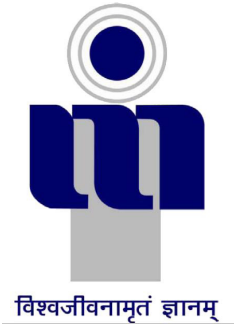
IMPROVED DETECTION OF MALARIA PARASITE IN THIN BLOOD SMEAR IMAGES USING MULTILAYERED 3-DIMENSIONAL CNN

A project report submitted in partial fulfillment of the requirements for B.Tech. Project

B.Tech.

by

Shivendra Pratap Singh (2016IPG-101)
Prakhar Bansal (2016IPG-070)



ABV INDIAN INSTITUTE OF INFORMATION TECHNOLOGY AND MANAGEMENT
GWALIOR-474 010

2019

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS

CNN	Convolutional Neural Network
WHO	World Health Organisation
US	United States
Relu	Rectified Linear Unit
API	Application Program Interface
IP	Internet Protocol
AUC	Area Under Curve
ROC	Receiver Operating Characteristics
TRP	True Positive Rate
FPR	False Positive Rate
GPU	Graphical Processing Unit

NOTATIONS

$\hat{m}$	Bias corrected estimator of first moment
$\hat{v}$	Bias corrected estimators for the second moment
w_t	Updated weight
y_i	Actual output of the $i^{th} training sample$
$\hat{y}_i$	Predicted output of the $i^{th} training sample$
N	Total number of inputs
L	Cross-entropy loss

CHAPTER 1

INTRODUCTION

This chapter includes the details of the background, our problem statement,the motivation and the objectives of our thesis. In this section we briefly describe our project that aims to automate the process of diagnosis of blood cells in order to detect the presence of blood cells infected by Malaria Parasite.

1.1 Background

- The diagnosis of malaria being done accurately heavily depends on human expertise and can be impacted by the variability in observation and the biases of the expert.
- A majority of software based upon computer-aided diagnosis and analysis use Machine Learning techniques with hand-engineered features for decision making. However, this process stated above required human expertise in order to precisely retrieve the features in order to apply machine learning techniques on them.
- But in recent times some advancement have been made in the field of image processing to automatically extract complex features and build classification models based on them.
- At present, researchers across the world have started to apply Deep Learning in the field of health-care and diagnosis to obtain better results in a several associated problems.

1.2 Literature Review

A majority of softwares based upon computer-aided diagnosis and analysis of images use Machine Learning techniques with hand-engineered features for decision-making

[10, 2, 8]. However, the processes stated above required human expertise in order to precisely retrieve the featured in order to apply machine learning techniques on them. Also, these algorithms do not scale well for large data. At present, researchers across the world have started to apply Deep Learning in the field of healthcare and diagnosis to obtain better results in a several associated problems like malaria parasite detection.

Dong et al. compared the performance of pre-trained DL models including LeNet , AlexNet, GoogLeNet and traditional ML based algorithms such as Support Vector Machines towards classifying infected and uninfected cells [3].

A 16-layer CNN was proposed by **Liang et al.** towards classifying the uninfected and parasitized cells [6]. Using the pre-trained AlexNet, features were extracted and then an Support Vector Machines classifier was trained on the extracted features. This model worked better than the pre - trained model. Due to the lack of computational resources, the resolution of images was reduced for smoother running.

A deep belief network of 6 layers was introduced by **Nair, Bibin et al (2017)** for malaria parasite diagnosis in peripheral blood smear images [1]. The accuracy of this model was a staggering 96.4 percent.

A customized convolutional neural network model was used by **Gopakumar et al. (2018)** for testing videos which contained a focus stack of Leishman stained slide images for the process of detecting the parasites [4].

Although the outcomes that resulted from the attempts of previously stated methods are promising, the current existing methods need to maintain their results for larger datasets of images by doing cross-validation testing at the patient level.

1.3 Problem Statement

The problem at hand is to build an automated system using deep learning that would detect the presence of malaria infected cells in the blood stream of patients using thin blood smear images. We also aim to deploy the above system using a web application.

1.4 Motivation

The motivation for our project is based on the nature and fatality of the disease.

- According to World Health Organization (WHO), there are over 200 million malaria cases and there are approximately 400,000 deaths due to malaria every year [9]. Due to this we need a fast, easy and effective way to detect malaria and diagnose the disease.

- a malaria parasite can stay in blood for over a year, a timely diagnosis can prove to be really helpful. Moreover, a large number of people can be diagnosed at

once due to lesser requirement of human expertise.

- Here is where the use of CNN's have proven to be really effective. These models work well with large training data and we don't need to do feature extraction by ourselves.

1.5 Objectives

The prime objective of our thesis work is to build a CNN (Convolutional neural network) based deep learning model that will accurately detect the presence of malaria infected cells in the bloodstream of the patients. This would require -

- Developing a CNN based Deep Learning Model that would classify the images of thin blood smear slides into either infected or uninfected category.
- Deploying our model using a web based application in which users will be able to upload the images of the red-blood cells (thin blood smear slides) and the app will display the results to the user.
- Also if the user is found to be infected with the disease, then a mail will automatically be sent to the users giving them the locations of nearby health-care and diagnosis centres for their treatment.

CHAPTER 2

DESIGN AND METHODOLOGY

To analyze any wireless ad hoc network we need a reference model. To understand the concept of ad hoc network we have considered the model which is discussed below.

2.1 DATASET

The dataset for our project is taken from the **U.S National Library of Medicine** [7]. The dataset consists of thin blood smear slide images from the Malaria Screener research activity.There are a total of 27,558 cell images with equal instances of parasitized and uninfected cells.The original dataset available is divided into three datasets - for training, for validation and for testing purposes in the ratio 0.8, 0.1 and 0.1 respectively. In each of these datasets, the ratio of infected and uninfected images is same to prevent baisedness towards any particular class during training, validation and testing . The images in the dataset are of variable dimensions so scaling is required before feeding it to the main model.

2.2 WORKING PROCEDURE

The main goal of our project is to build an automated model based on Convolutional Neural Network that would timely and correctly detect the presence of malaria infected cells in the blood stream of patients. The patients will be able to use our model through a user friendly web application.Our model will be deployed by the web application at the backend. The user will be required to upload the images of thin blood smear slides. Our model will then run on the uploaded image and make the prediction.The DFD below describes our work-flow briefly-

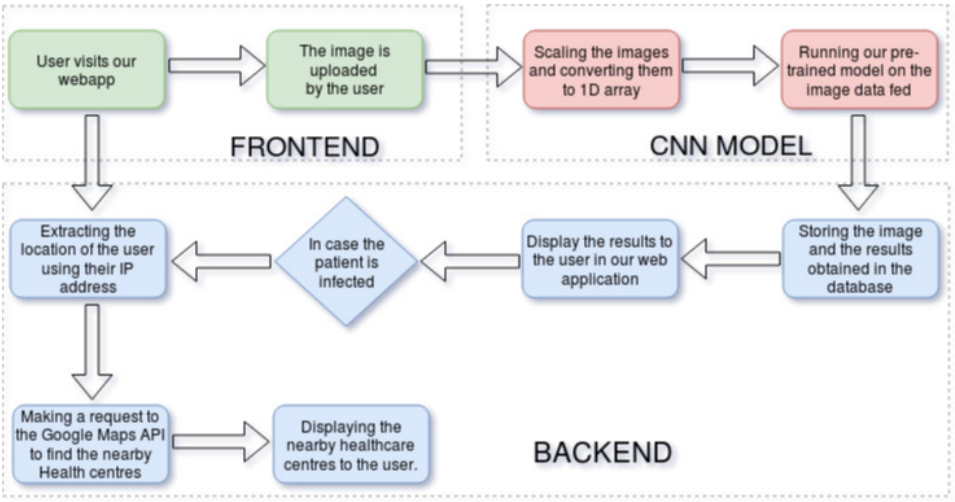


Figure 2.1: Data Flow Diagram

2.2.1 MODEL DESCRIPTION

Our CNN model consists of the following types of layers -

- **Input Layer** - This layer consists the dimensions similar to the dimensions of input image provided. Our input layer is of the dimension (50x50x3). All the images have been re-sized to the dimension (50x50x3) to create consistency in the data.
- **Hidden Layers** - These layers takes input from the previous layer and produces more complex output features.
- **Output Layer** - The layer uses the output of the previous hidden layer as input and produces 2 outputs in the form of probabilities corresponding to each class. The class with the higher probability is the correct answer.
- **Activation Function** - Except the output layer, the output of every other layer uses Relu activation function. The final output layer uses sigmoid function to do the binary classification.

The model uses the Adam optimizer to optimize the parameters of the layers.The optimizer used is Adam Optimizer given by: [5]

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}$$

$$\hat{v}_t = \frac{v_t}{1-\beta_2^t}$$

$$w_t = w_{t-1} - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}$$

The loss function used is binary cross entropy given by:[11]

$$\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) = -\frac{1}{N}\sum_{i}^{N}[y_i \log \hat{y}_i + (1 - y_i)\log(1 - \hat{y}_i)]$$

The model uses regularization for controlling the training data to overfit the model which may later result in reduced accuracy on the test set. We use dropout method to carry out the regularization.

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 48, 48, 32)	896
max_pooling2d_1 (MaxPooling2	(None, 24, 24, 32)	0
batch_normalization_1 (Batch	(None, 24, 24, 32)	128
dropout_1 (Dropout)	(None, 24, 24, 32)	0
conv2d_2 (Conv2D)	(None, 22, 22, 32)	9248
max_pooling2d_2 (MaxPooling2	(None, 11, 11, 32)	0
batch_normalization_2 (Batch	(None, 11, 11, 32)	128
dropout_2 (Dropout)	(None, 11, 11, 32)	0
conv2d_3 (Conv2D)	(None, 9, 9, 32)	9248
max_pooling2d_3 (MaxPooling2	(None, 4, 4, 32)	0
batch_normalization_3 (Batch	(None, 4, 4, 32)	128
dropout_3 (Dropout)	(None, 4, 4, 32)	0
flatten_1 (Flatten)	(None, 512)	0
dense_1 (Dense)	(None, 512)	262656
batch_normalization_4 (Batch	(None, 512)	2048
dropout_4 (Dropout)	(None, 512)	0
dense_2 (Dense)	(None, 2)	1026

Total params: 285,506
Trainable params: 284,290
Non-trainable params: 1,216

Figure 2.2: Architecture of the model

CHAPTER 3

ACTIVITIES COMPLETED

3.1 ACTIVITIES COMPLETED

- An **intermediate model** for the purpose of detection has been implemented using **Convolutional Neural Network**. The intermediate results have been observed and recorded.
- The web app that will be used to deploy the model is partially implemented. At the back-end of our web application, we are able to request Google Maps RestAPI to return the nearby health centres for the treatment of malaria. A javascript code has been implemented at the back-end to fetch the location (Latitude and Longitude) of the user using the IP address.
- The backend for uploading images and running the model on the uploaded image has been implemented.
- Testing of the model has been done for various test images.

CHAPTER 4

INTERMEDIATE RESULTS

In this chapter, we tend to produce the results that we achieved from our custom build multilayered CNN.

4.1 Accuracy

Our model achieved an accuracy of about **96%** on our training dataset and about **95%** on our testing dataset.

4.2 Confusion Matrix

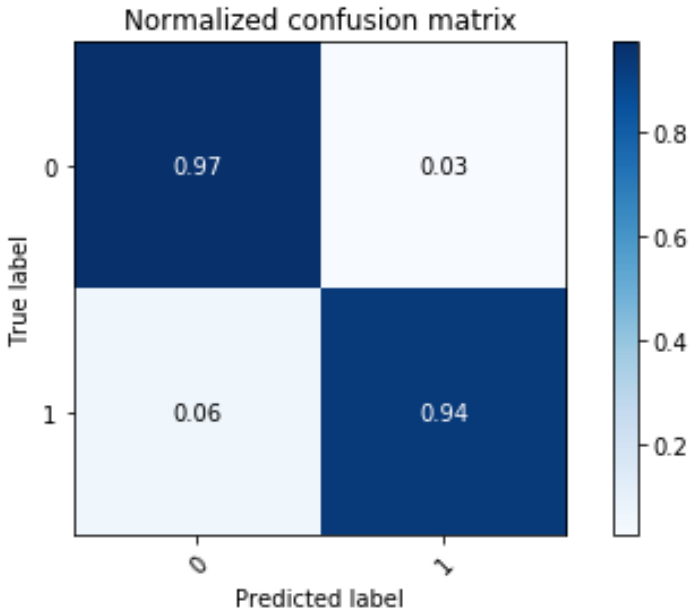


Figure 4.1: Confusion Matrix.

In the above matrix, 0 represents the uninfected cells whereas 1 represents the infected cells. Around 94% of the total infected cells were correctly classified whereas the remaining 6% of them were classified incorrectly. Similarily around 97% of the total uninfected cells were correctly classified whereas remaining 3% were incorrectly classified.

4.3 AUC score and ROC curve

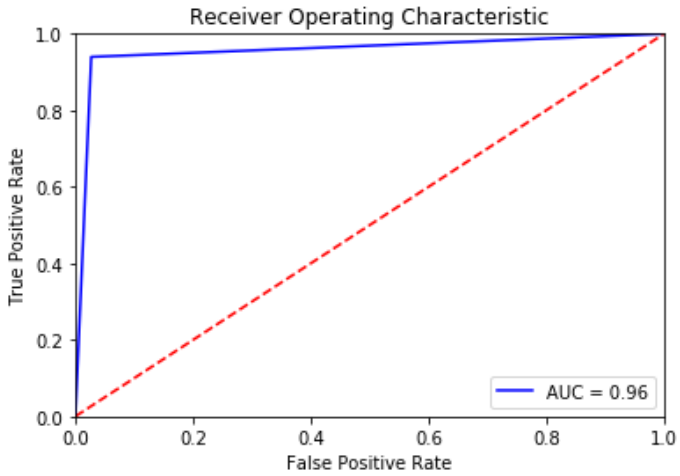


Figure 4.2: ROC Curve

AUC or area under the curve is a performance measurement that denotes the degree of separability i.e how good our model is in differentiating between the classes. Higher the AUC, better is the model. The AUC score for our model is 0.96 which is pretty good. ROC or the receiver operating characteristics represents the probability curve. It is plotted by keeping the TPR on y-axis and FPR on the x-axis. The TPR rate is also called as sensitivity and the FPR rate is also called as the fall-out. Thus, we can also say that the ROC curve is sensitivity as a function of fall out.

CHAPTER 5

FUTURE ACTIVITIES

- **Improving Accuracy of our model by making use of Transfer learning** - We intend to apply transfer learning to help improvise the model the model. Genrally, the custom build model for image processing can be a bit resource extensive. They can take a lot of time to train if not provided with modern **GPUs** for processing. Thus, we use ready to use models which are pre-trained on some different tasks. These models can we used used to train the pre-trained model further to produce more accurate results.
- **Building a front End for our Web Application** - We intend to build a user friendly front-end to simplify the process of detection for the patients. Also features at the backend such as geocoding and reverse geocoding are to be implemented.

REFERENCES

[1] Bibin, D., Nair, M. S. and Punitha, P.: 2017, Malaria parasite detection from peripheral blood smear images using deep belief networks, *IEEE Access* **5**, 9099–9108.

[2] Das, D. K., Ghosh, M., Pal, M., Maiti, A. K. and Chakraborty, C.: 2013, Machine learning approach for automated screening of malaria parasite using light microscopic images, *Micron* **45**, 97–106.

[3] Dong, Y., Jiang, Z., Shen, H., Pan, W. D., Williams, L. A., Reddy, V. V., Benjamin, W. H. and Bryan, A. W.: 2017, Evaluations of deep convolutional neural networks for automatic identification of malaria infected cells, *2017 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*, IEEE, pp. 101–104.

[4] Gopakumar, G. P., Swetha, M., Sai Siva, G. and Sai Subrahmanyam, G. R. K.: 2018, Convolutional neural network-based malaria diagnosis from focus stack of blood smear images acquired using custom-built slide scanner, *Journal of biophotonics* **11**(3), e201700003.

[5] Kingma, D. P. and Ba, J.: 2014, Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980* .

[6] Liang, Z., Powell, A., Ersoy, I., Poostchi, M., Silamut, K., Palaniappan, K., Guo, P., Hossain, M. A., Sameer, A., Maude, R. J. et al.: 2016, Cnn-based image analysis for malaria diagnosis, *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, IEEE, pp. 493–496.

[7] *Malaria Datasets*: n.d.
URL: *https://ceb.nlm.nih.gov/repositories/malaria-datasets/*

[8] Poostchi, M., Silamut, K., Maude, R. J., Jaeger, S. and Thoma, G.: 2018, Image analysis and machine learning for detecting malaria, *Translational Research* **194**, 36–55.

[9] Rajaraman, S., Antani, S. K., Poostchi, M., Silamut, K., Hossain, M. A., Maude, R. J., Jaeger, S. and Thoma, G. R.: 2018, Pre-trained convolutional neural net-

works as feature extractors toward improved malaria parasite detection in thin blood smear images, *PeerJ* **6**, e4568.

[10] Ross, N. E., Pritchard, C. J., Rubin, D. M. and Duse, A. G.: 2006, Automated image processing method for the diagnosis and classification of malaria on thin blood smears, *Medical and Biological Engineering and Computing* **44**(5), 427–436.

[11] Zhang, Z. and Sabuncu, M.: 2018, Generalized cross entropy loss for training deep neural networks with noisy labels, *Advances in Neural Information Processing Systems*, pp. 8778–8788.