability to comprehend intricate patterns have made CNNs widely applicable in real-world

scenarios, including autonomous vehicles, medical image analysis, and natural language processing. AlexNet and GoogLeNet are the two CNN models used.

AlexNet is a convolutional neural network based on a deep learning structure with 8 layers. Five of these layers are convolutional neural networks and the other three are Max Pooling Layers. AlexNet was born out as a result of the ImageNet competition in 2012. It is a network that is pre-trained using more than a million images [11]. A prerequisite to using AlexNet or any of the other pre-trained networks is to have a deep learning toolbox installed as part of the MATLAB platform. As for GoogLeNet, a typical GoogLeNet architecture consists of 22 layers. The architecture was developed with computing productivity in mind. The explanation for why the design can run on any device even with limited computing capability. Similar to AlexNet it was also the winner in ILSVRC, however, this was two years later in 2014 [12].

4.2 AlexNet Methodology

The reasons for choosing AlexNet as the deep learning tool were the following:

1- It does not suffer from vanishing gradients (VG) problem,

2- The ReLU activation function used by AlexNet makes the network 6 times faster.

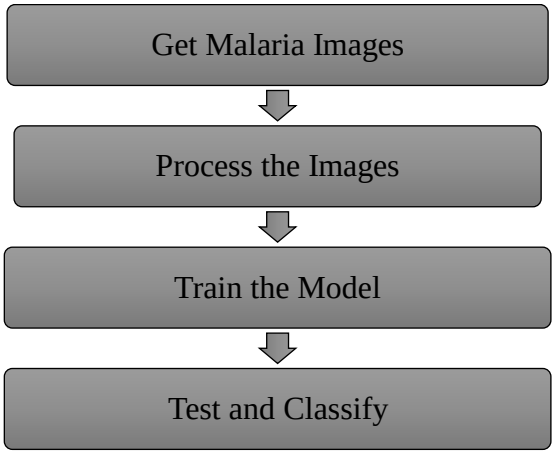


Figure 4.1: Methodology for Deep Learning Models

4.2.1 ReLU Activation Function

The ReLU (Rectified Linear Unit) function is a nonlinear activation function widely used in neural networks and deep learning models. It introduces nonlinearity into the network, allowing the model to capture and represent complex relationships in the data. The ReLU implementation is described as follows:

$$f(x) = \max(0, x) \tag{4.1}$$

In this equation the input to the function is denoted by x, and the function returns the maximum value between 0 and x. If input x is positive or zero, the ReLU function returns the same value. However, if x is negative, the ReLU function yields 0. The main advantage of ReLU is its computational efficiency, which makes it suitable for training larger artificial neural networks. This function is also straightforward and avoids the problem of missing slope that can occur with other functional functions, such as sigmoid or tanh functions by applying the ReLU function to the outputs of the hidden layer of the neural network, the model is able to identify complex and nonlinear patterns in the data ReLU helps the network to identify and provide relevant features works while preventing unnecessary information or noise. It is important to note that ReLU can generate dead neurons, where the neurons are turned off for negative outputs, and cannot support the learning process To address this, various types of ReLU are introduced, such as Leaky ReLU and Parametric ReLU they are allowed to do so. Overall, ReLU processing applications have played an important role in the success of deep learning models, enabling complex representation learning and state-of-the-art performance in a variety of industries, with computer vision, natural language processing, and language including acceptance.

4.3 Structure of AlexNet

As mentioned before AlexNet is a convolutional neural network based on a deep learning structure with 8 layers. Five of these layers are convolutional neural networks and the other three are Max Pooling Layers. Table 3.1 shows the structure in detail.

Table 4.1: AlexNet Structure

Layers	#Neurons	Filter Size	Strides	Padding	Size Feature Map	Activation Function
Conv 1	96	11×11	4	-	55×55×96	ReLU
Max Pool 1	-	3×3	2	-	27×27×96	-
Conv 2	256	5×5	1	2	27×27×256	ReLU
Max Pool 2	-	3×3	2	-	13×13×256	-
Conv 3	384	3×3	1	1	13×13×384	ReLU
Conv 4	384	3×3	1	1	13×13×384	ReLU
Conv 5	256	3×3	1	1	13×13×265	ReLU
Max Pool 3	-	3×3	2	-	6×6×256	-
Dropout 1	p=0.5	-	-	-	6×6×256	-

Figure 4.2 depicts a block diagram representation of AlexNet:

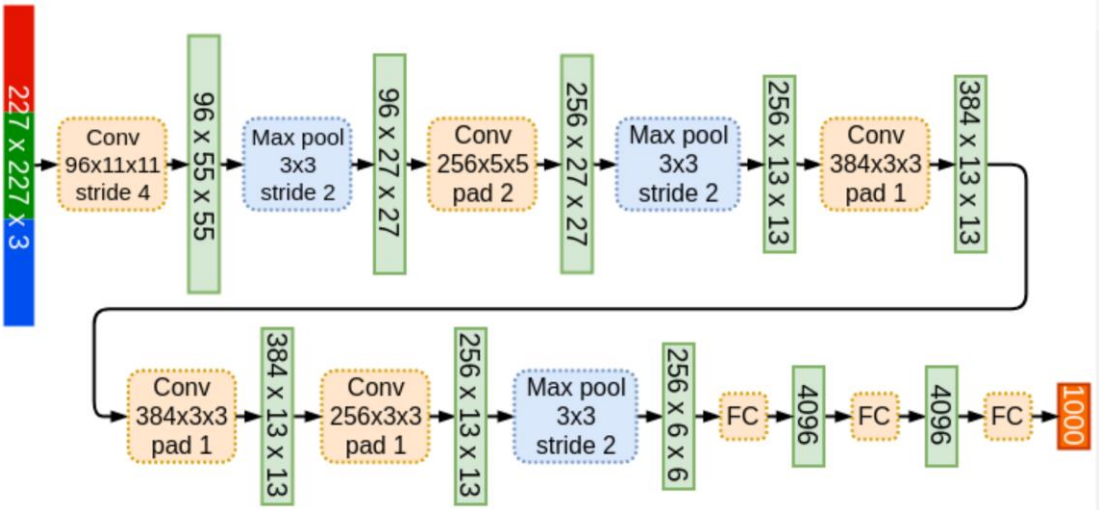


Figure 4.2: AlexNet Block diagram

The input image has the size feature map of 227×227×3. In the AlexNet system, the number of nodes refers to the number of statistical units or individual nodes in each segment of the neural network [24]. These neurons perform calculations on the input data and produce results that are then passed to the next level. The number of neurons in AlexNet varies from place to place. The number of neurons in the convolutional layers corresponds to the number of filters or feature maps used in each layer. Each neuron in the convolutional layer learns to search for specific patterns or features in the input data. By increasing the number of neurons, the network is able to learn more complex and non-absurd representations of the input [24]. The number of nodes in a fully connected layer refers to the number of nodes in each layer connected to all the nodes in the previous layer. These layers capture high-level positions by learning complex combinations that have been omitted by previous layers. With respect to "strides", the size or magnitude of the steps taken by the filter or kernel over the input data during a convolutional operation is defined. In the convolutional layer, a small filter is applied to the input data by moving a slide over the data using specific steps [25]. The stride value determines the spatial difference between successive operations of the filter. A stride of 1 implies that the filter moves one pixel at a time, resulting in a dense overlap of the filter's receptive field [25]. This dense overlap can help capture fine-grained details but increases the computational complexity. On the other hand, a stride greater than 1 allows for a larger step size, reducing the overlap and computational burden. However, using a larger stride may lead to a loss of some spatial information. In AlexNet, strides are utilized in convolutional layers to control the down sampling or pooling of feature maps, reducing the spatial dimensions and extracting more abstract features [24]. The specific stride values employed in each layer of

AlexNet are chosen to strike a balance between preserving spatial information and managing the computational complexity of the network.

4.4 GoogLeNet Methodology

GoogLeNet architecture takes RGB picture of size 224×224 [12]. Rectified Linear Units (ReLU) serve as the activation for each and every one of the convolutions in GoogLeNet. functions. Figure 4.3 shows the building blocks of GoogLeNet:

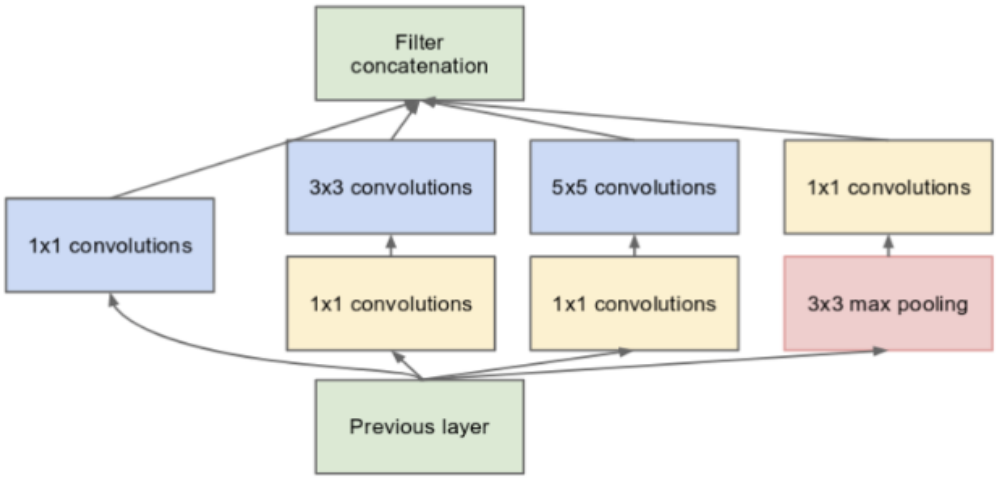


Figure 4.3: GoogLeNet Building Blocks

4.4.1 1×1 Convolution Layers

1×1 convolutions are utilized in the inception architecture's architecture. These convolutions reduced the weights and biases of the architecture. By diminishing the limits we furthermore increase the significance of the designing.

4.4.2 Global Average Pooling

The completely associated layers are utilized toward the organization's end in the previous architecture. These fully connected layers house the majority of architecture's parameters, which increases computation costs [12]. The GoogLeNet architecture makes use of a method known as global average pooling at the network end. This layer

averages a 7×7-feature map to a 1×1-feature map. Top-1 accuracy is also improved by 0.6% and the number of trainable parameters is reduced to zero as a result [12].

4.4.3 Inception Module

The inception module is different from AlexNet, in this architecture, The convolution size is constant for every layer 1×1, 3×3, 5×5 convolution, and 3×3 max pooling are all performed simultaneously at the input in the Inception module, and the stacked final output is the result [12]. This is centered on the idea that objects at multiple scales will be handled more efficiently by convolution filters of varying sizes.

4.5 Structure of GoogLeNet

GoogLeNet architecture consists of convolution layers, max pool, inception, and average pooling layers. Full details such as patch size, stride, depth and filter sizes used are provided in Figure 4.4 and Figure 4.5 provides a block diagram representation of the DL model GoogLeNet:

type	patch size/ stride	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj	params	ops
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								
inception (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
inception (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
inception (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
inception (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
inception (4e)		14×14×832	2	256	160	320	32	128	128	840K	170M
max pool	3×3/2	7×7×832	0								
inception (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
inception (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
avg pool	7×7/1	1×1×1024	0								
dropout (40%)		1×1×1024	0								
linear		1×1×1000	1							1000K	1M
softmax		1×1×1000	0								

Figure 4.4: GoogLeNet Structure

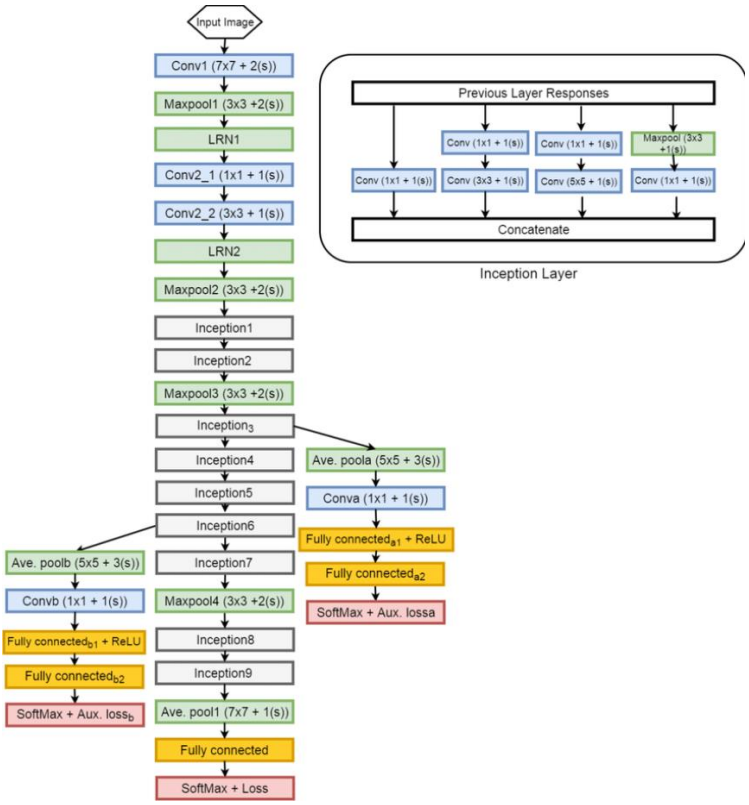


Figure 4.5: GoogLeNet Block Diagram [30]

Chapter 5

SIMULATION RESULTS

In this chapter three computer-based methods have been simulated to work out classification performances of each algorithm when applied to red blood cells extracted randomly from the NIH dataset. These methods include the traditional image processing, the Support Vector Machine (SVM) based classification and the Convolutional Neural Networks based Deep Learning (CNN-DL) models such as AlexNet and GoogLeNet. SVM was simulated with a subset of the NIH dataset which contained a total of 6,040 images due to the slow training time. All simulations have been carried out on a computer with Intel(R) Core(TM) i5-1135G7 CPU and with 8.00GB of RAM.

5.1 Simulation Parameters

For SVM based classifier and the two DL-CNN models the simulation parameters used are depicted in Tables 5.1- 5.3. For the traditional image processing method the full data set was used and data was not split into training/testing sets. Table 5.1 shows the parameters used for the SVM classifier and Tables 5.2, and 5.3 shows the parameters for CCN-DL models AlexNet and GoogLeNet respectively.

Table 5.1: SVM Simulation Parameters

Parameter	**Setting**
Total Number of Images	6,040
% of Images Trained	80%
Cell Size	16×16
Input Image Size	160×160
Kernel Function	RBF

Table 5.2: AlexNet Simulation Parameters

Parameters	Setting
Total Number of Images	27,558 & 6,040
% of Images Trained	80%
Input Image size	227×227
Learning Rate	1e-4

Table 5.3: GoogLeNet Simulation Parameters

Parameters	Setting(s)
Total Number of Images	27,558 & 6,040
% of Images Trained	80%
Input Image size	224×224
Learning Rate	1e-4

5.2 Performance Metrics

In the three computer-aided methods compared herein the effectiveness of the binary classifiers was assessed based on their confusion matrices and also using performance metrics such as Precision, Sensitivity, Accuracy and F1-score, as well as showing the ROC curve to represent how well each architecture performs.

The percentages given in a confusion matrix are based on the test set and in this study was 20% of the full set of 27,558 images. In our case this would mean 5,512 test images. The four percentages given at the top left corner of a confusion matrix represent the true predictions [true positive (TP) and true negative (TN)] and the false predictions [false positive (FP) and false negative (FN)]. As an example if out of 5,512 test images 2,658 were predicted as TPs (refer to Figure 5.6) then its corresponding percentage would be 48.2. Percentages for true and false predictions are all calculated out of 5,512.

The performance metrics used in this study are:

$$Precision = \frac{TP}{TP + FP} \tag{5.1}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{5.2}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{5.3}$$

$$F1 - score = 2 \times \frac{(Precision \times Sensitivity)}{(Precision + Sensitivity)} \tag{5.4}$$

where *TP* denotes true positive, *TN* is the true negative, *FP* denotes the false positive, and *FN* is the false negative.

Accuracy denotes the ratio of the correctly labelled subjects to the whole pool of subjects. Precision is the actual correct prediction divided by total prediction and sensitivity equals the number of true positives divided by the total number of true positives and false negatives. Finally, *F1*-score is the harmonic mean (average) of the precision and recall. For the traditional image processing, the classification is done without splitting the database but for the SVM and CNN-DL the data set had to be split to obtain training and testing sets. In this work, 80 % of the images were used for training and the remaining 20% were used for testing purposes.

5.3 *k*-fold Cross Validation Method

k-fold cross validation is when the training set is split into *k* subsets randomly, one subset will be used to validate and the rest for training. Once the training is finished another subset will be used for validation. After each fold an accuracy of validation will result. At the end of all the folds, an average validation of all the accuracies will be calculated and shown as the final validation accuracy. The advantage of using this method is getting a better performance estimate since the same images are used for

training and validation at separate times, this will also help in preventing overfitting. In this work we have chosen to use a 5-fold cross validation.

5.4 Evaluation of Performance Metrics

A smaller subset was obtained from the dataset obtained from NIH, since the SVM can not perform as well for large sets and the training time is way longer to the point where it is not very feasible to use the full set while using SVM, unlike the CNN-DL where we do not face this issue. The subset contains only 6,040 images taken randomly from the original 27,558 images. The results are shown in details using confusion matrices and summarized in Table 5.4 below:

Table 5.4: Performance Metrics Achieved

Method	TP	TN	FP	FN	Accuracy	Precision	Sensitivity	F1-Score
AlexNet (Fullset)	2,607	2,670	86	149	95.74%	96.81%	94.59%	95.69%
GoogLeNet (Fullset)	2,658	2,651	105	98	96.32%	96.20%	96.44%	96.32%
AlexNet (Subset)	596	588	16	8	98.01%	97.39%	98.68%	98.03%
GoogLeNet (Subset)	578	601	3	26	97.76%	98.00%	97.52%	97.76%
HOG SVM	531	539	65	73	88.58%	89.09%	87.91%	88.50%
LBP SVM	546	572	32	58	92.55%	94.46%	90.40%	92.39%
Image Processing	12,800	12,544	979	1,235	91.97%	92.89%	91.20%	92.04%

Figures 5.1 -5.11 show the confusion matrices for all the methods used as well as the ROCs for the deep learning methods:

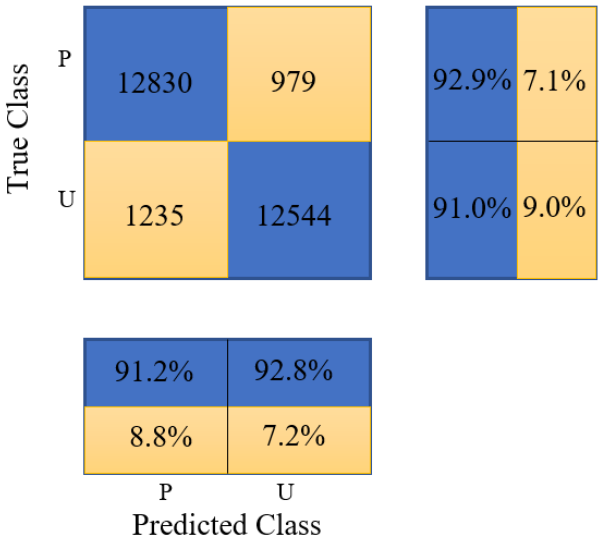
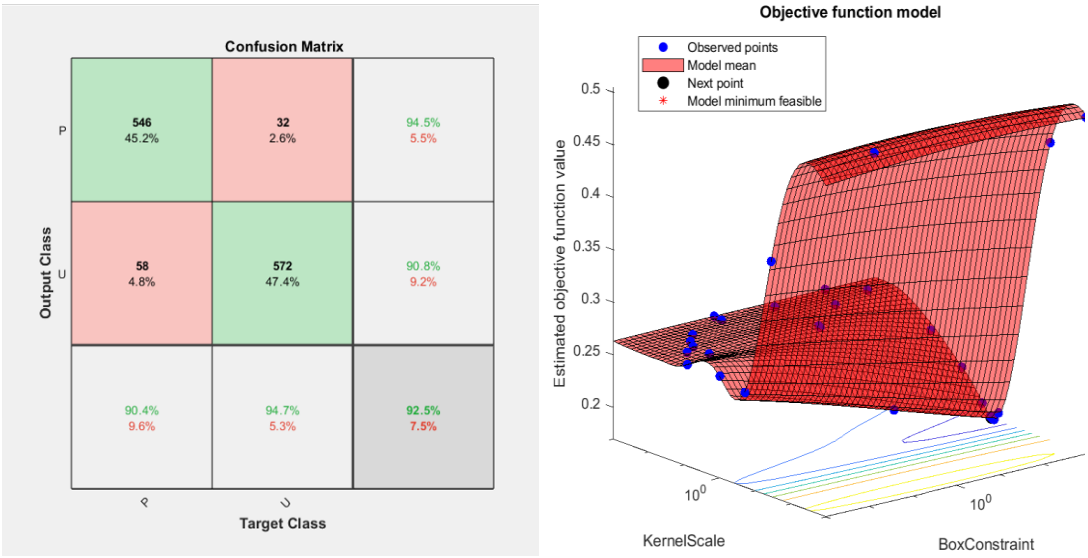
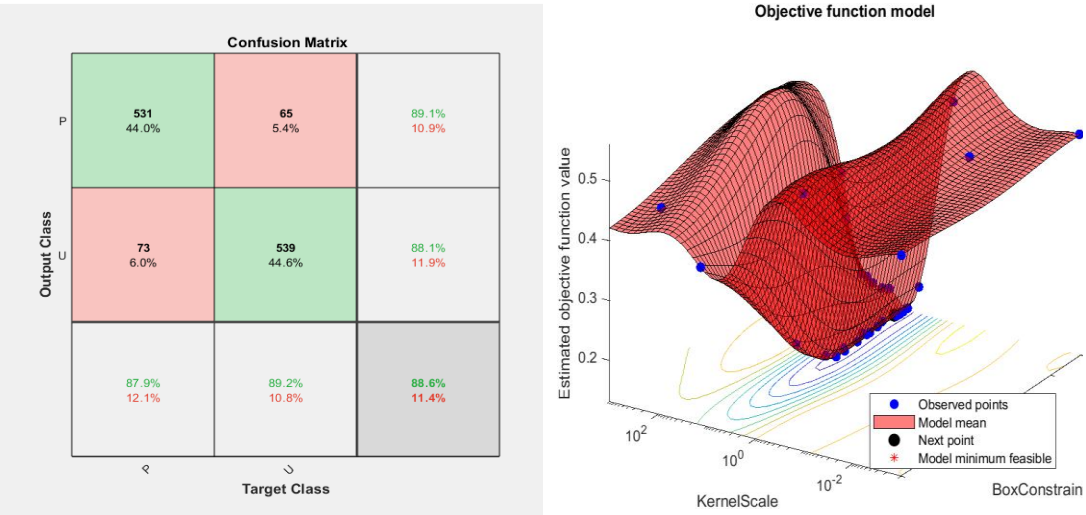


Figure 5.1: Traditional Image Processing Confusion Matrix



(a) Cosfusion Matrix (b) The Objective Function Model

Figure 5.2: SVM (LBP Features) Confusion Matrix and the Objective Function Model of SVM



(a) Confusion Matrix (b) The Objective Function Model

Figure 5.3: SVM (HOG Features) Confusion Matrix and the Objective Function Model with SVM

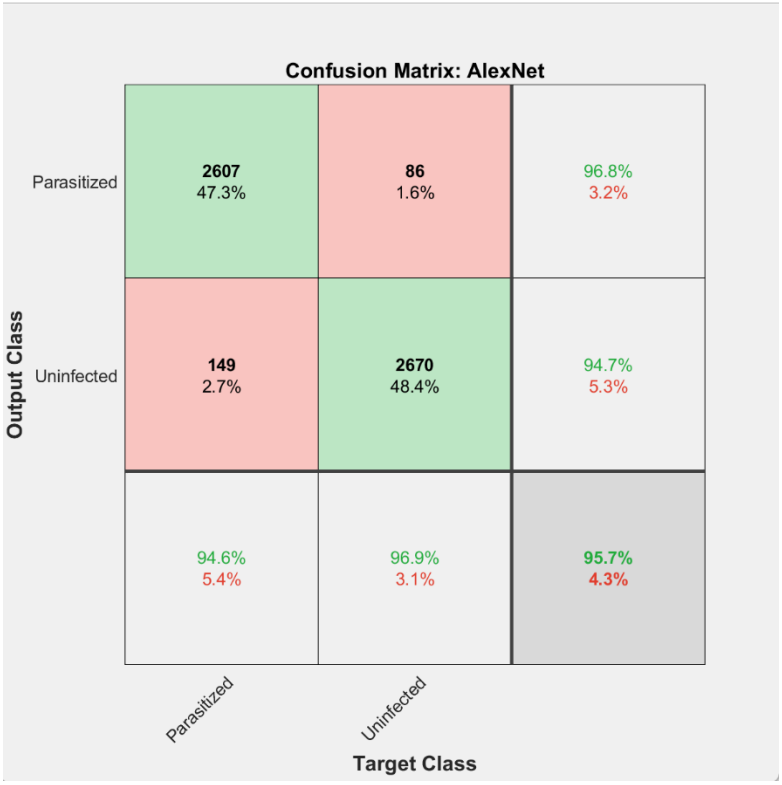


Figure 5.4: CNN-DL Using AlexNet Confusion Matrix (Full set)

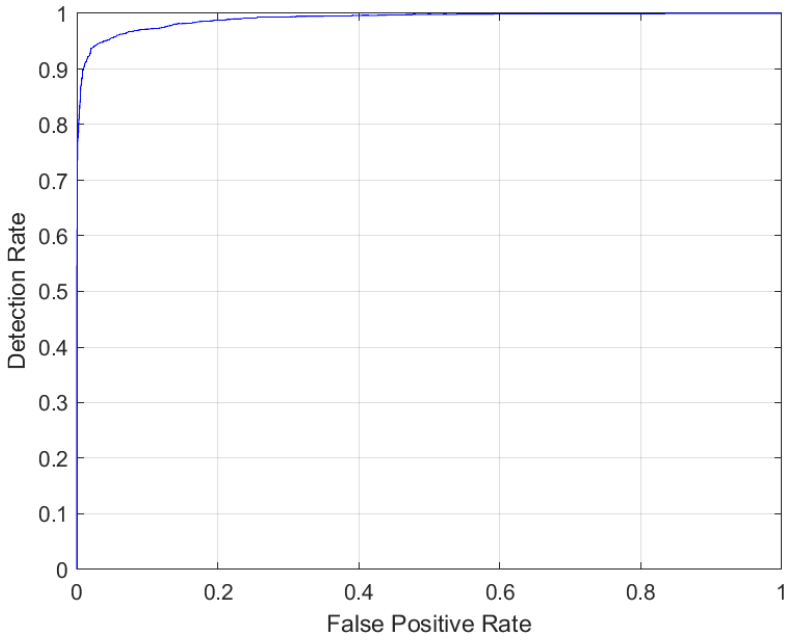


Figure 5.5: ROC for AlexNet (Full set)

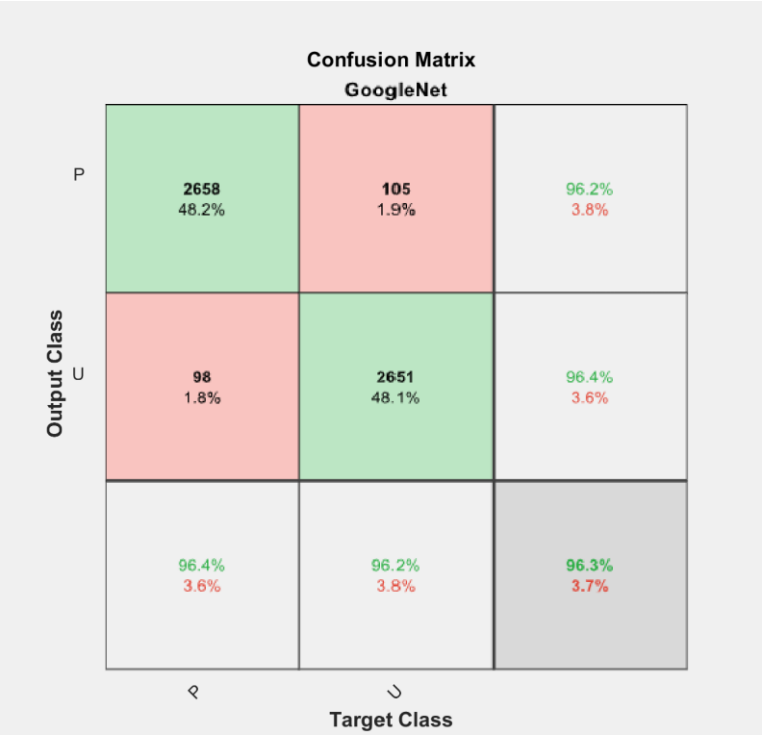


Figure 5.6: CNN-DL Using GoogLeNet Confusion Matrix (Full set)

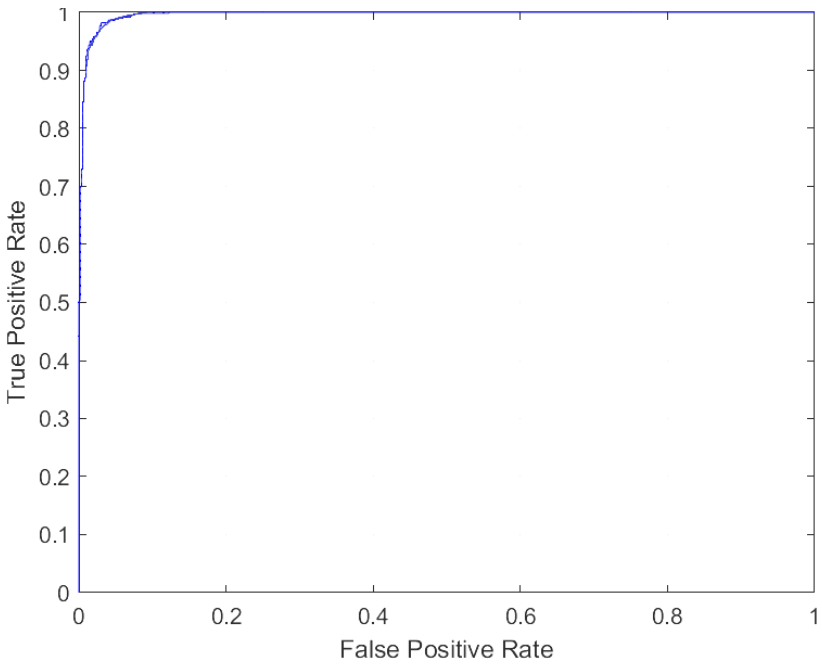


Figure 5.7: ROC for GoogLeNet (Full set)

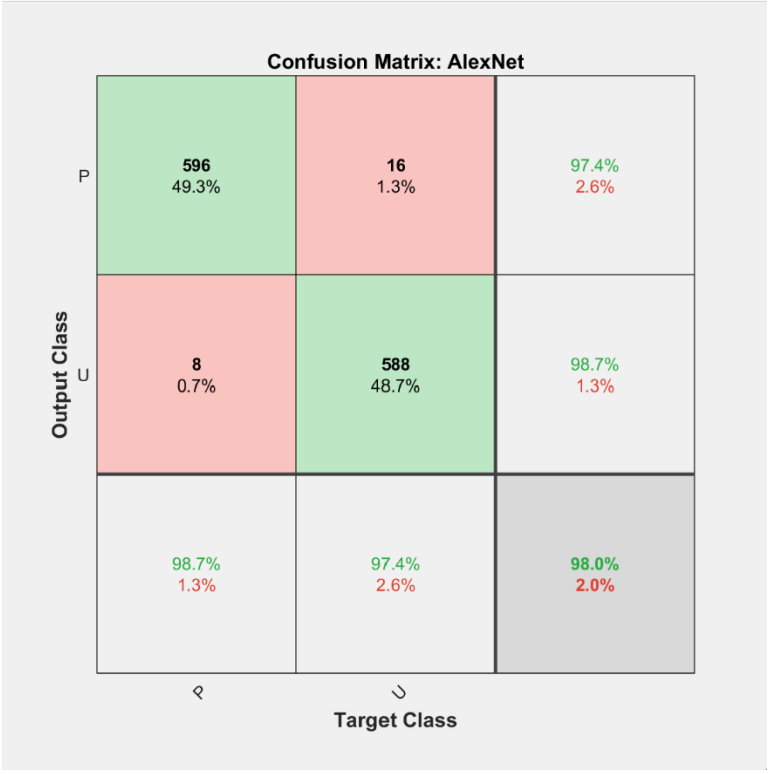


Figure 5.8: CNN-DL Using AlexNet Confusion Matrix (Subset)

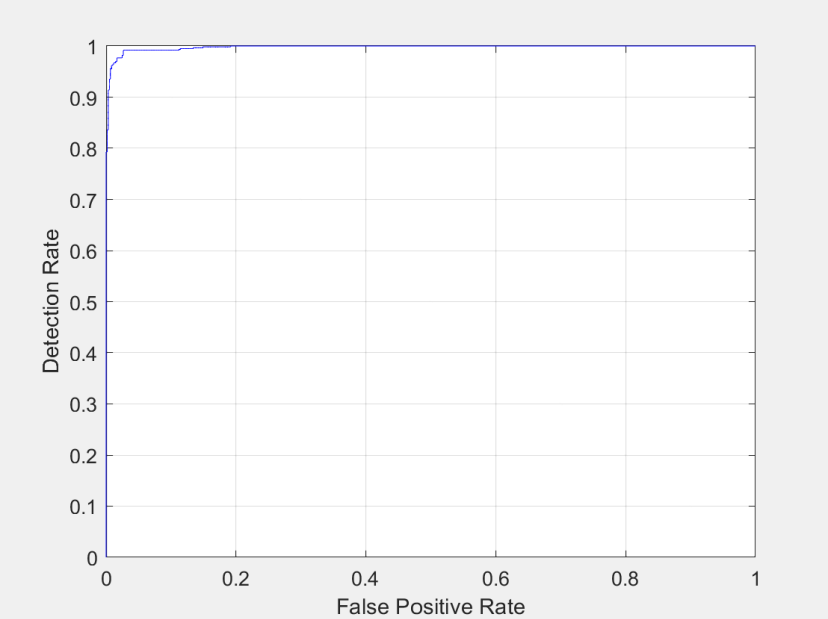


Figure 5.9: ROC for AlexNet (Subset)

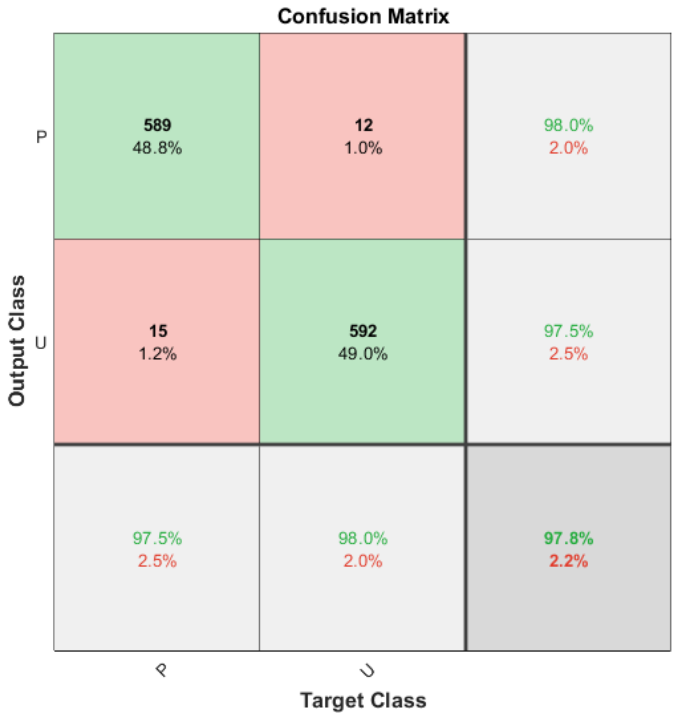


Figure 5.10: CNN-DL Using GoogLeNet Confusion Matrix (Subset)

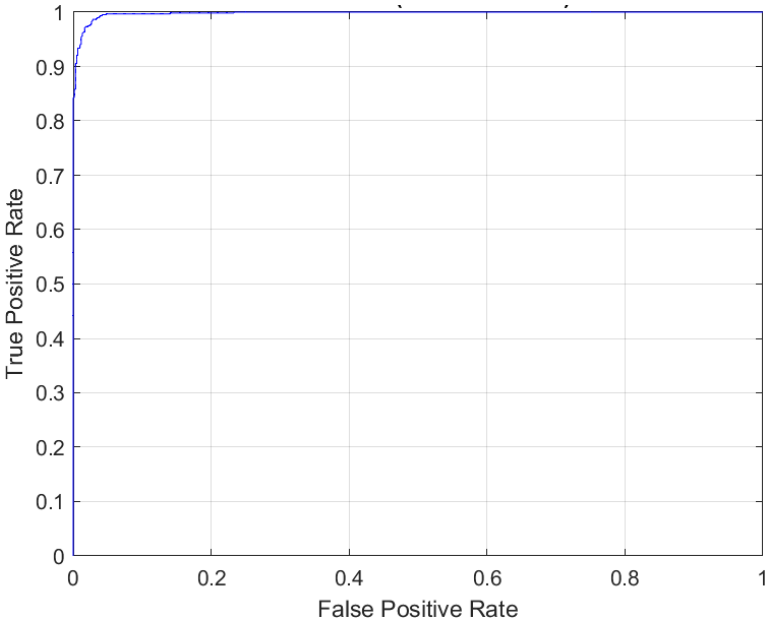
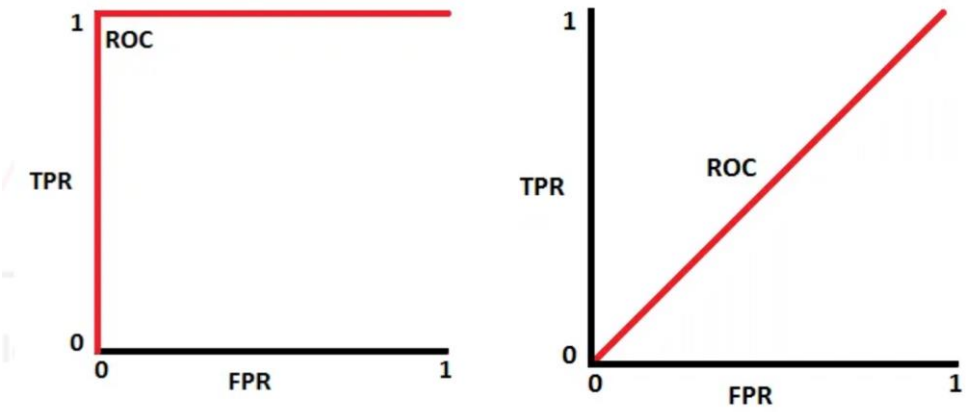


Figure 5.11: ROC for GoogLeNet (Subset)

A ROC curve shows how well a binary classification model performs by plotting the true positive rate vs. the false positive rate graph. To determine how well the model is performing one should take notice of the top left corner of the graph, the higher it is, the better the performance of classification is. This will make the area under the curve which is between 0 and 1 larger so the closer it is to 1 the better the classification performance will be. Figure 5.12 shows an example of a ROC curve with an area under the curve equal to 1 as well as another with area under the curve of 0.5. In our simulations the results for the area under the curve were closer to 1 which means better performing than the ROC curve in Figure 5.12 (b). The area under the curve is usually an indication and summary of the performance metrics.



(a) Perfect AUC (b) 0.5 AUC

Figure 5.12: Different Representations of the ROC Curve

The results indicate that the deep learning method provides the most accurate classification. This is mainly due to the many convolution layers as well as using an architecture that can learn which gives it the ability to improve and optimize during the classification. SVM on the other hand uses only one layer, and the image processing despite being highly accurate is still very prone to human errors in any of the processing steps and not as open to improvement similar to CNN.

In addition, accuracy of the SVM classifier is heavily dependent on the features selected for training. For instance, when HOG features are used the performance of SVM is approximately 4% inferior as opposed to when LBP features are used.

Moreover, we can see in Figure 5.5 for example that AlexNet for the full data set has a smaller area under the curve and further from the left corner compared to GoogLeNet in Figure 5.7

The smaller subset was used to compare the results with the SVM results. However, since the CNN-DL methods are clearly the more efficient and more accurate we

compare their simulation results for the full set. We took accuracy and training time into consideration and found GoogLeNet to be 0.58% more accurate than AlexNet, this however comes with the cost of training time. We notice a considerable difference in training time where GoogLeNet took 925 minutes to finish training whereas it took AlexNet only 40 minutes, this difference is too large considering that all the parameters that affect the training time like the size of the data set and the batch size which affects the number of iterations per epoch are constant for both AlexNet and GoogLeNet. This difference in time is due to the complexity of GoogLeNet in comparison to AlexNet. In Figure 5.13 we can see a comparison of the training time between the DL architectures:

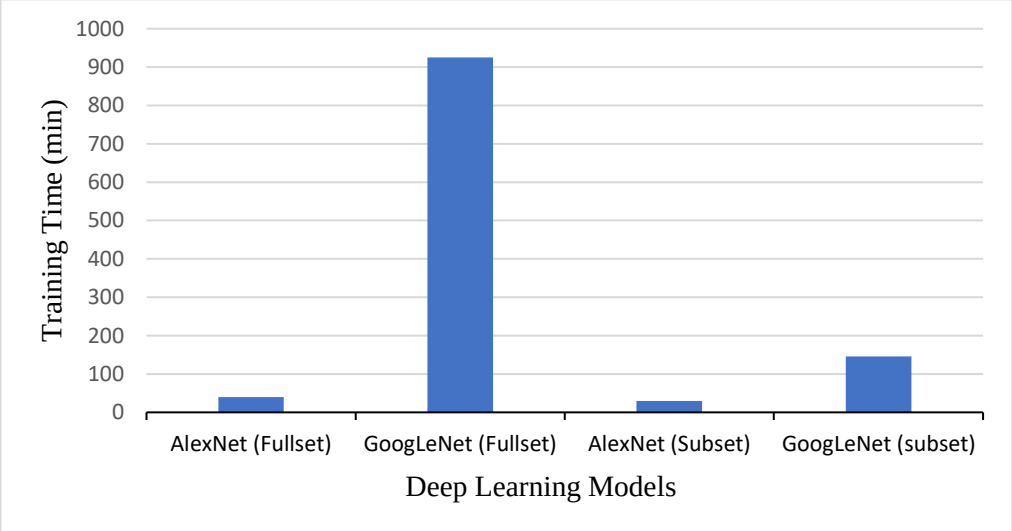


Figure 5.13: Training Time Comparison

Table 5.5 provides a comparison for binary classifier accuracies given the two proposed DL-CNN models and some recent state-of-the-art techniques.

Table 5.5: Proposed DL-CNN models vs. state-of-the-art Malaria detection approaches

Author\Method	Accuracy(%)
Shah et al (Custom CNN) [3]	95%
Vijayalakshmi (VGG19-SVM) [4]	93.1%
Reddy (ResNet-50) [5]	95.91%
Diker (ShuffleNet) [6]	96.44%
Rajaraman et al (ResNet-50) [7]	95.7%
Proposed DL-CCN model#1 (AlexNet)	95.74%
Proposed DL-CNN model#2 (GoogLeNet)	96.32%

5.5 Expanding the National Institute of Health Dataset

The National Institute of Health (NIH) dataset that contains 27,558. Since some deep learning methods work better for larger datasets, providing more diverse set of input samples will make the occurrence of overfitting more difficult, and statistical significance of the results can increase so model can draw more accurate conclusions about the underlying data distribution, leading to better performance on new data.

Expanding the dataset is possible by collecting images of red blood cells (RBCs) from thin blood smears using a segmentation algorithm that makes use of the watershed transform. Watershed transform aims to partition an image into distinct regions or segments based on the intensity or gradient information of the image. It starts by computing the gradient of the image using various methods like Sobel. Next, the

algorithm generate a marker in the image to indicate the starting points for the segmentation process and then labelling those markers to partitions the image into initial regions around each marker. The algorithm iteratively expands the regions from the markers. At each iteration, pixels are assigned to the region with the nearest marker, taking care not to merge regions that belong to different objects. As the regions expand, they at some point come in touch with each other. The points where the regions come in contact is where the watershed lines form. To avoid over-segmentation and create meaningful regions, the algorithm performs a merging step to remove some of the watershed lines and merge adjacent regions based on certain criteria, such as similarity of intensity or texture. This process helps in localizing the individual boundaries of RBCs before they are extracted from the blood smear and saved into designated files based on the “Infected” or “Uninfected” conditions.

Figure 5.14 (a) below shows an image of a thin blood smear and Figure 5.14 (b) depicts a mask generated for segmenting the RBCs. For the overlapping cells in the green box, the watershed transform helps to localize the individual boundaries of each cell.

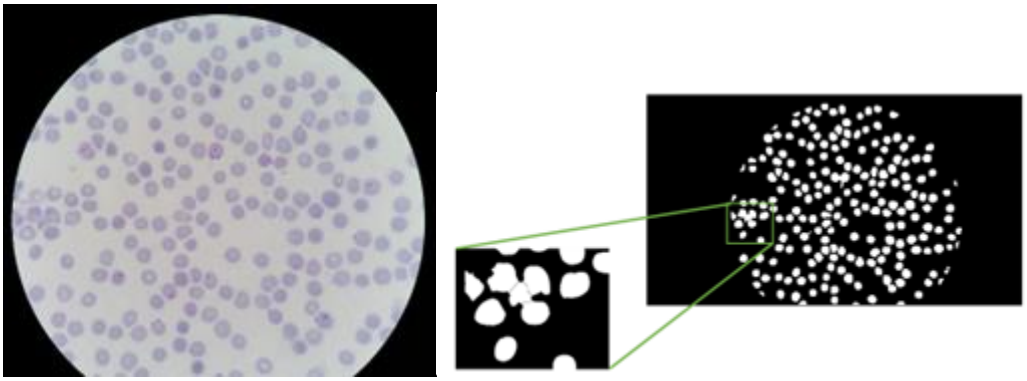


Figure 5.14: Thin Blood Smear and Segmented RBCs Using Watershed Transform

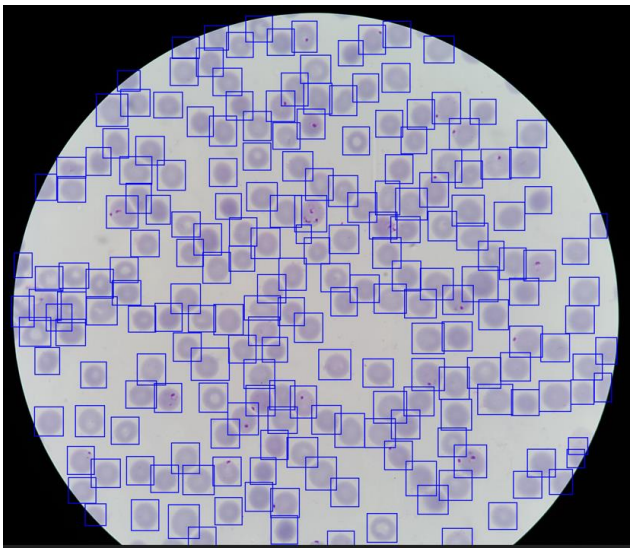


Figure 5.15: Individual Localization of RBCs

Figure 5.15 shows all the overlapping and non-overlapping RBCs that have been localized given a thin blood smear.

Chapter 6

CONCLUSION AND FUTURE WORK

Malaria is a communicable disease transmitted by mosquitoes, and diagnosing it requires a meticulous examination of blood samples under a microscope. This diagnostic process is not only time-consuming but also heavily reliant on the proficiency of pathologists. In recent times, machine learning has emerged as a popular approach for addressing complex real-world problems. In this thesis, we utilized three methods to replace the traditional way of diagnosis in the lab. In those four methods, we managed to successfully classify and diagnose the infected cells to the highest level of accuracy available to us right now. After analysing the results of the simulations, it was clear that the deep learning methods are the more accurate methods, and after comparing two models (AlexNet & GoogLeNet) for large and small datasets it showed that for the larger sets, AlexNet had a slightly lower accuracy when compared with GoogLeNet but it was found to be significantly faster than the GoogLeNet.

The preferred method chosen is AlexNet since the accuracy trade-off for training time is not worth mentioning. We also noticed the same thing for the subset where the difference in accuracy was very insignificant as both models performed almost the same but AlexNet was again faster than GoogLeNet.

Generally, pre-trained deep learning models give an accuracy of 92% to 98% depending on the quality of the images in the set, and the images chosen for training which in our

case are chosen randomly. To achieve even higher accuracy, features can be extracted from a combination of the models and from optimal layers as shown in [26].

6.1 Future Work

Since machine learning algorithms have been designed to work for large amounts of data, in the future the NIH dataset can be expanded by RBCs extracted from thin blood smears collected. For overlapping cells, the watershed transform [27] can be used to localize the individual boundaries of each cell before segmenting them.

An alternative approach would be to employing combinations of deep learning (DL) models such as AlexNet, GoogLeNet, VGG-19, ResNet and SqueezeNet and work out ensemble classification accuracies.

REFERENCES

[1] World Health Organization. “Malaria microscopy quality assurance manual version2”. 2016. [Online]. Available: https://apps.who.int/iris/handle/10665

[2] U. Kumari, M. Memon, S. Narejo, M. Afzal, “Malaria Disease Detection Using Machine Learning”, *2nd International Conference on Computational Sciences and Technologies (INCCT 20),* Jamshoro, Pakistan, Dec 2020.

[3] D. Shah, K. Kawale, M. Shah, S. N. Randive, and R. G. Mapari, “Malaria Parasite Detection Using Deep Learning : (Beneficial to humankind)”, *4th International Conference on Intelligent Computing and Control Systems (ICICCS),* 2020, pp. 984-988, doi: 10.1109/ICICCS48265.2020.9121073.

[4] A. Vijayalakshmi and R. K. B, “Deep learning approach to detect malaria from microscopic images*,” Multimedia Tools and Applications*, vol. 79, no. 21–22, pp. 15297–15317, Jan. 2019, doi: 10.1007/s11042-019-7162-y.

[5] A. S. B. Reddy and D. S. Juliet, “Transfer Learning with ResNet-50 for Malaria Cell-Image Classification,” *2019 International Conference on Communication and Signal Processing (ICCSP)*, Chennai, India, 2019, pp. 0945-0949, doi: 10.1109/ICCSP.2019.8697909.

[6] A. Diker, “Sıtma Hastalığının Sınıflandırılmasında Evrişimsel Sinir Ağlarının Performanslarının Karşılaştırılması”, *Bitlis Eren Üniversitesi Fen Bilimleri Dergisi*, pp. 1825-1835, Dec. 2020, doi:10.17798/bitlisfen.783031

[7] S. Rajaraman et al., “Pre-trained convolutional neural networks as feature extractors toward improved malaria parasite detection in thin blood smear images,” *PeerJ*, vol. 6, p. e4568, Apr. 2018, doi: 10.7717/peerj.4568.

[8] Malaria cell images dataset, A. Arunava. 2018, [Last revision 2019].

[9] T.Nguyen ,T. Nguyen, A.D.Le, “Support Vector Machines - An Overview”. in *International Journal of Advanced Computer Science and Applications*, 2016.

[10] C. J. C. Burges. “A Tutorial on Support Vector Machines for Pattern Recognition”. *Data Mining and Knowledge Discovery*, pp. 121-167, June 1998, doi: https://doi.org/10.1023/A:1009715923555.

[11] ImageNet. March 2021. Available: http://www.image-net.org.

[12] P.wang. “Understanding GoogLeNet model - CNN architecture.” Geeksforgeeks.org. https://www.geeksforgeeks.org/understanding-googlenet-model-cnn-architecture.

[13] T. Ojala, M. Pietikäinen, and D. Harwood. “A comparative study of texture measures with classification based on featured distributions.” *Pattern Recognition*, 29, no.1, pp.51-59, January 1996, doi: https://doi.org/10.1016/0031-3203(95)00067-4.

[14] D. G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints”. in *International Journal of Computer Vision*, 2004.

[15] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. Pearson, 2018.

[16] J. Canny, "A Computational Approach to Edge Detection," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679-698, Nov. 1986, doi: 10.1109/TPAMI.1986.4767851.

[17] A. Malhotra, A. Sankaran, A. Mittal, M. Vatsa, R. Singh. "Fingerphoto Authentication Using Smartphone Camera Captured Under Varying Environmental Conditions", *in Human Recognition in Unconstrained Environments*, pp.119-144, January 2017.

[18] T. Ojala, M. Pietikainen and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971-987, July 2002.

[19] R. Lienhart, A. Kuranov, V. Pisarevsky, "Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection", *in Conf. Pattern Recognition*, Magdeburg, Germany, 2003, pp. 511-518.

[20] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, San Diego, CA, USA, 2005, pp. 886-893 vol. 1, doi: 10.1109/CVPR.2005.177.

[21] C. Cortes and V. Vapnik, “Support-vector networks”, *Machine Learning*, vol. 20, no. 3, pp. 273-297, 1995.

[22] H. Bay,T. Tuytelaars, L. Van Gool, “Speeded-Up Robust Features (SURF)”, *in Computer Vision and Image Understanding (ECCV’06)*, Berlin, Germany, 2006, pp. 404-417, doi: 10.1155/2014/695910

[23] P. F. Felzenszwalb, R. B. Girshick, D. McAllester and D. Ramanan, “Object Detection with Discriminatively Trained Part-Based Models,” *in IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627-1645, September 2010, doi: 10.1109/TPAMI.2009.167.

[24] S. Bangar, “AlexNet Architecture Explained - Siddhesh Bangar - Medium,” *Medium*,Jun.24,2022.[Online].Available:https://medium.com/@siddheshb008/alexnet-architecture-explained-b6240c528bd5

[25] DeepAI, “Stride (Machine Learning),” *DeepAI,* Jun. 2020, [Online]. Available: https://deepai.org/machine-learning-glossary-and-terms/stride

[26] Rajaraman, S. et al. (2019) “*Performance evaluation of deep neural ensembles toward malaria parasite detection in thin-blood smear images*”, PeerJ [Online]. Available at: https://peerj.com/articles/6977/

[27] Chourasiya, S. (2014). Automatic Red Blood Cell Counting using Watershed Segmentation. https://www.semanticscholar.org/paper/Automatic-Red-Blood-Cell-Counting-using-Watershed-Chourasiya/ef24ae99ca56cdb65d66d23cf3767ee8a73972d7

[28] G. Shekar, S. Revathy and E. K. Goud, "Malaria Detection using Deep Learning," *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, Tirunelveli, India, 2020, pp. 746-750.

[29] L. Chen, “Support vector machine — simply explained - towards data science,” *Medium*,Dec.07,2021.[Online].Available:https://towardsdatascience.com/support-vector-machine-simply-explained-fee28eba5496

[30] S. P. K. Karri, D. Chakraborty, and J. Chatterjee, “Transfer learning based classification of optical coherence tomography images with diabetic macular edema and dry age-related macular degeneration,” Biomedical Optics Express, vol. 8, no. 2, p. 579, Jan. 2017, doi: 10.1364/boe.8.000579.

APPENDIX

SIU2023 Published Paper

Ulusal Sağlık Enstitüleri Veri Kümesinden Rastgele Seçilmiş Kırmızı Kan Hücrelerinin Sınıflandırılması

Classification of Red Blood Cells Randomly Selected from Dataset of National Institutes of Health

Wasem Q. Bashi (MS Student), and Erhan A. İnce, (Senior Member, IEEE)

Electrical and Electronics Engineering Department

Eastern Mediterranean University

Famagusta, North Cyprus

21508325@emu.edu.tr , erhan.ince@emu.edu.tr

***Özetçe*— Dünya çapında milyonlarca insan, potansiyel olarak ölümcül bir hastalık olan sıtmadan muzdariptir. Tıbbi durumun başarılı bir şekilde tedavi edilmesi ve yönetilmesi için erken ve kesin teşhis şarttır. Bu makalede, Ulusal Sağlık Enstitüleri (USE) veri setinden rastgele seçilerek oluşturulmuş deneme setleri içindeki kırmızı kan hücrelerinin parazitik olma veya enfekte olmama yüzdelerini belirlemek adına üç bilgisayar destekli yöntem kıyaslanmaktadır. Kullanılan yöntemler sırası ile geleneksel görüntü işleme, destek vektör makinesi (DVM) ve evrişimsel sinir ağları tabanlı derin öğrenme (ESA-DÖ) olmuştur. Benzetimler için 27,558 kırmızı kan hücresi içeren bir veri seti kullanılmıştır. Geleneksel görüntü işleme yöntemi %91,97'lik bir doğruluk ile temiz ve parazitli hücreleri ayırabilmiştir. Yönlendirilmiş gradyanlar histogramı (YGH) özelliklerini kullanan DVM sınıflandırıcının doğruluğu %88.6 iken yerel ikili modeller (YİM) kullanılarak çıkarılan özelliklerle sınıflandırma sonrası doğruluk %92,5'e yükselmiştir. Önceki iki yöntem, %95,7'lik bir doğruluk veren ESA-DÖ sınıflandırıcısıyla karşılaştırıldığında en etkin yöntemin evrişimsel sinir ağları tabanlı derin öğrenme yöntemi olduğu belirlenmiştir.**

Anahtar Kelimeler* — *Destek Vektör Makinesi; Yerel İkili Modeller; Yönlendirilmiş Gradyanlar Histogramı; Evrişimsel Sinir Ağları Tabanlı Derin Öğrenme.

***Abstract*— Millions of people worldwide suffer from malaria, a potentially fatal disease. Early and precise diagnosis is essential for the medical condition to be successfully treated and managed. This paper employs three computer aided methods to determine percentages of red blood cells that are either parasitic or uninfected given test set(s) randomly obtained from National Institutes of Health (NIH) dataset. The three methods employed are traditional image processing, Support Vector Machine (SVM), and Convolutional Neural Networks based Deep Learning (CNN-DL). The simulations were performed using a dataset that had 27,558 images of red blood cells. The traditional image processing method achieves an accuracy of 91.97%. SVM classifier using Histogram of Oriented Gradients (HOG) features had accuracy of 88.6% and with features extracted using Local Binary Patterns (LBP) accuracy had improved to 92.5%. The two previous methods were proved to be inferior when compared with the CNN- DL classification tha gave an accuracy of 95.7%.**

Keywords* — *Support Vector Machine; Local Binary Patterns; Histogram Oriented Gradients; Convolutional Neural Networks Based Deep Learning.

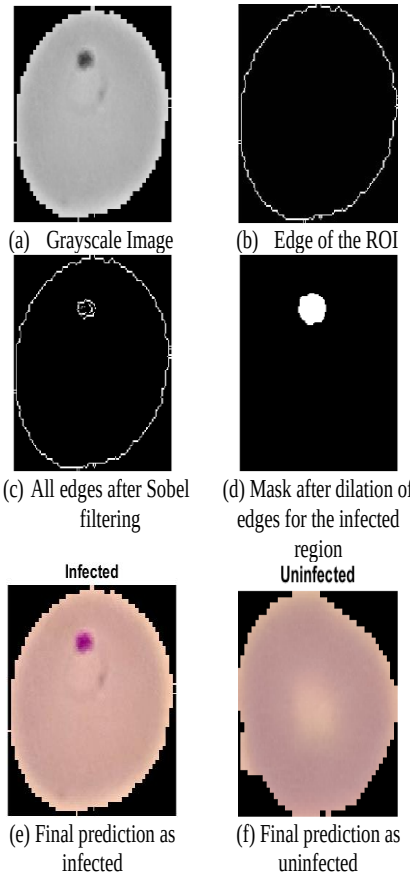
I. INTRODUCTION

Plasmodium is a class of parasitic protozoan that causes the illness referred to as malaria. The red blood cells are infected by this parasite, which is spread by the bites of infected female Anopheles mosquitoes. Malaria is a primary cause of juvenile neuro-disabilities and is particularly harmful to children in Africa, where one dies from it virtually every minute. According to the World Malaria Report 2016 [1], 3.2 billion people in 95 nations and territories are at risk of contracting malaria with 1.2 billion of those people living in sub-Saharan Africa. Every year, skilled microscopists manually count parasites and infected red blood cells in digitized copies of hundreds of millions of blood slides to diagnose malaria. However, microscopy-based diagnostics are not standardized and heavily rely on the microscopist's expertise [1]. In low-resource settings, microscopists often work in isolation without proper training or a system in place to improve their skills, leading to incorrect diagnostic decisions in the field [1]. False-positive instances can result in the unnecessary use of anti-malarial medications and their possible adverse effects, whereas false-negative cases can result in the needless use of antibiotics, a second consultation, missed workdays, and the development into severe malaria.

In this paper, we will employ three computer aided methods to determine what percent of the tested red blood cells from National Institutes of Health (NIH)[2] dataset are parasitic or uninfected. Traditional image processing, support vector machine (SVM) based classification [3-4] and classification based on a machine learning architecture known as AlexNet [13] are the three methods studied. We have also compared AlexNet's classification performance with those of ResNet50 [17,18], VGG-16 and VGG-19[17,18].

I. Traditional Image Processing

This method uses a gradient-based edge detector and morphological operations on the cells images. The data set is from National Library of Medicine (NLM) [2] and contains 27,558 images separated into two classes: parasitic and uninfected. Each class has a total of 13,779 red blood cell images. We start by preparing the images in our dataset by converting them into grayscale. Then region of interest (ROI) in each image is determined by edge detection. A second edge detection is then employed in the ROI using a Sobel filter with a threshold of 0.05 to obtain both the perimeter of the ROI and the edges of the infection (if it exists). The difference between the two edge images would then be dilated to create the mask that shows position of the parasites in the analyzed cell. In the final step, area of the mask is compared with an empirical threshold (100 herein). If the area is larger than the set threshold the cell is classified and marked as infected, otherwise it is considered to be uninfected. Figure 1 below depicts the images obtained after each step.



(a) Grayscale Image (b) Edge of the ROI

(c) All edges after Sobel filtering (d) Mask after dilation of edges for the infected region

(e) Final prediction as infected (f) Final prediction as uninfected

Figure. 1. Classification of Red Blood Cells using Traditional Image Processing.

II. Support Vector Machine Classification

A support vector machine (SVM) is a machine learning algorithm that is commonly used for classification tasks. This algorithm works by analysing labelled data to identify patterns and relationships between features. In binary classification problems, the SVM identifies a hyperplane that divides the two classes, and it chooses the hyperplane that maximizes the distance between t hyperplane and the closest points from each class, known support vectors. SVMs are a popular choice for classification tas because they are effective at handling high-dimensional data a can handle both linearly and non-linearly separable data. They a also less likely to overfit when compared with other machi learning algorithms [3-4]. For the SVM, the dataset will be small with a total of 6,040 images. The reason for smaller dataset is t slow training time of SVM. During our simulations with SVM, v have used two different feature extraction methods:

1- Local Binary Patterns (LBP)
2- Histogram of Oriented Gradients (HOG)

Table I below shows the simulation parameters considered for t SVM classifier used:

TABLE I. SVM SIMULATION PARAMETERS

Parameter	Setting
Total number of images	6040
% of images trained	80%
Cell Size	16x16
Image size	160x160
Kernel Function	RBF

LBP is a technique that helps in identifying patterns in images by analysing the neighbouring pixels and comparing them to the current pixel [5]. It is used to describe the texture of images and can efficiently capture spatial patterns and contrast in grayscale. The function *extractLBPFeatures* under MATLAB returns extracted uniform binary patterns from a grayscale image. The LBP features encode local texture information as explained in [6]. HOG is a feature descriptor used in computer vision for object recognition and detection tasks. It was introduced in 2005 by Dalal and Triggs et.al. It has been shown that HOG is effective in a wide range of applications, including pedestrian detection [7], face detection [8], and object recognition [9]. It has also been used as a feature descriptor in machine learning algorithms, such as support vector machines [10]. One of the strengths of HOG is its ability to capture the shape and structure of an object, regardless of its color or texture. However, HOG is computationally intensive and requires careful parameter tuning to achieve optimal performance. There have been many extensions and variations of HOG proposed over the years, such as multi-scale HOG [11] and dense HOG [12], which aim to address some of these limitations.

III. Deep Learning Using Transfer Learning From AlexNet

As the third computer aided method, we have used AlexNet, based deep learning to classify the images in the NIH dataset.

AlexNet is a convolutional neural network based on a deep learning structure with 8 layers. Five of these layers are convolutional neural networks and the other three are Max Pooling Layers as indicated in Table II. AlexNet was born out as a result of the ImageNet competition in 2012. It is a network that is pre-trained using more than a million images [13]. A prerequisite to using AlexNet or any of the other pre-

parameters we have assumed for the case of CNN-DL. The reasons for choosing AlexNet as the deep learning tool were the following: 1. It does not suffer from vanishing gradients (VG) problem, 2. The ReLU activation function used by AlexNet makes the network 6 times faster.

TABLE II. ALEXNEX STRUCTURAL DETAILS

Layer	#neurons	Filter Size	Stride	Padding	Size of feature map	Activation Function
Input	-	-	-	-	227×227×3	-
Conv 1	96	11×11	4	-	55×55×96	ReLu
Max Pool 1	-	3×3	2	-	27×27×96	-
Conv 2	256	5×5	1	2	27×27×256	ReLu
Max Pool 2	-	3×3	2	-	13×13×256	-
Conv 3	384	3×3	1	1	13×13×384	ReLu
Conv 4	384	3×3	1	1	13×13×384	ReLu
Conv 5	256	3×3	1	1	13×13×256	ReLu
Max Pool 3	-	3×3	2	-	6×6×256	-
Dropout 1	p=0.5	-	-	-	6×6×256	-

TABLE III. CNN-DL SIMULATION PARAMETERS

Parameter	Setting
Total number of images	27,558
% of images for training	80%
Learning rate	1e-4
Image size	227×227
Mini Batch size	64

I. SIMULATION RESULTS

In the three computer aided methods compared herein the effectiveness of the binary classifiers were assessed based on their confusion metrics and also using performance metrics such as Precision, Sensitivity, Accuracy and F1-score. Figure 2 depicts the confusion matrices for the simulated methods and Table IV depicts the classification performance of each method. The values for the specified metrics were calculated using equations (2)-(5):

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

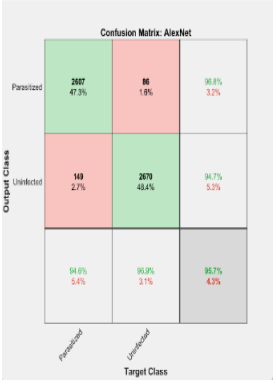
$$\text{Sensitivity} = \frac{TP}{TP + FN} \tag{3}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \times 100\ \% \tag{4}$$

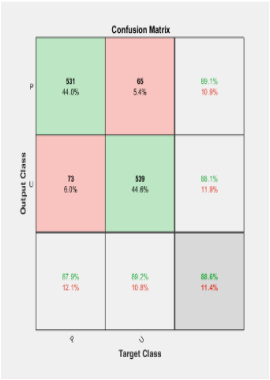
$$F1\text{-score} = 2 \times \frac{(\text{Precision} \times \text{Sensitivity})}{(\text{Precision} + \text{Sensitivity})} \tag{5}$$

where TP denotes true positive, TN is the true negative, FP denotes the false positive, and FN is the false negative.

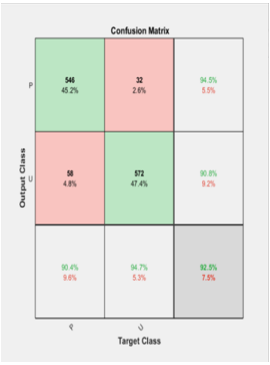
For the traditional image processing the classification is done without splitting the database but for the SVM and AlexNet the data set had to be split to obtain training and testing sets. In this work 80 % of the images were used for training and remaining 20 % were used for testing purposes. In addition, the AlexNet-Deep Learning based simulation used a 5-fold cross validation in order to avoid over fitting. Value of 5 for the k-fold validation was chosen so that bias will not be too large and computational complexity will not be too high.



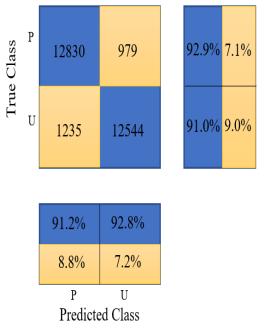
(a) AlexNet CNN Confusion Matrix



(b) Confusion Matrix for SVM with HOG Features



(c) Confusion Matrix for SVM with LBP Features



(d) Confusion Matrix for Traditional Image Processing

Figure. 2. Confusion matrices for the three computer aided methods compared.

TABLE IV. CLASSIFICATION PERFORMANCE OF COMPUTER AIDED METHODS

	TP	TN	FP	FN	Precision	Sensitivity	Accuracy	F1-score
AlexNet	2607	2670	86	149	96.81%	94.59%	95.74%	95.69%
HOG SVM	531	539	65	73	89.09%	87.91%	88.58%	88.50%
LBP SVM	546	572	32	58	94.46%	90.40%	92.55%	92.39%
Image Process.	12800	12544	979	1235	92.89%	91.20%	91.97%	92.04%

Accuracy denotes the ratio of the correctly labeled subjects to the whole pool of subjects. Precision is the actual correct prediction divided by total prediction and sensitivity equals the number of true positives divided by the total number of true positives and false negatives. Finally, *F1*-score is the harmonic mean of the precision and recall.

As can be seen from Table 4 the deep learning model AlexNet provides the most accurate classification with an accuracy of 95.74%. In the literature accuracy for ResNet50 [17,18] is 95.73% and for VGG-16 and VGG-19 [17,18] are respectively 94.5% and 96.09.

Furthermore, accuracy of the SVM classifier is heavily dependent on the features selected for training. For instance, when HOG features are used the performance of SVM is approximately 4% inferior as opposed to when LBP features are used.

I. CONCLUSION & FUTURE WORK

Malaria is a communicable disease transmitted by mosquitoes, and classical way of diagnosing it requires a meticulous examination of blood samples under a microscope. This diagnostic process is not only time-consuming but also heavily reliant on the proficiency of pathologists. In this work, we have compared three computer aided methods to determine which one could possibly replace the traditional way of diagnosis. With each one of the three methods considered, we managed to successfully classify and diagnose the infected cells with different levels of accuracy. Results indicate that the deep learning based CNN approach is the most accurate, and is able to correctly classify the tested red blood cells with an accuracy of approximately 96%. In [19], it was shown that extarcting features from optimal layers and combining different models can help achieve higher ensemble accuracies. Hence as future work authors could try employing these new strategies using combinations of AlexNet, GoogleNet, VGG-19, ResNet and SqueezeNet.

REFERENCES

[1] World Health Organization. (2016). Malaria microscopy quality assurance manual. https://apps.who.int/iris/handle/10665/204266

[2] Arunava, A., "Malaria cell images dataset", National Intitutes of Health, *National Library of Medicine* (NLM). https://lhncbc.nlm.nih.gov/LHC-research/LHC-projects/image-processing/malaria-datasheet.htmlLee

[3] Nguyen, T. T., Nguyen, T. Q., and Le, A. D., (2016). Support Vector Machines - An Overview, In *Inter. J. of Advanced Computer Science and Applic.*,vol.7. no:9, 2016.

[4] Burges, C. J. C., "Tutorial on Support Vector Machines for Pattern Recognition", In *Data Mining and Knowledge Discovery*, vol. 2, no: 2, 1998.

[5] Malhotra, A., Sankaran, A., Mittal, A., Vatsa, M., and Singh, R., " Finger Photo Authentication Using Smartphone Camera Captured Under Varying Environmental Conditions", In *Human Recognition in Unconstrained Environ.*, pp.119-144, 2017

[6] Ojala, T., Pietikainen, M., and Maenpaa, T., " Multiresolution Gray Scale and Rotation Invariant Texture Classification with Local Binary Patterns", In *IEEE Trans. on Pattern Anal. and Mach. Intell*., vol. 24, no. 7, pp. 971-987, 2002.

[7] Dalal, N., and Triggs, B., " Histograms of Oriented Gradients for Human Detection", In *IEEE Comp. Soc. Conf. on Comput. Vis. and Patt. Recognit.*, San Diego, CA, USA, vol. 1, pp. 886-893, 2005.

[8] Lienhart, R., Kuranov, A., and Pisarevsky, A., "Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection", In *Patt. Recognit. Conf.*, 25th DAGM Symposium, Magdeburg, Germany, pp. 511-518, 2003.

[9] Lowe, D.G., "Distinctive Image Features from Scale-Invariant Keypoints", In *Inter. J. of Comput. Vis.*, vol. 60, pp. 91-110, 2004.

[10] Cortes, C., and Vapnik, V., "Support-vector networks", In *Machine Learning*, vol. 20, no: 3, pp. 273-297, 1995.

[11] Bay, H., Tuytelaars, T., and Van Gool, L., "Speeded-Up Robust Features (SURF)", In *Comput. Vis. and Image Understanding (ECCV'06)*, Berlin, Germany, pp. 404-417, 2006.

[12] Felzenszwalb, P. F., Girshick, R. B., McAllester, D., and Ramanan, D., "Object Detection with Discriminatively Trained Part-Based Models", In *IEEE Trans. on Patt. Anal. and Mach. Intell.*, vol. 32, no. 9, pp. 1627-1645, 2010.

[13] Krizhevsky, A., Sutskever, I., and Hinton, G. E., "ImageNet classification with deep convolutional neural networks", In *Advances in Neural Inf. Process. Syst.*, vol. 1, pp. 1097–1105, 2012.

[14] Poostchi, M., Silamut, K., Maude, R.J., Jaeger, S., and Thoma, G., "Image Analysis, and Machine Learning for Detecting Malaria", In *J. of Labor. and Clinical Medic*., vol. 194, pp. 36-55, 2014.

[15] Maduri, P. K., Shalu, S., Agrawal, A. R., and Chaubey, S., "Malaria Detection Using Image Processing And Machine Learning", In *3rd Inter. Conf. on Advances in Comput., Commun. Control and Networking*, pp. 1789-1792, 2021.

[16] Çınar, A., and Yıldırım, M., "Classification of Malaria Cell Images with Deep Learning Architecture", In *J. of Ingénierie des Systèmes d'Information*, vol. 25, pp. 35-39, 2020.

[17] Rajaraman, S., Antani, S.K., Poostchi, M., Silamut, K., Hossain, M.A., Maude, R.J., Jeager, S., and Thoma, G.R., "Pre-trained convolutional neural networks as feature extractors towards improved malaria parasite detection in thin blood smear images", *PeerJ:6:e4568*, http://doi.org/10.7717/peerj.4568.

[18] Jameela, T., Athota, K., Singh, N., Gunjan, V.K., and Kahali, S., "Deep Learning and Transfer Learning for Malaria Detection", Computational Intelligence and Neuroscience, Vol. 2022, Article ID 2221728, https://doi.org/10.1155/2022/2221728.

[19] Rajaraman, S., Jaeger, S., and Antani, S.K., "Performance evaluation of deep neural ensembles towards malaria parasite detection in thin-blood smear images", *Peer J 7:e6977*, http://doi.org/10.7717/peerj.6977.