A project report on

Hybrid Feature Extraction and Ensemble Modeling for Malaria Parasite Classification: A Comprehensive Approach

Submitted in partial fulfillment for the award of the degree of

Bachelor of Technology in Computer Science and Engineering

by

AVISI SACHAN(20BCE1770)

VIT®

Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)
CHENNAI

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

April,2024

Hybrid Feature Extraction and Ensemble Modeling for Malaria Parasite Classification: A Comprehensive Approach

Submitted in partial fulfillment for the award of the degree of

Bachelor of Technology in Computer Science and Engineering

by

AVISI SACHAN (20BCE1770)

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)
CHENNAI

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

April, 2024

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

CHENNAI

DECLARATION

I hereby declare that the thesis entitled “Hybrid Feature Extraction and Ensemble Modeling for Malaria Parasite Classification: A Comprehensive Approach” submitted by me, for the award of the degree of Bachelor of Technology in Computer Science and Engineering, Vellore Institute of Technology, Chennai is a record of bonafide work carried out by me under the supervision of Dr. Leninisha Shanmuham.

I further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place: Chennai

Date: Signature of the Candidate

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

CHENNAI

School of Computer Science and Engineering

CERTIFICATE

This is to certify that the report entitled **“Hybrid Feature Extraction and Ensemble Modeling for Malaria Parasite Classification: A Comprehensive Approach”**is prepared and submitted BY Avisi Sachan(20BCE1770) to Vellore Institute of Technology, Chennai, in partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology in Computer Science and Engineering programme** is a bonafide record carried out under my guidance. The project fulfills the requirements as per the regulations of this University and in my opinion meets the necessary standards for submission. The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma and the same is certified.

Signature of the Guide:

Name: Dr./Prof.

Date:

Signature of the Internal Examiner	Signature of the External Examiner
Name:	Name:
Date:	Date:

Approved by the Head of Department,
B.Tech. CSE

Name: Dr. Nithyanandam P

Date: (Seal of SCOPE)

ABSTRACT

This study addresses the pressing need for efficient and accessible malaria detection and classification systems, particularly in regions with limited resources and high disease prevalence. With millions affected annually, accurate diagnosis and classification are essential for effective treatment and transmission control. Traditional methods like rapid test kits and microscopic blood smear analysis have limitations, prompting the exploration of automated solutions using pretrained convolutional neural network (CNN) models. However, challenges arise due to the scarcity of reliably labelled data. This research aims to develop a preprocessing methodology and identify the most suitable pretrained CNN models to accurately classify malaria parasites in blood smears as Uninfected, Falciparum, or Vivax, thus facilitating improved diagnosis and treatment in resource-constrained settings.

ACKNOWLEDGEMENT

It is my pleasure to express with deep sense of gratitude to Dr. Leninisha Shanmuham, Associate Professor School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, for her constant guidance, continual encouragement, understanding; more than all, her taught me patience in my endeavor. My association with her is not confined to academics only, but it is a great opportunity on my part of work with an intellectual and expert in the field of computer vision

It is with gratitude that I would like to extend my thanks to the visionary leader Dr. G. Viswanathan our Honorable Chancellor, Mr. Sankar Viswanathan, Dr. Sekar Viswanathan, Dr. G V Selvam Vice Presidents, Dr. Sandhya Pentareddy, Executive Director, Ms. Kadhambari S. Viswanathan, Assistant Vice-President, Dr. V. S. Kanchana Bhaaskaran Vice-Chancellor i/c & Pro-Vice Chancellor, VIT Chennai and Dr. P. K. Manoharan, Additional Registrar for providing an exceptional working environment and inspiring all of us during the tenure of the course.

Special mention to Dr. Ganesan R, Dean, Dr. Parvathi R, Associate Dean Academics, Dr. Geetha S, Associate Dean Research, School of Computer Science and Engineering, Vellore Institute of Technology, Chennai for spending their valuable time and efforts in sharing their knowledge and for helping us in every aspect.

In jubilant state, I express ingeniously my whole-hearted thanks to Dr. Nithyanandam P, Head of the Department, B.Tech. CSE and the Project Coordinators for their valuable support and encouragement to take up and complete the thesis.

My sincere thanks to all the faculties and staffs at Vellore Institute of Technology, Chennai who helped me acquire the requisite knowledge. I would like to thank my parents for their support. It is indeed a pleasure to thank my friends who encouraged me to take up and complete this task.

Place: Chennai

Date: **Avisi Sachan**

CONTENTS

CHAPTER 1

INTRODUCTION

CHAPTER 2

BACK GROUND

CHAPTER 3

DATASET

CHAPTER 4

PREPROCESSING

CHAPTER 8

PROCESS DIAGRAMS

CHAPTER 9

RESULTS AND DISCUSSION

CHAPTER 10

COLCLUSION AND FUTURE WORK

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

CNN	Convolutional neural network
RESNET	Residual Network
GWN	Gray World Normalization

Chapter 1

Introduction

1.1 MALARIA

Malaria is a serious, life-threatening disease that affects millions every year. According to the consensus of WHO in the year 2023, a total of 167 million people were infected with the disease. Africa, being one of the most severely affected nations, accounted for 75% of these cases. The disease affects those with weak or compromised immune systems to a greater degree, which can be seen from the fact that out of the 700,000 to 2.3 million deaths from malaria each year, 75% of them are young children and women. Detection and treatment of malaria are vital not only to cure the disease but also to reduce its high transmission. Malaria has four types of viral species: Falciparum, Vivax, Ovale, and Malariae, with Falciparum and Vivax being the most common ones. Early parasite detection and classification in the infected can ensure not only the appropriate course of treatment but also guards against anti-malarials being used injudiciously, which is crucial in co-endemic areas.

1.1.1 MALARIA DETECTION

Malaria detection is usually performed with the help of rapid test kits or through microscopic blood smear image analysis by trained professionals. While Rapid test kits can provide parasite detection, they cannot classify the virus into one of the four categories. Blood Smear detection can be performed with i) Thin Blood Smears and ii) Thick Blood Smear. Thick blood smears are a concentrated level of dehemoglobinized red blood cells; this high density of cells allows for easier and more efficient parasite detection but does not provide a good environment for studying parasite morphology. Thin blood smears analysis can provide parasite classification in the positively detected samples.

1.1.2 IMAGE PROCESSING SYSTEM

Due to the high skill demand and Malaria being a prevalent disease in developing countries, it is important to develop a system that can process blood smear data with accuracy and proficiency by anyone with basic knowledge of technology. Pretrained CNN models like VGG16, ResNet50, etc., can provide ways to address the problem; however, the study suffers from a problem with the collection of reliably labelled data. The purpose of this study is to tackle the problem of finding an efficient preprocessing methodology and to figure out the most efficient pretrained models to build a reliable system to classify parasites accurately with limited availability of data and accurately label blood smears as Uninfected, Falciparum, and Vivax.

Chapter 2

Background

2.1 Introduction

Malaria detection and classification has been a prevalent problem with ongoing research for making use of image processing techniques to solve it. Neural networks are used to extract features from blood smear images to train models to be able to detect malaria. The blood smear features for different stages and parasites of Malaria are very nuanced and thus need extensive research and experimentation to train the models efficiently.

2.2 Literature Survey

A good amount of research has been done in the field of Malaria detection and classification which used different image processing models and techniques to detect malaria parasites in blood smear images

2.2.1 Implementation of Malaria Parasite Detection and Species Classification Using Dilated Convolutional Neural Network

The study classified samples into different species of malaria using 3 convolutional layers, and a convolution2D convolution operation, ReLU for activation function, and a dilation rate of 2 as hyper-parameters. For these layers a dilation rate of 2 was used, the study used a publicly available dataset of 27699 for detecting parasites and performed with an accuracy of 99.9%. The study was able to classify the virus at an accuracy of 99% for Falciparum, 64.6% for Malariae, 39.1% for Ovale, and 37.3% for Vivax. The study had low and unreliable levels of accuracy for Vivax which is also one of the two most common forms of Malaria, also the study does not experiment with different pre-processing to make the dataset more reliable and remove any unnecessary features from the dataset.

2.2.2

The study experimented with deep convolutional networks such as DenseNet121, DenseNet201, ResNet152V2, NasNetLarge, MobileNetV2, and the hybridized DenseNet201 and ResNet152V2. The study showed good results with MobileNetV2 computing the highest accuracy, and ResNet152V2 having the best loss value by 0.005. For this study data was collected from three sources. Watershed segmentation was used which is a digital image processing technique used for segmenting objects in an image based on the topography of the image intensity. The study had good accuracy with these deep CNN models, but training these models takes a large amount of time and processing power and the lack of a big enough dataset can lead to overfitting

2.2.3

Studies like these used transformer models for malaria parasite detection, and the study used transformer models and generative adversarial networks for multi-class plasmodium classification and malaria detection. A Residual Convolutional Neural Network utilizing a Bayesian method was used to classify blood cell images as infected or uninfected. The study deals with new ways of dealing with the issue of lack of data by using GANs but falls short in ensuring data reliability, as labeling of the generated data is not cross-verified by biologists or medical professionals models trained on such data cannot be relied on for real-world usage. The data available of labeled blood smears also contains wrongly labelled data thus generating images using that data cannot be relied on.

2.2.4

The study investigates the problem of choosing the best pre-processing methodology for malaria blood smear images. With microscopic images regular image pre-processing involving sharpening and noise reduction can lead to the loss of certain features and the exaggeration of others. The study runs CNN models on three iterations of the same data once with no processing, then with Colour Normalization: Gray-world Normalization, and lastly with Colour Normalization: Comprehensive Normalization. The study uses data of thin smear blood images from researchers at Lister Hill National Center for Biomedical Communications (LHCBC), on this data the study post-pre-processing does binary classification to do parasite detection and label the blood smears as infected or uninfected. Through this study, it was determined that Gray World Normalization provides better accuracy. The study provides a good insight into the preprocessing techniques and their working, but it does not determine their effect on parasite classification.

2.2.5

The study researches the shortcomings of the traditional malaria detection and classification techniques. To resolve this, it experiments with Convolutional Neural Networks, specifically the VGG-16 pre-trained model. The study uses this model for feature extraction and classifies malaria into four subspecies. After feature extraction, SVM is used for training and testing. This three-phase model structure provided with 93% accuracy in detecting the virus type. With this accuracy, the study provided with a good insight into model structures that can be used to tackle the issue, but left room for improvement in making the results more reliable.

2.2.6

Chromatin found in Eukaryotic cells is a complex of DNA and protein, Chromatin profiling can help us access this chromatinized DNA. The study does a deeper analysis of the effect of malaria on genetic material to study how it's regulated and organized. For this regulatory sequence patterns were used including TF binding, chromatin accessibility, and histone modification profiles. The study developed a framework MalriaSED, which was tested to be utilized to look into epigenetic regulatory implications of reported noncoding genetic variations. In conclusion, it was observed how malaria exploits the immune system and leads to drug resistance. This technology studied further can help scientists develop drugs for personalized attacks to the virus in the way it best counters the effect it has on the body. Malaria Detection, classification, and studying the virus all together can provide a way for fast and efficient recovery of the patients.

2.2.7

As researched in the study, understanding Malaria's effects on Chromatin is an important part of understanding the virus, but another vital step is to analyze the stage of the disease. Malaria has 3 stages, and in [7] a dataset comprising of the three classes(stages) of Falciparum is used for developing a model trained to detect the stage of the disease. For the study, the dataset was first put through image enhancement and augmentation and used deep convolutional networks like GoogleNet and VGG-19. The primary focus of the disease is to detect malaria at its schizont stage, which is an early stage and makes recovery easier. To achieve this goal a 10-fold cross-validation method was used, and fine-tuning was used to alter the pre-trained model to better fit the needs of the study. It concluded that GoogleNet gave better performance than VGG-19 with an accuracy of 97.48%. Building a reliable model for accurately predicting the stage of disease can revolutionize the recovery approach and can also be vital for better drug management. To facilitate a system that can predict the stage of a disease, we also need a fast and reliable methodology to detect the virus and classify it into one of the four categories. Unless we establish this by the time detection and classification are performed by traditional methods, the detecting stage will become futile as it may have progressed.

2.2.8

This is another study that aimed at classifying blood smear images into falciparum, malariae, and vivax. For the study in total, a dataset of 90 images was used. Three SVM models were then compared by training and testing on this dataset. Out of the three, Linear SVM, Polynomial SVM, and Gaussian SVM, Gaussian SVM outperformed with an accuracy of 86.67%. The study lacks in credibility as the dataset it trained the models on is not diverse enough. This can lead to overfitting and an unreliable prediction system. Also, having a dataset this small can lead to the model not being trained to identify a diverse range of geographically modified strains of the virus. Due to a small training data resulting in poor approximation, the model needs to be enhanced by training on a larger set of blood smear images.

2.2.9

In traditional diagnosis of Malaria, staining the blood smear film is an important part of the process, it is specifically critical for the identification of the malaria species. For malaria, Geimsa stain is used commonly and is also considered the most reliable. It can be used for staining both thick and thin blood smears. Giemsa solution which is used for this staining technique is composed of eosin and azure (methylene blue). This study investigates the importance and advantage of staining in the process of malaria classification, different components of the dye help in enhancing features of the blood smear and making the identification process easier. Eosin stains the parasite nucleus red, and methylene blue stains the cytoplasm blue. The paper focuses on taking a dataset of stained blood smears of Falciparum and Vivax-infected samples. A seven-stage algorithm was derived, and the dataset was pre-processed to reduce noise and improve image differentiation. This would allow for an easier feature extraction. The seven stages included Image Acquisition, Extraction of RBC, Detection of edge, Binary Image, RBC Counting, Thresholding, and Extraction of the parasite. This study gives important insight into valuable pre-processing for better parasite detection. However, to attain this type of data with enough geographical, stage, and parasite diversity we would require a large amount of sample collection and a large number of trained professionals to prepare the stained blood smear films.

2.2.10

This research studies to build a system that can accurately detect the virus, as well as determine the virus's kind and stage of infection. Automating the whole process makes the treatment process more efficient and faster. The study faces the huddle of detecting malaria in the early stage due to its complex structure. To facilitate the process the database images were pre-processed with GM and median filter, this would help make the feature extraction process simpler. Segmentation was done using the U-Net model, specifically focusing on segmenting red blood cells (RBC) in blood smear images for malaria diagnosis. With the VGG model, the study got an accuracy of 95.5%, an accuracy of 95% in classifying the cells as infected or uninfected, and an accuracy of 90.17% in classifying the species.

Chapter 3

Dataset

3.1 Introduction

The study uses two datasets, one with a total of 27,558 which had 13,779 samples of both infected and uninfected thin blood smear samples. This dataset was for parasite detection. The other dataset was collected from a variety of sources and has 2895 images of Uninfected, Vivax, and Falciparum, with 965 images in each category. The images are of dimensions 5312 X 2988pixels, with optical zooms to optical microscope images.

3.1 Parasite Detection Dataset

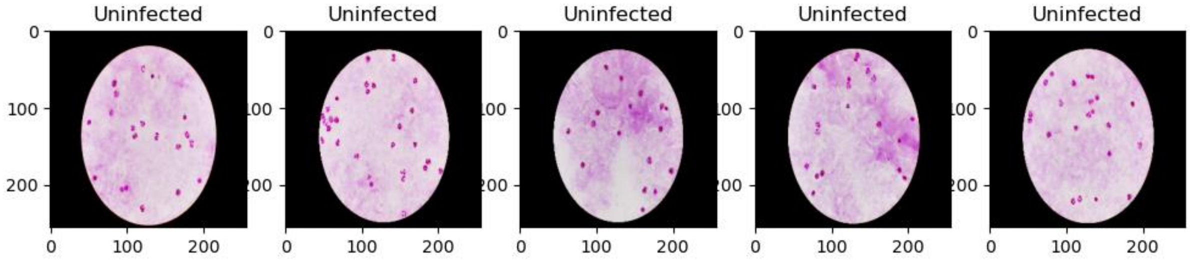


Figure 1 PARASTITE DETECTION IMAGE DATA

3.2 Parasite classification Dataset

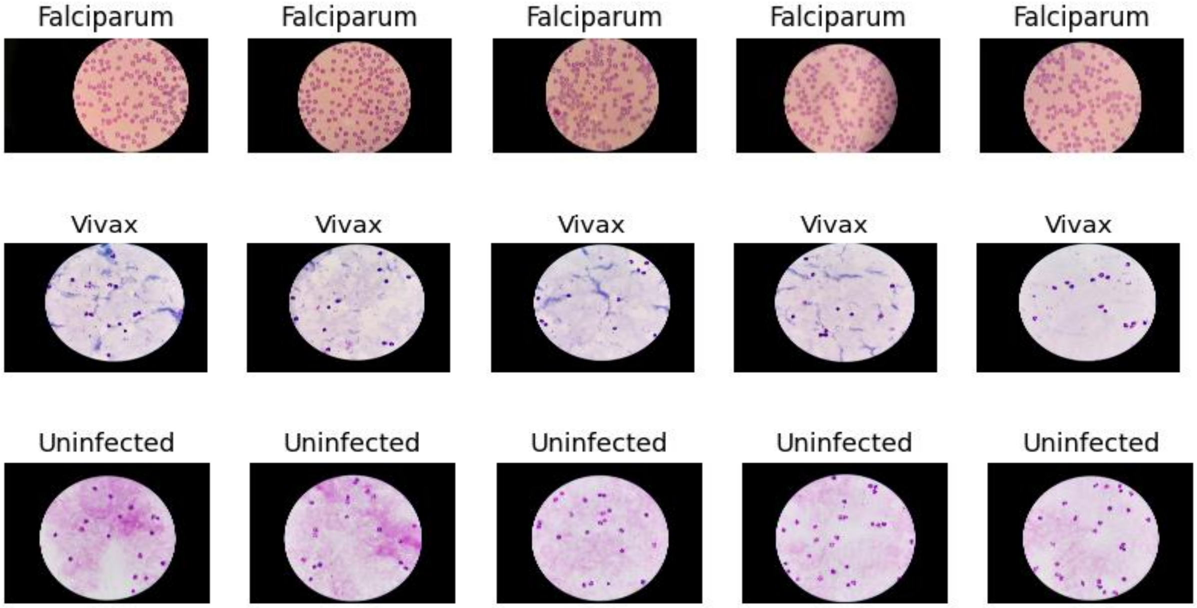


Figure 2 PARASITE CLASSIFICATION IMAGE DATA

Chapter 4

Preprocessing

4.1 Introduction

Image preprocessing can help in better identification of the features which can lead to easier and more reliable detection and classification of the parasite. For the study the models were trained and tested on three forms of the dataset.

4.2 Types of Pre-processings

1. Unprocessed
2. Noise reduction, sharpening, and Histogram Equalization
3. Gray World Normalization

4.2.1 Gaussian Blur

Gaussian blur is an image processing technique used to reduce noise and smooth images. In the project, it is applied during preprocessing to minimize high-frequency noise by averaging pixel values around each pixel, emphasizing the central pixel. By removing fine details and noise, Gaussian blur enhances image quality, making it easier for subsequent steps to focus on vital cell features. The application of Gaussian blur in malaria classification is crucial for improving accuracy and reliability by reducing noise and artifacts in blood cell images.

4.2.2 Sharpening

Sharpening filtering is an image processing method that accentuates edges and fine details by enhancing rapid changes in pixel intensity. This technique is valuable for highlighting subtle image features, improving local contrast, and aiding in precise feature recognition, such as identifying specific characteristics in malaria-infected blood cell images. Sharpening filters are beneficial in malaria classification as they enhance the visibility of critical features within blood cell images. By accentuating edges and fine details, these filters make it easier for machine learning models to recognize specific characteristics of infected and uninfected cells, like staining patterns and cell morphology.

4.2.3 Histogram Equalization

Histogram equalization is an image processing technique used to enhance the contrast and overall visibility of details in an image. It works by redistributing the intensity values of an image's pixels to cover the entire available range. This process can help bring out fine details in both dark and light regions of the image, making it particularly useful when dealing with images that have uneven lighting or limited contrast. Histogram equalization contributes to improved image quality, making it easier to identify and analyse features within the image, which is valuable in various image processing and computer vision applications. Histogram equalization enhances image contrast and feature visibility, which is crucial in malaria classification. By addressing uneven lighting and low contrast issues in blood cell images, it ensures that staining patterns and cell texture variations are discernible. This improvement aids machine learning models in accurately distinguishing between infected and uninfected cells, ultimately contributing to more reliable malaria classification results.

4.2.4 Gray world Normalization

Gray world normalization leads to better outcomes in detecting malaria from thin smear blood sample images. In gray scale normalization it is initially assumed the the global average colour of the image is a shade of gray, this works on the assumption that the intensity in any image of the red, green and blue channel are equal. To process the image the process starts with calculating the average intensity of each of the aforementioned color channels. By calculating this the scaling factor for making the intensity of these channels equal can be calculated. The scaling factor is applied to each of the pixel in the image and thus a volor balanced version of the image is attained. The formula for gray-world normalization and comprehensive normalization:

$$R_{new} = \frac{R_{old}}{R_{avg}}, G_{new} = \frac{G_{old}}{G_{avg}}, B_{new} = \frac{B_{old}}{B_{avg}}$$

$$R_{new} = \frac{R_{old}}{R_{old}+G_{old}+B_{old}} \quad G_{new} = \frac{G_{old}}{R_{old}+G_{old}+B_{old}} \quad B_{new} = \frac{B_{old}}{R_{old}+G_{old}+B_{old}}$$

Figure 3 GWN FORMULA

The new pixel intensities for the red, green, and blue pixels are denoted by the letters Rnew, Gnew, and Bnew. The original pixel intensities for the red, green, and blue pixels are, respectively, Rold, Gold, and Bold. The average pixel intensities for red, green, and blue pixels are, respectively, Ravg, Gavg, and Bavg.

4.3 ImageDataGennerator

The study also uses ImageDataGenetaror It is used to create data generators for both the training and testing datasets. These generators apply various transformations and augmentations to the image data to improve model generalization and performance. Two generators are used one for train data and one for test data with parameters such as rescale, zoom_range, horizontal_flip and rotation_range. After configuring the data generators, the flow method is used to generate batches of preprocessed and augmented data. For the training data generator, it takes the training features X_train and labels y_train and specifies a batch size of 64. The purpose of using data generators is to efficiently feed batches of data to the neural network during training and testing, making it possible to work with large datasets without loading all data into memory at once. Data augmentation techniques applied during training can help improve the model's ability to generalize and perform well on new, unseen data.

Chapter 5

Parasite Detection

5.1 Introduction

The study performed parasite detection on the dataset with 27,558 images. In this stage binary classification is performed on a model trained on this infected and uninfected data. The process starts with image acquisition, after which the image data is pre-processed with noise reduction, filtering and Histogram equalization. This augmented data is used to train and test two models: CNN and VGG-19. Multiple factors are used to verify the working of the model which include: training accuracy, training loss, validation accuracy, validation loss. These help in making sure that the model gives a reliable accuracy and is also not subjected to over or under fitting.

5.1.1 CNN

CNN stands for Convolutional Neural Networks, they come in the class of deep neural networks, and are an important tool for image classification and analysis. There are multiple components to a CNN model the core one being the Convolutional layer, the operations performed in this block involves calculating the dot product of the sliding filter and the local region of the input image. After this calculation the next step is the Activation function this step is responsible for introducing non-linearity, some of the commonly used activation functions include ReLu which is also the one used in the study, other than which there are Sigmoid and Tanh. Being a deep neural network it is important to control the chances of overfitting in the model which is done with the help of Pooling Layer. The final two layers are Fully Connected Layer and Flattening layer which function to produce the final output by connecting each neuron of the layer to the previous layer which gives a way for complex mapping. The flattening layer is used to convert the output of previous layers into a 1D vector for FC layer.

5.1.2 VGG-19

VGG-19 was proposed by the Visual Geometry Group, similar to CNN it is a deep convolutional neural network. The model is very useful for large scale image recognition. As its name suggests the model in total has 19 layers. Out of these 19, 16 are convolutional layers and the rest three are fully connected layers. The model makes use of 3X3 convolutional filters in the convolutional layer and for max pooling it employs a 2X2 filter with a stride of 2 which is used for down sampling the spatial dimensions. Similar to CNN the study uses ReLu as the activation function for VGG-19 which introduces non-linearity. The fully connected layers of VGG-19 comprise of 4096 neurons each which is followed by a softmax layer for the purpose of classification.

Chapter 6

Parasite Classification

6.1 Introduction

For the purpose of parasite classification, the study divided the process in two parts, one on the dataset without any pre- processing and one on the dataset with guassian blur and sharpening. Multiple Deep convolutional networks were trained and tested on these two forms of a dataset containing 956 images per each class of Falciparum, Vivax and Uninfected. This classification would lead to models being able to predict whether a patient has malaria or not, and if the sample is infected it would classify it into one of the two leading types of malaria: Falciparum and Vivax. Multiple factors are used to verify the working of the model which include: training accuracy, training loss, validation accuracy, validation loss. These help in making sure that the model gives a reliable accuracy and is also not subjected to over or under fitting. The models used for this part of the study are:

6.1.1 VGG-19

VGG-19 was proposed by the Visual Geometry Group, similar to CNN it is a deep convolutional neural network. The model is very useful for large scale image recognition. As its name suggests the model in total has 19 layers. Out of these 19, 16 are convolutional layers and the rest three are fully connected layers. The model makes use of 3X3 convolutional filters in the convolutional layer and for max pooling it employs a 2X2 filter with a stride of 2 which is used for down sampling the spatial dimensions. Similar to CNN the study uses ReLu as the activation function for VGG-19 which introduces non-linearity. The fully connected layers of VGG-19 comprise of 4096 neurons each which is followed by a softmax layer for the purpose of classification.

6.1.2 Xception

Xception which stands for Extreme Inception in another deep convolutional neural networks model used in the study to determine it’s effectiveness in classifying the blood samples. Xception as the name suggests is a variation of Inception architecture. To make it more computationally effective the modules of a regular Inception model are changed with Depth-Wise separable convolutions. With these modifications it becomes possible to achieve great efficiency with reduced number of parameters.

6.1.3 InceptionV3

Another model used for the purpose of the study is InceptionV3, it’s a convolutional neural network developed by Google. The model makes the task of image classification simpler by increasing efficiency and effectiveness. Inception modules, batch normalization, and factorization are some of the techniques used for the purpose of effective image classification. These layes also help in speeding us the classification process without compromising on reliability. The convolutional filters in such a model are used parallelly, which results in better capturing of features.

6.1.4 ResNet-152

ResNet152 is a deep convolutional neural network architecture which was developed by Microsoft.

The model was developed with the purpose of addressing gradient problem. The architecture of the mosel includes deep stack of residual blocks, and each of these consists of multiple convolutional layes which are followed by skip connections which serve the purpose of letting the gradients flow more directly through the network. The model provised an efficient and effective architecture for image classification, and is an excellent model of the ResNet family.

Chapter 7

Hybrid Model for Feature Extraction with Ensemble Learning for Classification

7.1 Introduction

The study uses a hybrid InceptionResnetV2 model for feature extraction. This is done on all three forms of data i) Unprocessed ii) Gaussian blur and sharpening iii) Gray world normalization. Using Feature Extraction simplifies the complexity of extracting features from the image data as it can be too complex to directly analyse the voluminous pixel values. Feature Extraction also reduces the issue of overfitting by doing the same. This happens as this reduces the chances of the model learning from the noise or spurious correlations present in the image data. Multiple factors are used to verify the working of the model which include: accuracy, precision, recall, confusion matrix. These help in making sure that the model gives a reliable accuracy and is also not subjected to over or under fitting.

7.2 InceptionResNetV2

InceptionResNetV2 is a hybrid deep convolutional neural networks architecture. It is a combination of Inception and ResNet family architectures which was developed by Google. It takes the residual networks from Resnet architectures and incorporates with them the inception modules which help in reducing the training speed and performance while the residual networks help Gradient problem. InceptionV3 and ResNet architecture components are combined in this design to provide a deep neural network with enhanced training capabilities and performance.
The architecture integrates components from ResNet and InceptionV3 as follows:

1. Stem Block: The network's first layers are similar to those of InceptionV3. A stem block made up of convolutional layers and max-pooling operations precedes the layers.

2. Initiation Blocks (such as Blocks 35, 17 and 8): These bricks bear some small alterations from those found in Inception V3. They are made up of parallel convolutional branches that have varying numbers of filters and kernel sizes. The network can record features at various resolutions and scales thanks to these blocks.

3. Residual Connections: Residual connections—akin to those found in ResNet—are added to each Inception block. The network can alleviate the vanishing gradient problem and make training very deep networks easier because to these links.

4. Final Convolution Block: A convolutional block, global average pooling, and a classification layer come next in the network's conclusion. This section is similar to InceptionV3's last layers, which handle global pooling and classification.

Inception-ResNet-V2, which is relatively deep and simple to train, provides state-of-the-art performance on a variety of computer vision applications by merging elements from InceptionV3 (the Inception blocks) and ResNet (the residual connections).

With the features extracted by InceptionResNetV2, ensemble learning models: Random Forest and XG Boost are trained by splitting the features into training and testing.

7.3 Ensemble Model

In machine learning, ensemble models are a potent paradigm that combine the advantages of several distinct models to enhance prediction performance. The voting ensemble is a popular kind of ensemble model in which a final prediction is derived by summing the predictions of multiple base models. While in regression tasks the final forecast could require averaging the predictions, in classification tasks the final prediction is typically decided by a majority vote among the individual models. This method produces forecasts that are more reliable and accurate by utilising the diversity of the base models to lessen the shortcomings of any one model.

Bootstrap aggregating, sometimes known as "bagging," is another well-liked ensemble approach. Training several copies of the same base model on various subsets of the training data—usually sampled with replacement—is known as bagging. Bagging lowers variance and enhances generalisation performance by averaging these models' predictions. The Random Forest technique, which creates an ensemble of decision trees trained on bootstrapped samples of the training data, is a well-known example of bagging. By voting or averaging, each tree adds to the final forecast, resulting in scalable and reliable models.

Bootstrap aggregating, sometimes known as "bagging," is another well-liked ensemble approach. Training several copies of the same base model on various subsets of the training data—usually sampled with replacement—is known as bagging. Bagging lowers variance and enhances generalisation performance by averaging these models' predictions. The Random Forest technique, which creates an ensemble of decision trees trained on bootstrapped samples of the training data, is a well-known example of bagging. By voting or averaging, each tree adds to the final forecast, resulting in scalable and reliable models.

The study uses the following models for training and testing purpose:

7.3.1 Random Forrest

Random Forest is a type of an ensemble learning model which can be used for the purposes of classification, regression etc. the architecture of the model includes multiple decision trees which are trained on different subsets of the data. The prediction of this architecture depends on the votes of each individual tree, the final outcome depends on majority voting for classification.

7.3.2 AdaBoost

Adaptive Boosting, or AdaBoost, is a potent ensemble learning technique that builds a strong classifier by combining several weak learners. Using the same dataset, iteratively trains a series of weak classifiers, increasing the weight assigned to the incorrectly categorised data points with each iteration. AdaBoost modifies its learning technique by concentrating on the cases that were previously misclassified, hence enhancing overall performance. The final prediction is then formed by summing the weights of the predictions made by these weak classifiers. Due to its reputation for managing intricate datasets, preventing overfitting, and achieving excellent accuracy, AdaBoost is a well-liked option for classification tasks across a range of industries.

Chapter 8

Process Diagram

8.1 Introduction

A process diagram is a graphic representation of the stages or actions needed to finish a given activity or arrive at a desired result. It usually includes of a number of components, like nodes that stand in for the distinct steps and lines or arrows that connect them to show how the process flows. Process diagrams are extensively utilised for the documentation, analysis, and optimisation of workflows and procedures in a variety of sectors, such as business, engineering, and project management. Stakeholders can better understand how activities are carried out, spot possible bottlenecks or inefficiencies, and streamline operations to increase overall effectiveness and efficiency by visually mapping out the processes involved in a process. Process diagrams are useful tools that organisations can use for decision-making, communication, and ongoing improvement projects.

8.2 Parasite Detection

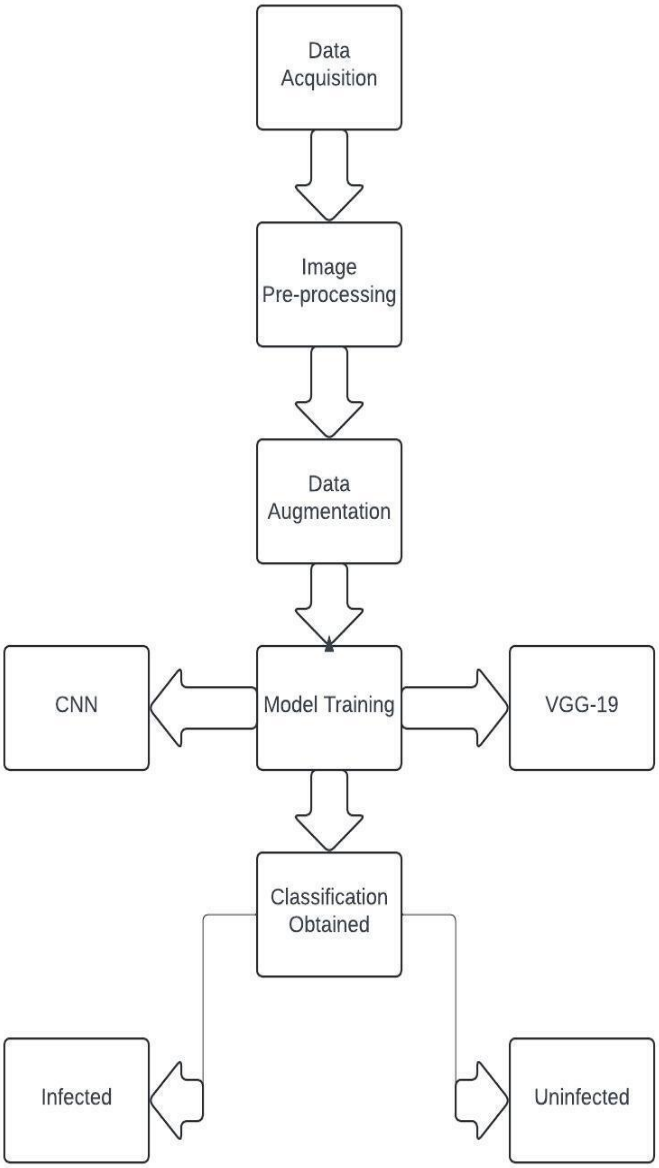


Figure 4 PROCESS DIAGRAM: PARASITE DETECTION

8.3 Parasite Classification

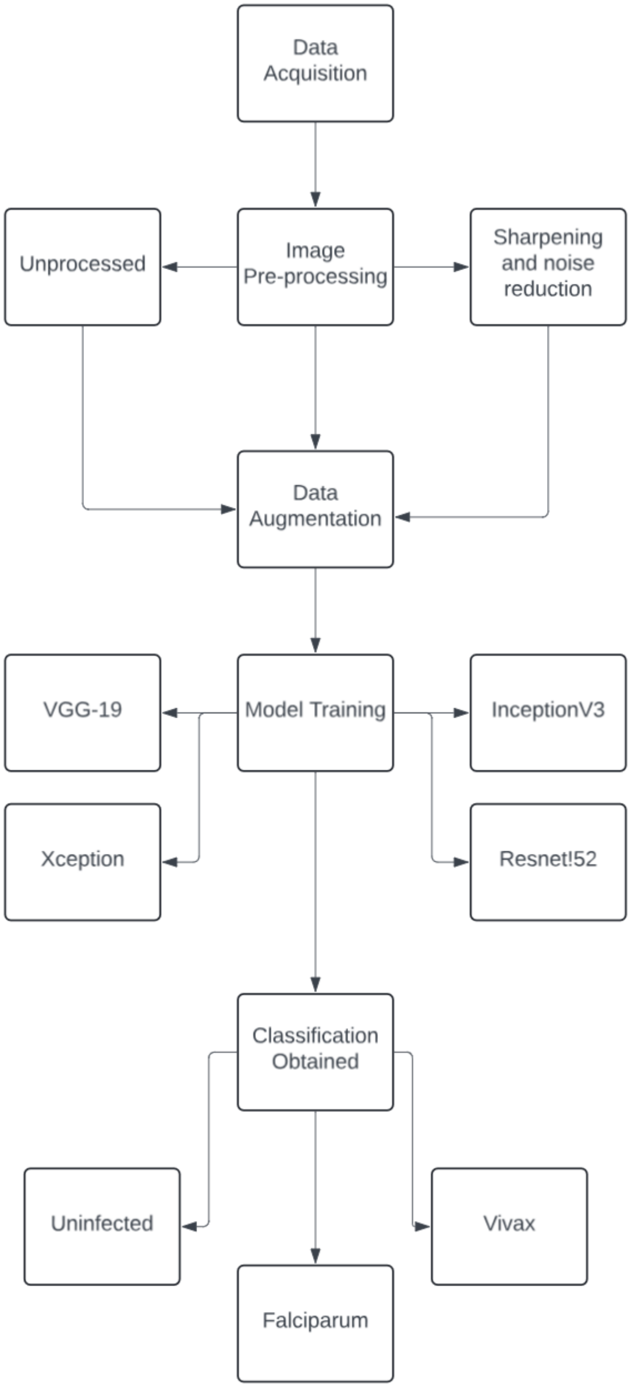


Figure 5 PROCESS DIAGRAM: PARASITE CLASSIFICATION

8.4 Hybrid Model for Feature Extraction with Ensemble Learning for Classification

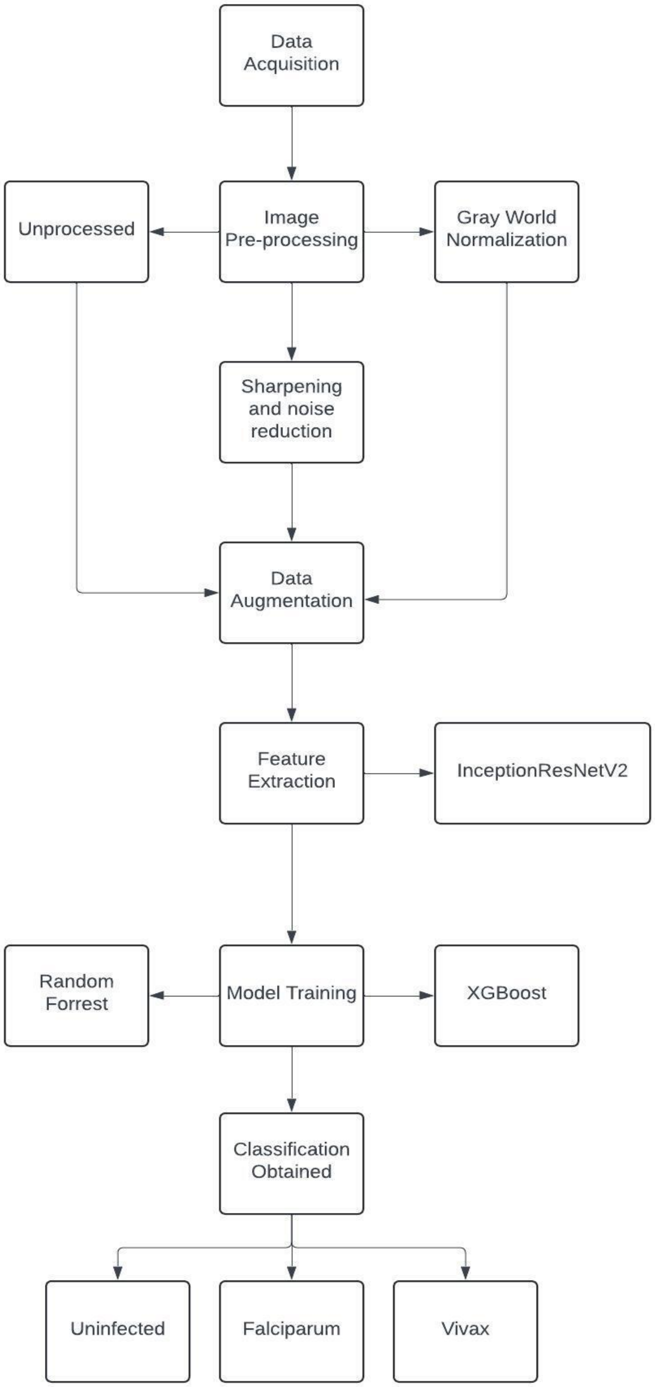


Figure 6 PROCESS DIAGRAM: HYBRID MODEL FOR FEATURE EXTRACTION AND ENSEMBLE LEARNING FOR CLASSIFICATION

Chapter 9

Results and Discussion

9.1 Parasite Detection

Preprocessing methods including noise reduction, sharpening, and histogram equalisation are essential for improving the effectiveness and precision of machine learning models in the field of parasite detection utilising thin blood smear pictures. The goal of noise reduction techniques is to remove undesired artefacts and abnormalities from images that may occur from a variety of sources, including environmental conditions or equipment restrictions. Noise reduction preprocessing makes sure the models can concentrate on retrieving pertinent information about the presence of parasites without being impacted by irrelevant distortions by using filters or algorithms meant to suppress noise while keeping significant features.

Furthermore, sharpening techniques are used to improve the definition and clarity of edges in the photos, which raises the general quality of the visual data that the models may use. This is especially important for thin blood smear images, where parasites might show up as minute differences in texture or as a stark contrast to the surrounding area. Sharpening these traits makes the models more capable of accurately identifying and categorising parasites. By improving the contrast and pixel intensity distribution throughout the photos, histogram equalisation helps with preprocessing even more and makes sure that crucial features are not lost in the shadows or light. When combined, these preprocessing methods help the machine learning models work more effectively by concentrating on extracting significant features that indicate the presence of parasites while reducing the influence of unimportant elements like noise or visual artefacts.

9.2 Image Processing for Parasite Detection

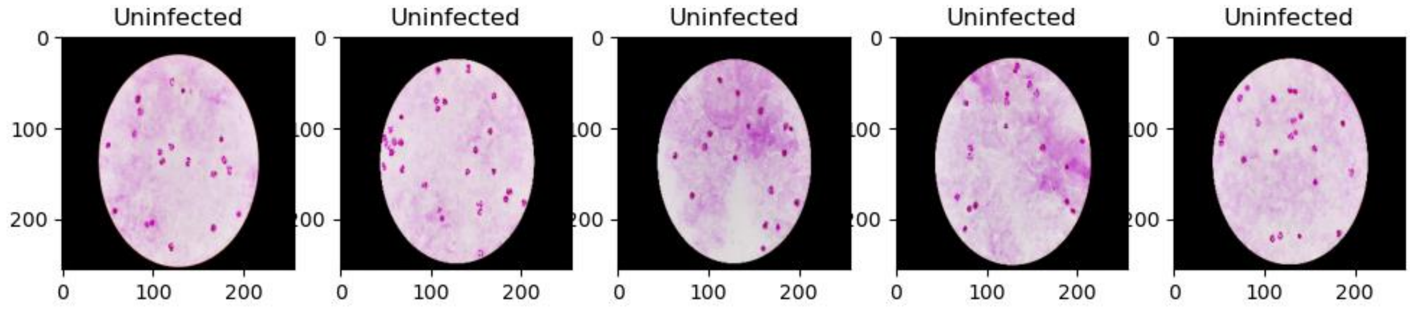


Figure 7 PARASITE DETECTION IMAGE DATA: BEFORE PREPROCESSING

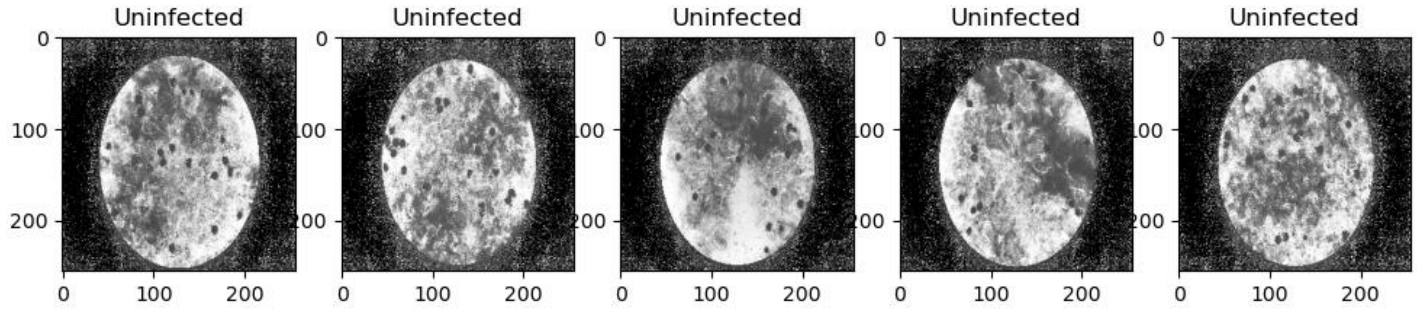


Figure 8 PARASITE DETECTION IMAGE DATA: AFTER PREPROCESSING

Deep learning models such as Convolutional Neural Networks (CNN) and VGG-19 architecture are used to train and test the preprocessed dataset for parasite detection. Thirty percent of the dataset is set aside for testing, and the remaining seventy percent is devoted to training. This division offers an impartial assessment of the models' performance on unseen samples and makes sure they are trained on enough data to identify relevant patterns. ImageDataGenerator is used to enable augmentation techniques and to streamline the management of the picture data. This adaptable application provides an easy way to add value to image datasets by applying various transformations, including flips, shifts, rotations, standardisation, and brightness modifications. These augmentation strategies increase the diversity of the dataset, which helps the models acquire robust features that are invariant to a variety of transformations frequently seen in real-world scenarios and improves their ability to generalise.

Additionally, by adding variability to the training data, ImageDataGenerator not only makes the process of data augmentation more efficient but also helps avoid overfitting. By ensuring that pixel values are scaled to a common range, standardisation approaches help to facilitate smoother convergence during the training of the model. By adding geometric alterations to the dataset through rotation, shifts, and flips, the models are able to learn invariant representations of parasites from various angles. By replicating fluctuations in lighting conditions, brightness changes significantly enrich the dataset and improve the models' adaptability to a variety of environmental circumstances. CNN and VGG-19 models can learn discriminative features for parasite detection and show resilience to changes in image look and quality by utilising these augmentation approaches in the training process.

9.3 Parasite Detection: CNN

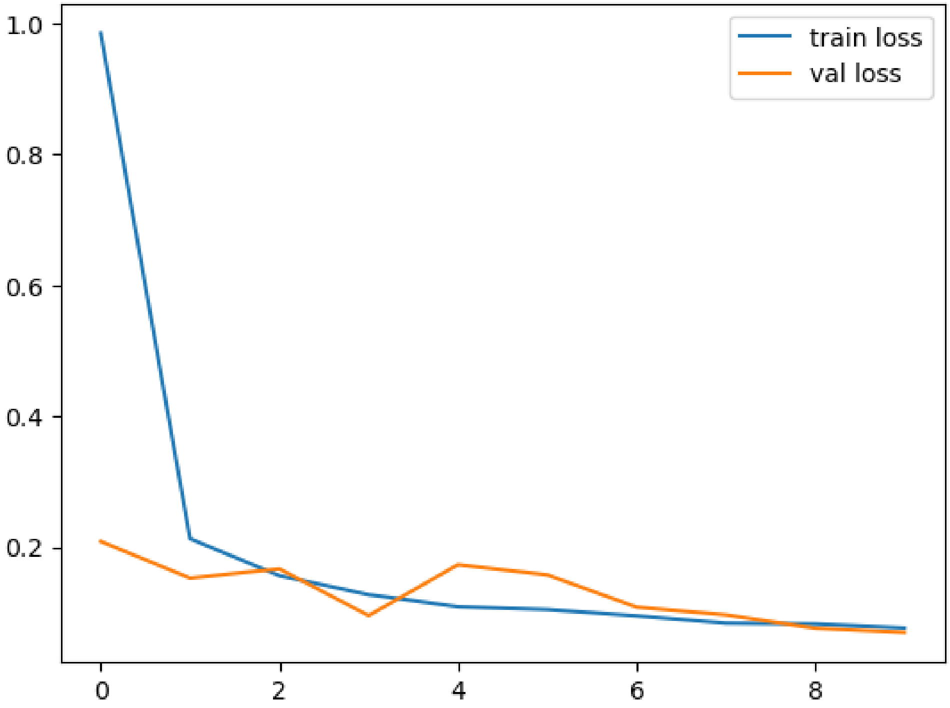


Figure 9 LOSS FUNCTION: CNN FOR PARASITE DETECTION

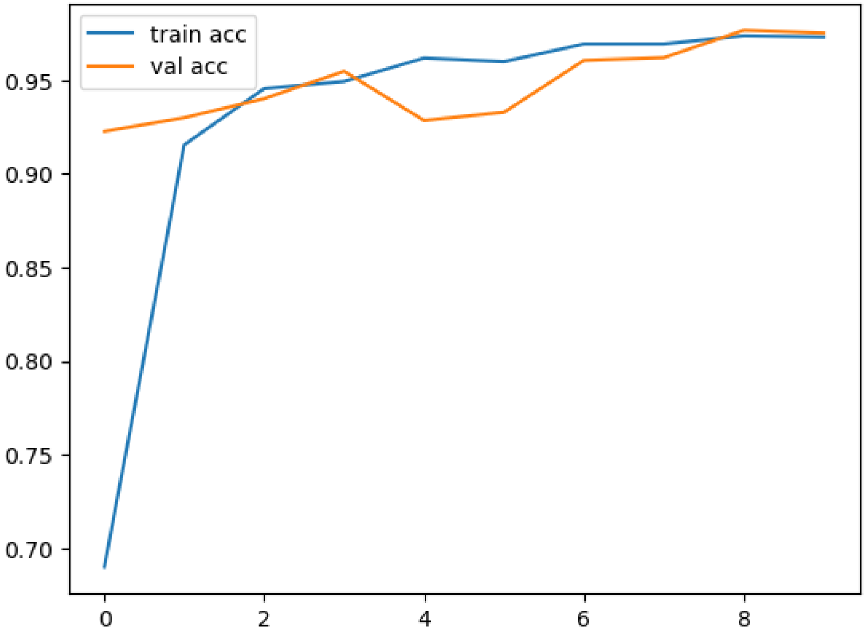


Figure 10 ACCURACY FUNCTION: CNN FOR PARASITE DETECTION

The output that is given illustrates how a Convolutional Neural Network (CNN) model is trained to detect malaria parasites. Ten training epochs are used to train the model, with updates shown at the end of each epoch. 'loss' and 'accuracy', or 'val_loss' and 'val_accuracy', represent the metrics that are tracked during training and validation, respectively, for both loss and accuracy.

On the training set, the model first achieves an accuracy of 0.69 and a reasonably high loss value of 0.9847 during the first epoch. This suggests that the accuracy of the model is low and that the predictions are far from the real labels. But as the training goes on, the precision gets better and the loss goes down progressively. As the training process comes to a close, the accuracy increases greatly to 0.9731 and the loss drastically drops to 0.0763, suggesting that the model has learned to make more accurate predictions. Similar trends are apparent on the validation set, which acts as an impartial gauge of the model's efficacy on unobserved data. Over the course of the epochs, the validation accuracy rises from 0.9226 to 0.9752, while the validation loss falls from 0.2084 to 0.0694. The model's steady increase in validation measures indicates that it is not just doing well on the training data but also generalising to new samples.

All things considered, the output shows how the CNN model was gradually enhanced through iterative training epochs, leading to better performance metrics on the training and validation sets. The model is learning to extract important features from the data and produce increasingly accurate predictions regarding the presence of malaria parasites in thin blood smear images, as seen by the falling loss values and increasing accuracy.

9.4 Parasite Detection: VGG-19

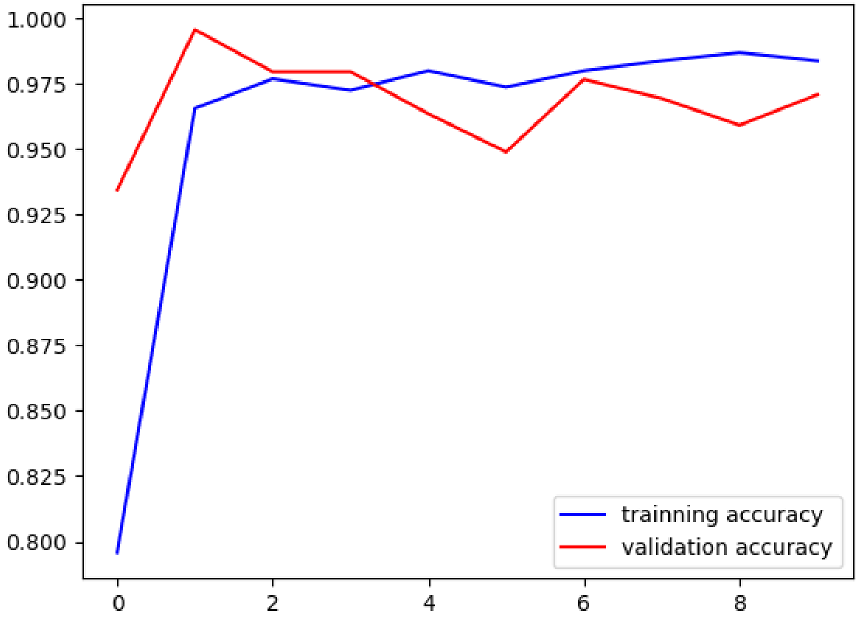


Figure 11 ACCURACY FUNCTION: VGG-19 FOR PARASITE DETECTION

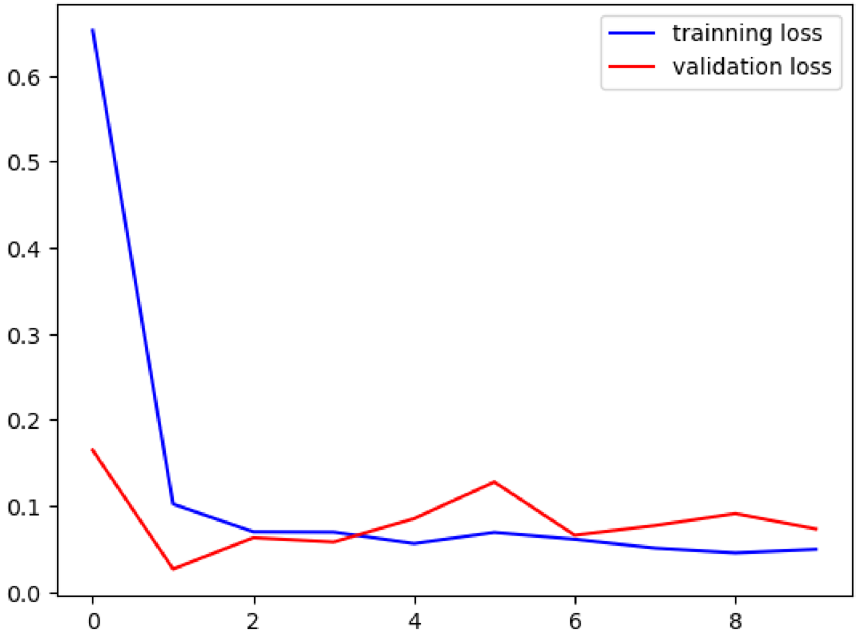


Figure 12 LOSS FUNCTION: VGG-19 FOR PARASITE DETECTION

The output that is given describes the steps involved in training a VGG-19 model over ten epochs to detect malaria parasites. The dataset is iterated through at each epoch, with updates displayed following each epoch. Both the loss and accuracy metrics are tracked during training for the validation data ('val_loss' and 'val_accuracy') as well as the training data ('loss' and 'accuracy').

Using the training data, the VGG-19 model achieves an accuracy of 0.7959 and a loss of 0.6528 in the first epoch. These metrics imply that the model's accuracy is moderate and that its predictions are somewhat off from the true labels. Performance does, however, significantly improve as training goes on. As training progresses, the accuracy rises noticeably to 0.9837 and the loss significantly drops to 0.0496). This suggests that throughout training, the model has improved its prediction accuracy while minimising loss.

On the validation set, where the model performs admirably, similar patterns are seen. Over the course of the epochs, the validation accuracy rises from 0.9343 to 0.9708, while the validation loss steadily falls from 0.1649 to 0.0734. These findings show that the VGG-19 model consistently improves validation measures, indicating that it not only learns from the training data but also generalises well to new data. Overall, the result demonstrates how the VGG-19 model was improved over time through repetitive training epochs, which improved performance in tasks involving the detection of malaria parasites.

9.5 Pre-processing for parasite classification

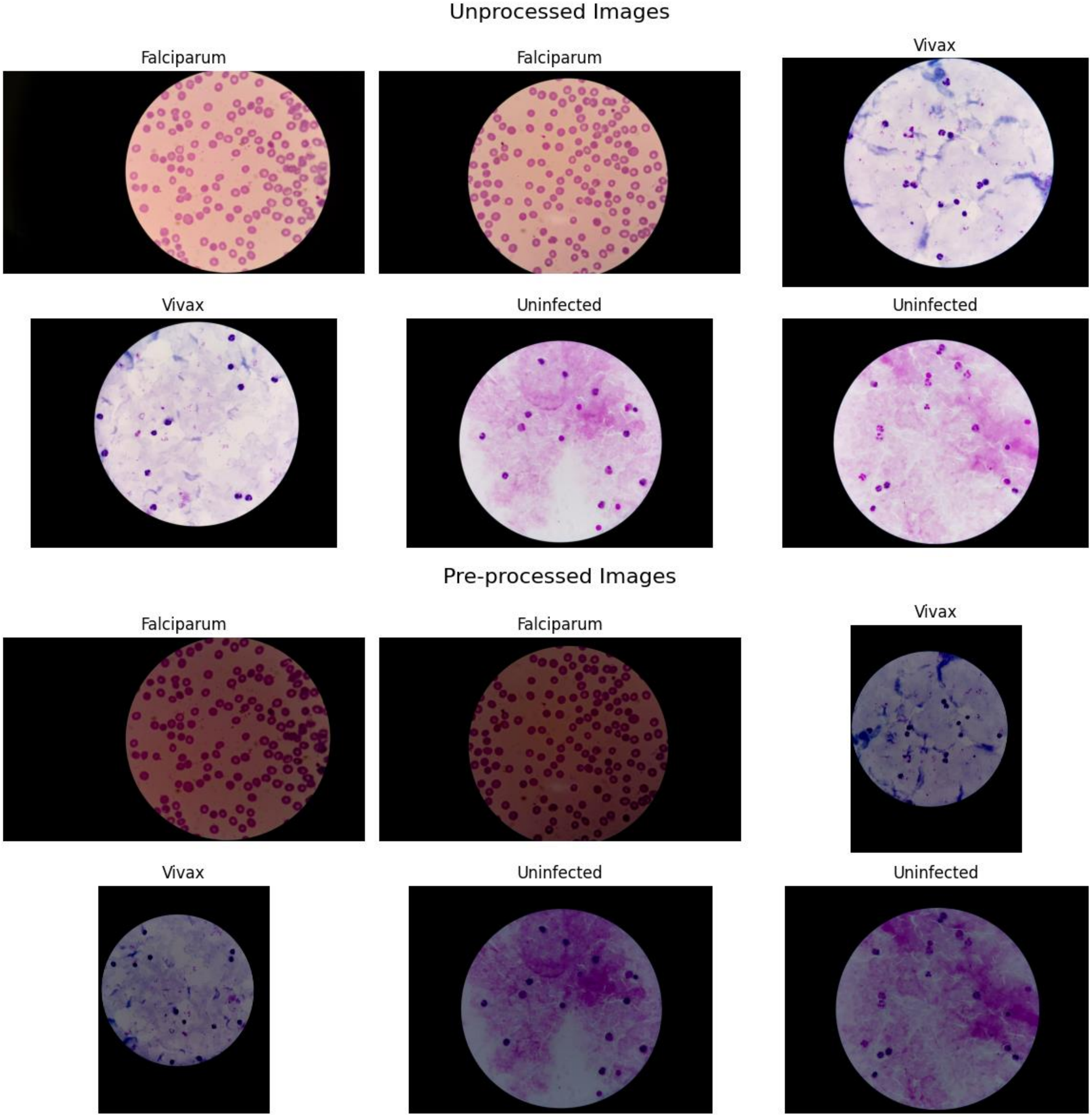


Figure 13 PREPROCESSING FOR PARASITE CLASSIFICATION

Numerous benefits arise for the identification and categorization of malaria parasites when sharpening and noise reduction techniques are used to blood smear pictures in malaria datasets. First of all, these methods improve the definition and clarity of important details in the pictures. Images from blood smears frequently provide minute features, such the morphology and form of the parasites, that are essential for precise classification. These traits become more noticeable when the photos are sharpened, which facilitates more accurate detection and classification of parasites by machine learning algorithms.

Numerous benefits arise for the identification and categorization of malaria parasites when sharpening and noise reduction techniques are used to blood smear pictures in malaria datasets. First of all, these methods improve the definition and clarity of important details in the pictures. Images from blood smears frequently provide minute features, such the morphology and form of the parasites, that are essential for precise classification. These traits become more noticeable when the photos are sharpened, which facilitates more accurate detection and classification of parasites by machine learning algorithms.

Additionally, the use of noise reduction and sharpening techniques strengthens the malaria detection models' resistance to dataset variability. The quality of the staining, the lighting, and other elements can vary greatly in blood smear photos. These methods lessen the effect of such unpredictability by improving feature clarity and cutting down on noise, guaranteeing that the models can correctly diagnose parasites in a variety of photos and environments. Overall, malaria detection models based on blood smear pictures are more accurate, robust, and reliable when sharpening and noise reduction approaches are integrated.

9.6 Parasite classification: VGG-19

9.6.1 Without Pre-processing

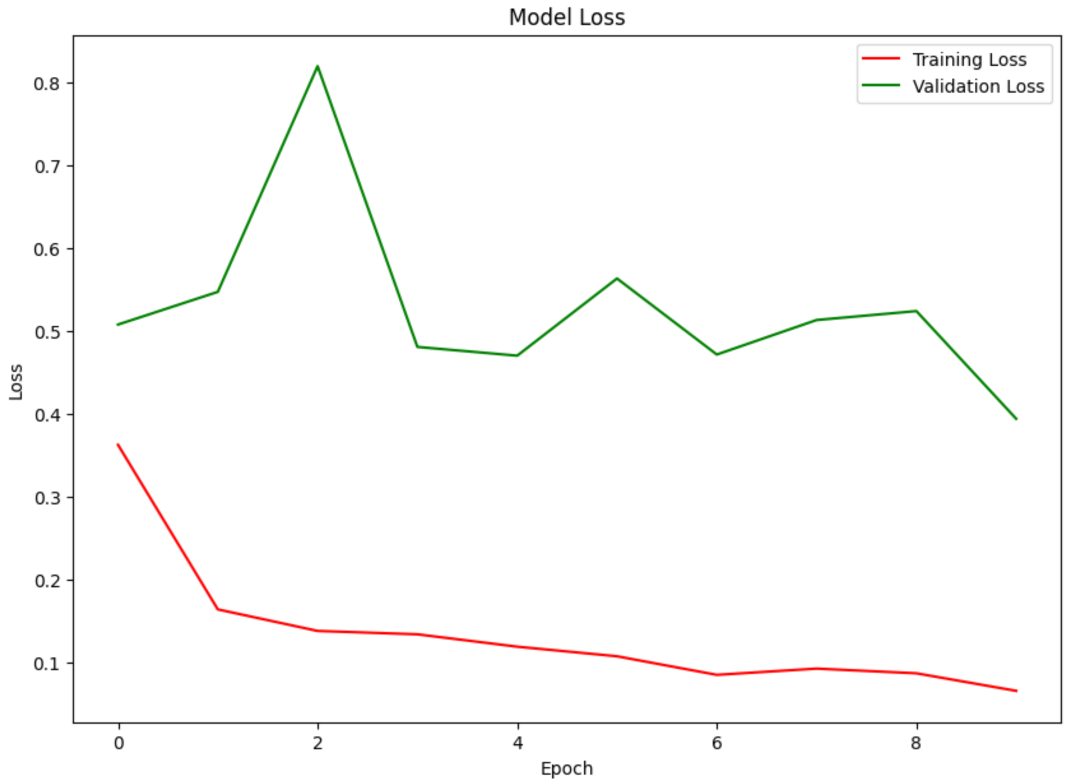


Figure 14 LOSS FUNCTION: VGG-19 FOR PARASITE CLASSIFICATION WITHOUT PREPROCESSING

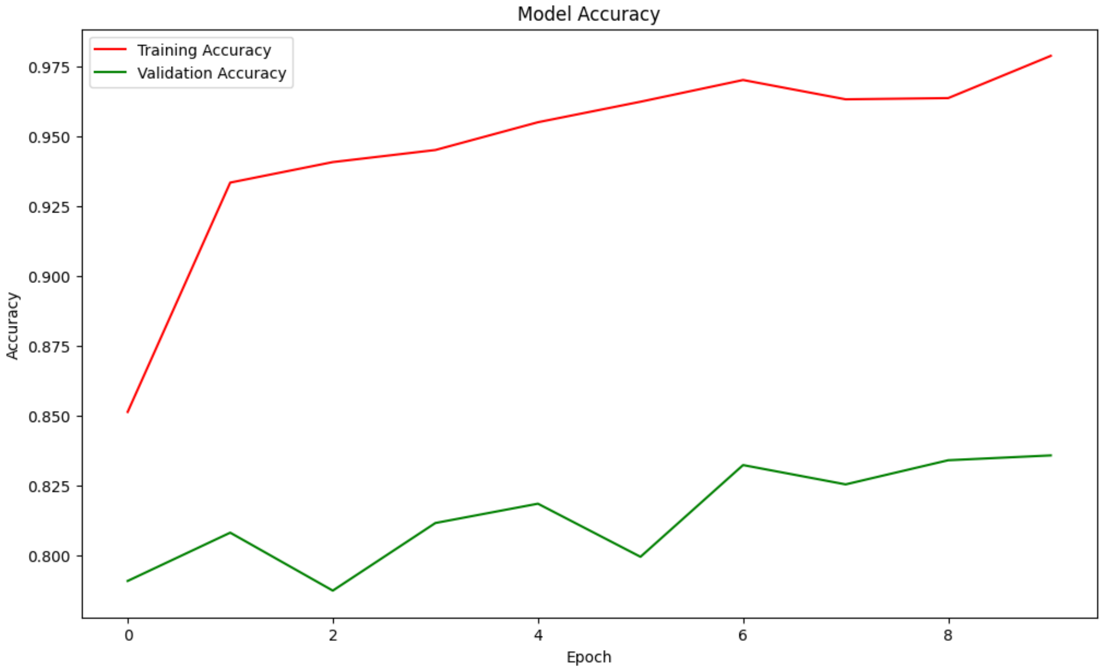


Figure 15 ACCURACY FUNCTION: VGG-19 FOR PARASITE CLASSIFICATION WITHOUT PREPROCESSING

The performance of a VGG-19 model trained to classify blood smear images into three categories—uninfected, falciparum malaria, and vivax malaria—is described in the training output that was provided. The model is trained iteratively over ten epochs, gaining the ability to identify patterns and features in the input images and use them to generate predictions. The model achieves a training accuracy of 85.15% with a corresponding loss of 0.3623 in the first epoch. These measures point to a mediocre performance level, indicating potential for development.

Performance data show a noticeable gain with training, especially precision. The model reaches a much higher training accuracy of 97.88% at the last epoch, proving its capacity to pick up complex properties that distinguish between the three classes. As training progresses, the associated loss significantly drops to 0.0652, suggesting that the model's predictions get more precise and closer to the genuine labels.

The model shows a similar trend of improvement across epochs on the validation set. Despite beginning at 79.10% in the first epoch, the validation accuracy climbs progressively, reaching 83.59% by the end of the epoch. This suggests that the model's ability to consistently improve accuracy on the validation set is indicative of its good generalisation to new data. Over epochs, the validation loss—a measure of the difference between expected and actual values—also diminishes, providing more evidence of the model's efficiency in gathering pertinent data for classification tasks.

Overall, it appears from the training output that the VGG-19 model can successfully categorise blood smear images into groups that correspond to vivax, falciparum, and uninfected malaria. The model successfully learns discriminative features and patterns inherent in the dataset, making it a potential tool for malaria parasite identification and classification, as evidenced by the large increase in accuracy and decrease in loss measures over epochs. However, in order to properly evaluate the model's performance and generalisation abilities, additional testing and validation on untested datasets are required.

9.6.2 With Pre-processing

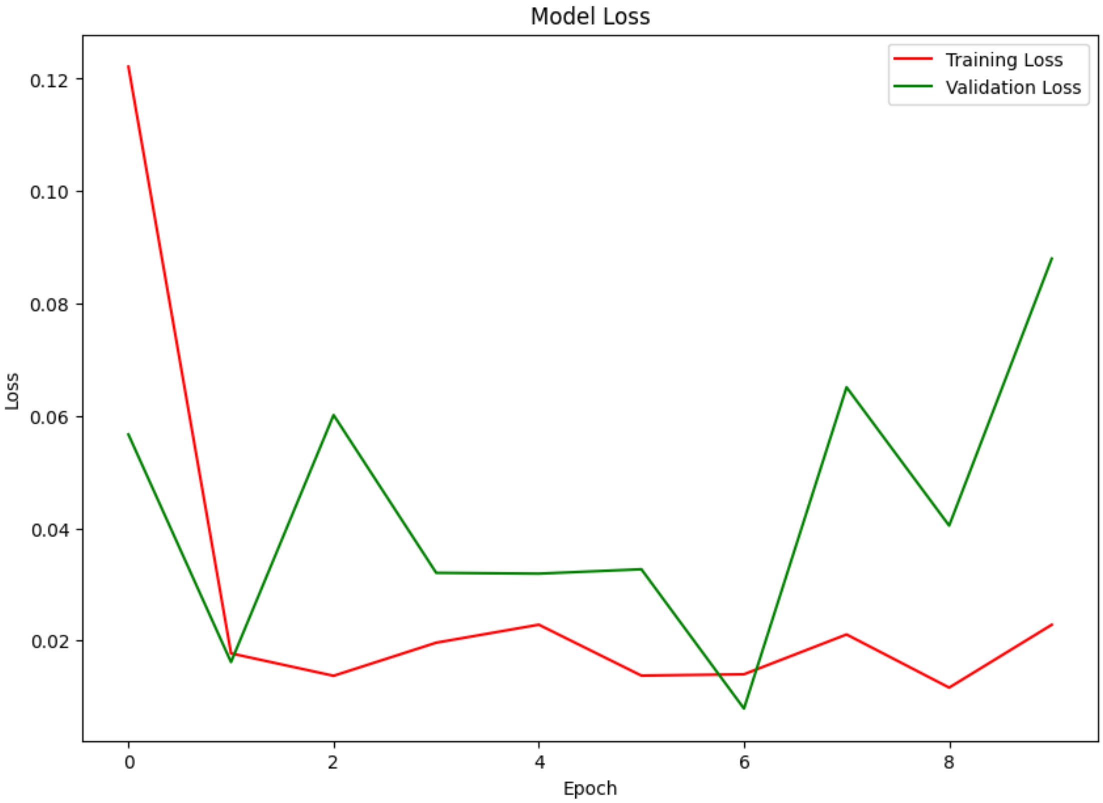


Figure 16 LOSS FUNCTION: VGG-19 FOR PARASITE CLASSIFICATION WITH PREPROCESSING

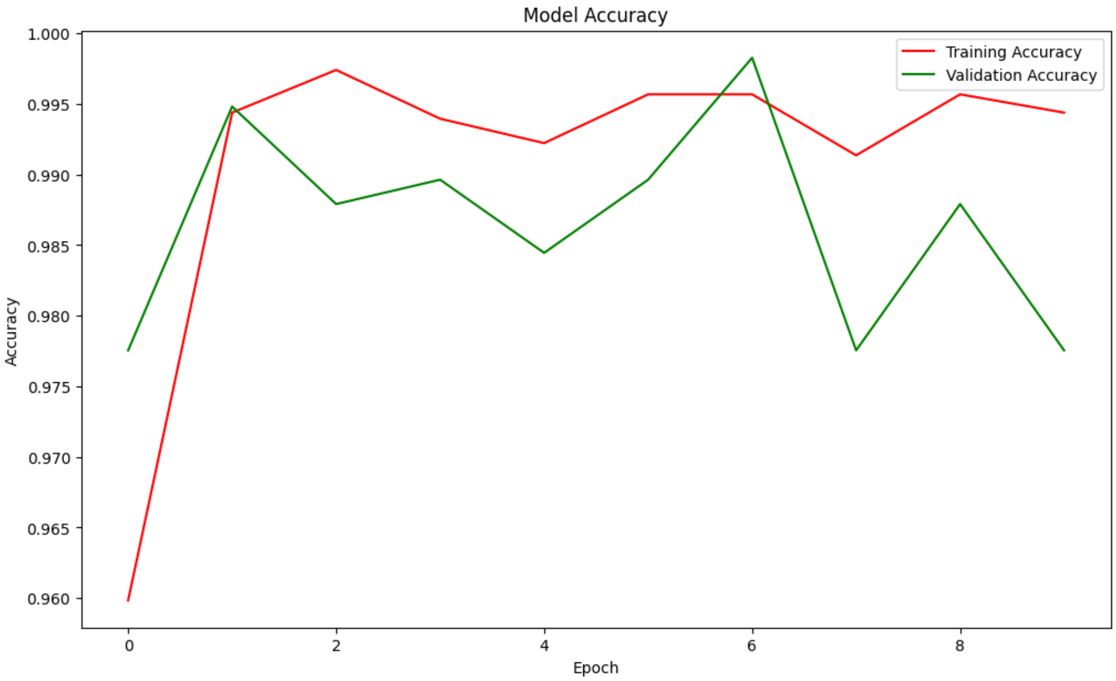


Figure 17 ACCURACY FUNCTION: VGG-19 FOR PARASITE CLASSIFICATION WITH PREPROCESSING

The performance of a VGG-19 model trained to categorise blood smear images into three groups—uninfected, falciparum malaria, and vivax malaria—is demonstrated in the training output that has been supplied. The model goes through iterative training over ten epochs, learning to extract pertinent features from the input photos in order to produce precise predictions. The model shows a good beginning performance level with a training accuracy of 95.98% and a corresponding loss of 0.1222 in the first epoch.
Performance data show a significant gain with training, especially precision. The model reaches a higher training accuracy of 99.44% by the last epoch, proving that it can pick up complex features that distinguish between the three classes. As training progresses, the associated loss significantly drops to 0.0228, showing that the model's predictions get more precise and closer to the genuine labels.

The model shows a similar trend of improvement across epochs on the validation set. The validation accuracy maintains a high level during training, reaching 97.75% by the last epoch, while starting at a high level of 97.75% in the first epoch. The model's ability to attain constant accuracy on the validation set suggests that it generalises effectively to new data. The validation loss, a metric that gauges the difference between expected and actual values, also stays comparatively low during training, providing more evidence of the model's ability to accurately capture pertinent data for classification tasks.

Overall, it appears from the training output that the VGG-19 model can successfully categorise blood smear images into groups that correspond to vivax, falciparum, and uninfected malaria. The model successfully learns discriminative features and patterns inherent in the dataset, making it a potential tool for malaria parasite identification and classification, as evidenced by the large increase in accuracy and decrease in loss measures over epochs. To completely evaluate the model's performance and generalisation skills in real-world circumstances, more testing on untested datasets could be required.

9.7 Parasite Classification: InceptionV3

9.7.1 Without pre-processing

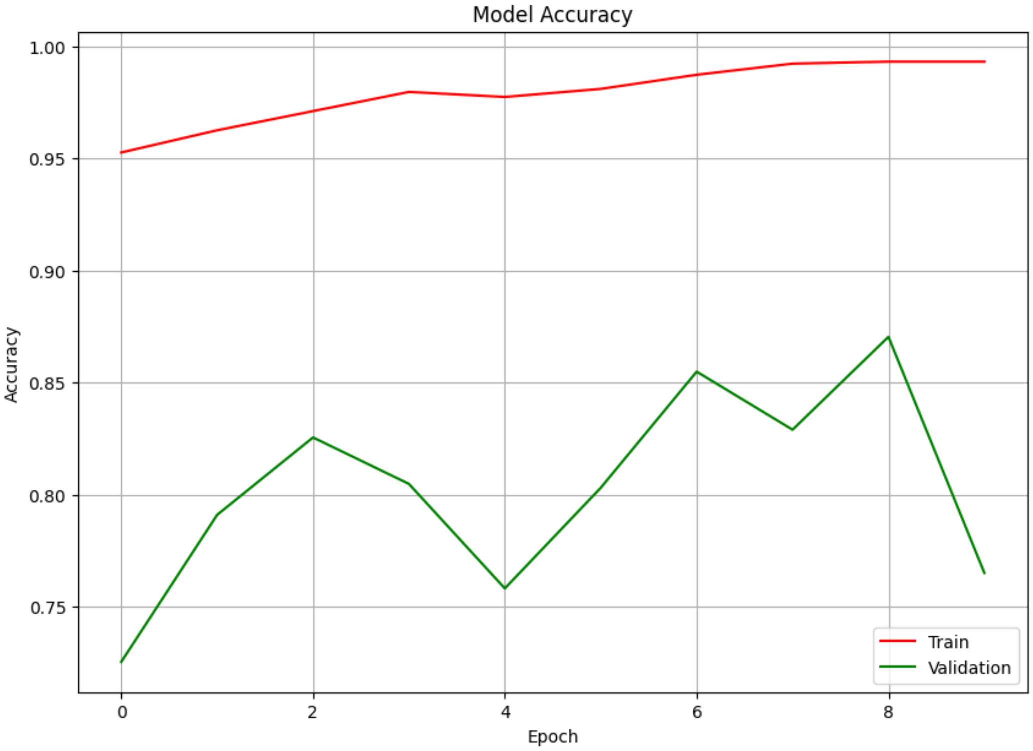


Figure 18 ACCURACY FUNCTION: INCEPTIONV3 FOR PARASITE CLASSIFICATION WITHOUT PROPROCESSING

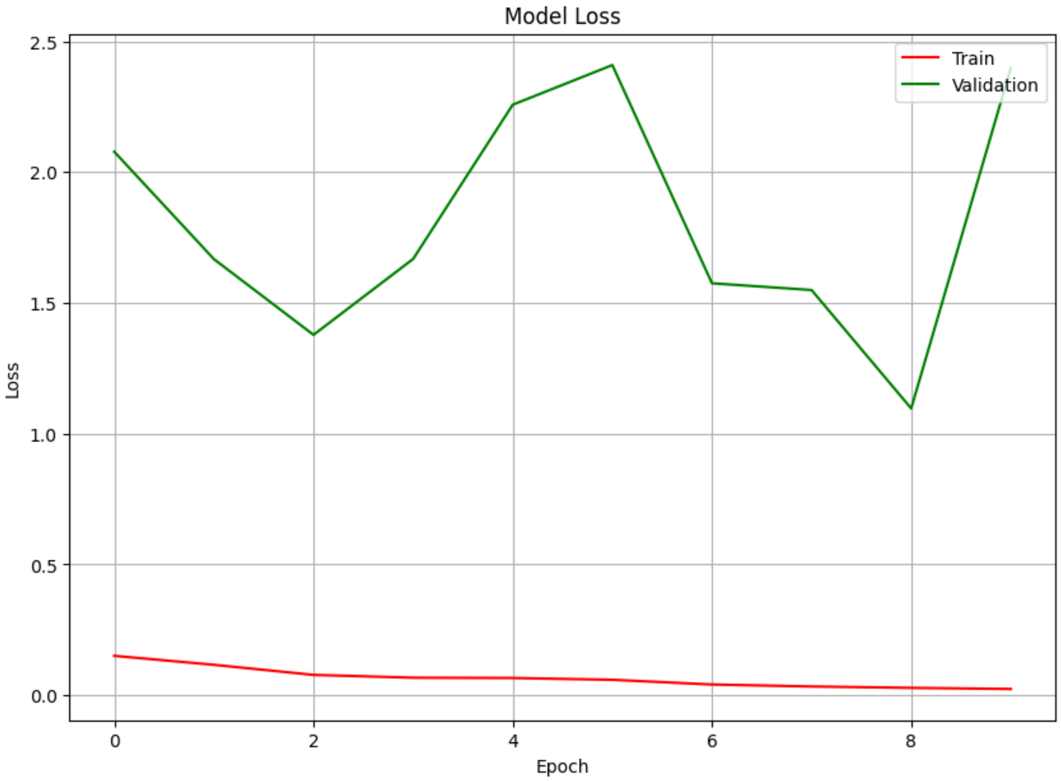


Figure 19 LOSS FUNCTION: INCEPTIONV3 FOR PARASITE CLASSIFICATION WITHOUT PROPROCESSING

The performance of an InceptionV3 model trained for 10 epochs to classify malaria data into three classes—uninfected, falciparum, and vivax—is described in detail in the training log that is provided. The model has good performance metrics, showing steady increases in training and validation accuracies throughout the course of the epochs, suggesting efficient learning.
The model starts out promisingly, with a training accuracy of roughly 87.7% and a validation accuracy of roughly 71.3% in the first epoch. The training accuracy reaches 99.1% in the subsequent epochs, whereas the validation accuracy peaks at 84.3% in the last epoch, indicating a significant improvement. This increasing trend indicates that the model is doing a good job at generalising to samples that have not yet been observed and capturing the underlying patterns in the data.

Over the course of the epochs, the loss function—which quantifies the difference between anticipated and true labels—slowly declines, suggesting that the model is getting better at making predictions. On the other hand, variations in the validation loss over epochs point to either overfitting or difficulties in generalising to new data. The model design might be adjusted, or regularisation strategies could be used to handle these variations.

All things considered, the offered InceptionV3 model performs admirably when it comes to categorising malaria data into several classifications. To further comprehend its decision-making process, model interpretability methodologies, performance evaluation on test data that hasn't been seen yet, and strategies to enhance generalisation performance could all be part of future investigation. Furthermore, before implementing the model in actual clinical settings, thorough validation and testing protocols would be necessary due to the nature of medical data.

9.7.2 With pre-processing

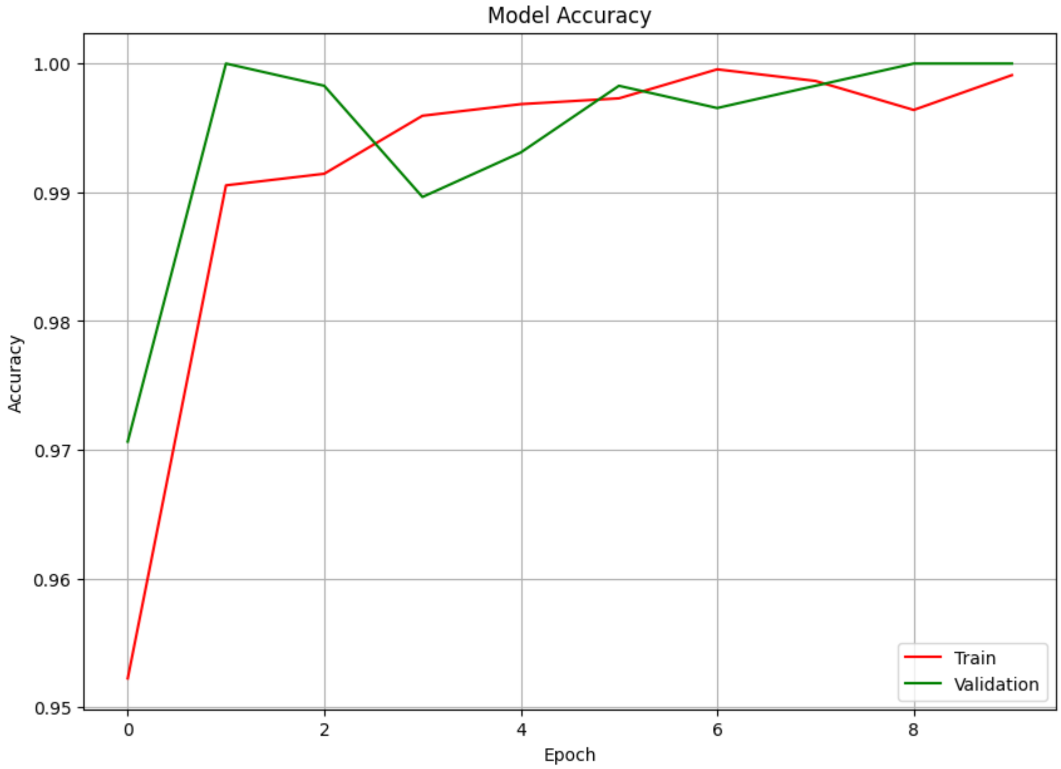


Figure 20 ACCURACY FUNCTION: INCEPTIONV3 FOR PARASITE CLASSIFICATION WITH PROPROCESSING

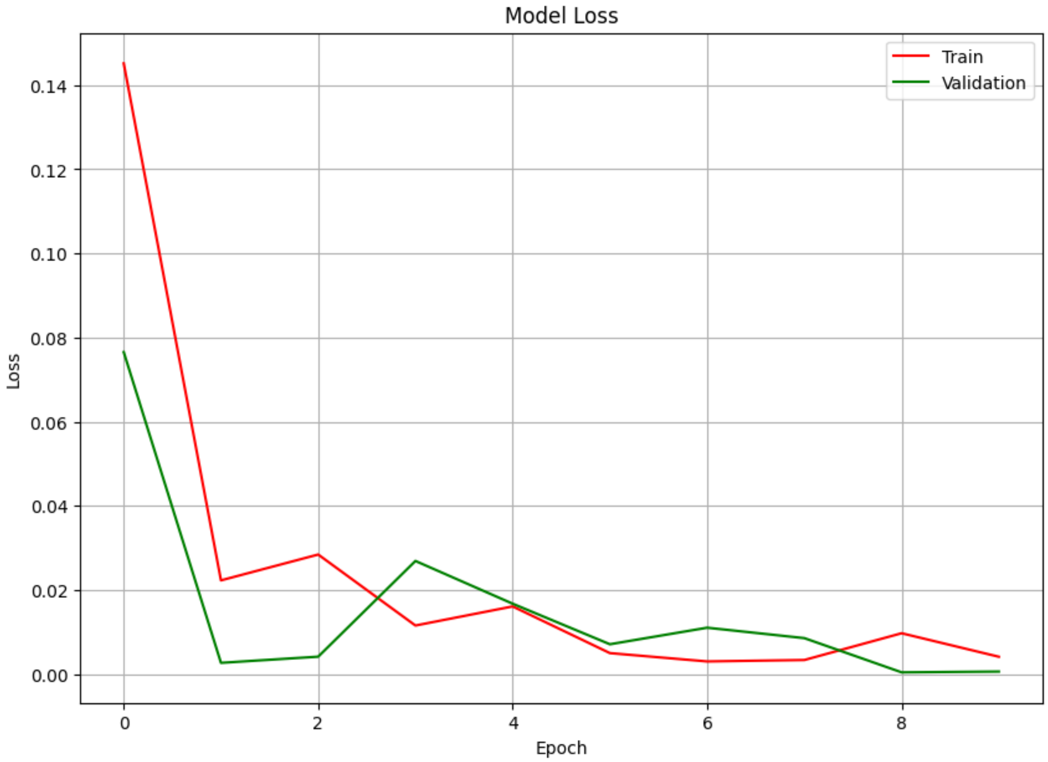


Figure 21 LOSSFUNCTION: INCEPTIONV3 FOR PARASITE CLASSIFICATION WITH PROPROCESSING

The performance of an InceptionV3 model trained to classify blood smear images into three categories—uninfected, falciparum malaria, and vivax malaria—is displayed in the training output that has been provided. The model is trained iteratively over ten epochs, gaining the ability to identify patterns and features in the input images and use them to generate predictions. The model demonstrates a great initial performance level in the first epoch, achieving a training accuracy of 95.23% with a corresponding loss of 0.1452.

Performance indicators show significant increase with training, especially with precision. The model reaches a higher training accuracy of 99.91% at the last epoch, proving that it can pick up complex features that distinguish between the three classes. As training progresses, the associated loss significantly drops to 0.0041, suggesting that the model's predictions get more precise and closer to the genuine labels.

The model shows a similar trend of improvement across epochs on the validation set. The validation accuracy maintains a high level during training, reaching 100.00% by the last epoch, while starting at a high 97.06% in the first epoch. The model's ability to attain constant accuracy on the validation set suggests that it generalises effectively to new data. Throughout training, the validation loss—a measure of the difference between expected and actual values—also stays very low, providing more evidence of the model's efficacy in gathering pertinent data for classification tasks.

The InceptionV3 model appears to be quite successful at classifying blood smear images into categories for vivax, falciparum, and uninfected malaria, based on the training output that has been provided. A strong tool for the detection and classification of malaria parasites, the model effectively learns discriminative features and patterns present in the dataset, as evidenced by the notable increase in accuracy and decrease in loss measures throughout epochs. To completely evaluate the model's performance and generalisation skills in real-world circumstances, more testing on untested datasets could be required.

9.8 Parasite Classification: Xception

9.8.1 Without Pre-processing

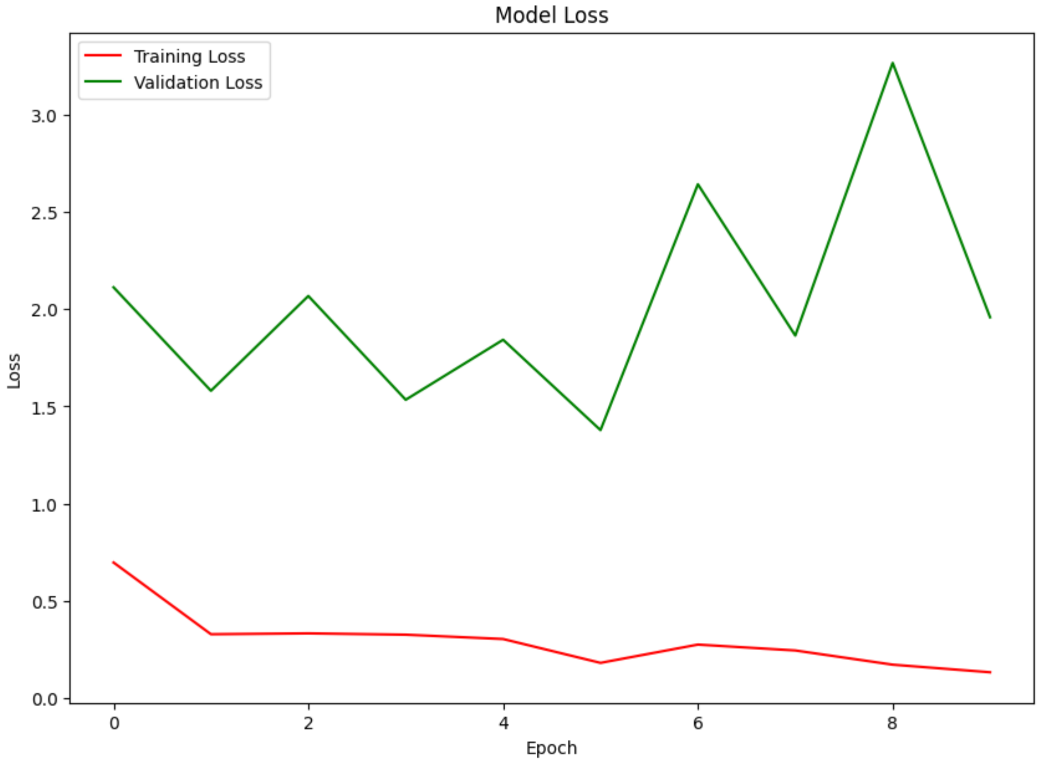


Figure 22 LOSS FUNCTION: XCCETION FOR PARASITE CLASSIFICATION WITHOUT PREPROCESSING

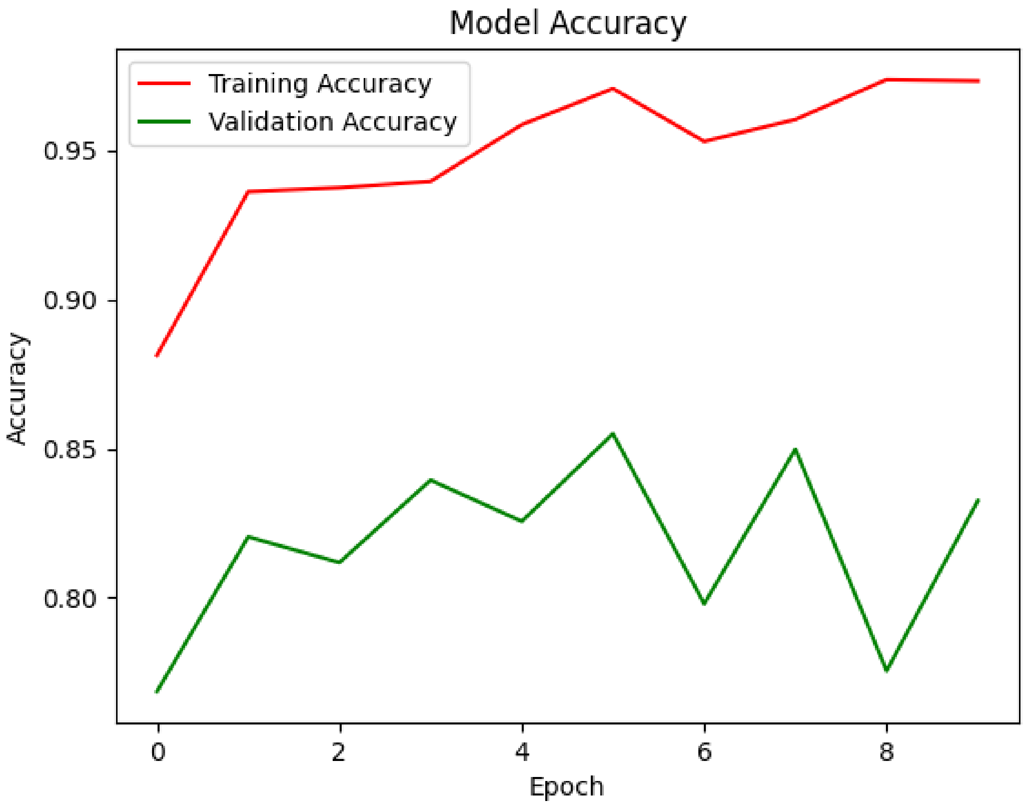


Figure 23 ACCURACY FUNCTION: XCCETION FOR PARASITE CLASSIFICATION WITHOUT PREPROCESSING

An Xception model trained to classify blood smear images into three categories—uninfected, falciparum malaria, and vivax malaria—performs as demonstrated by the training output that is supplied. The model is trained iteratively over ten epochs, extracting characteristics and patterns from the input images to enable precise prediction-making. The model shows a moderate beginning performance level in the first epoch, achieving a training accuracy of 88.13% with a corresponding loss of 0.6974.

The performance measures fluctuate as training goes on, with some epochs displaying modest losses and others showing improvements. The model reaches a higher training accuracy of 97.32% by the last epoch, proving that it can pick up complex features that distinguish between the three classes. At 0.1326, the training loss is still somewhat high, suggesting that there may be more space for improvement in terms of reducing prediction mistakes.

The model shows different epochs of accuracy on the validation set. During training, the validation accuracy varies, starting at a mild 76.86% in the first epoch and reaching 83.25% by the end. This suggests that while there may be some variation in the model's performance across various validation datasets or epochs, it does show some generalisation to previously unexplored data. Throughout training, the validation loss—a measure of the difference between expected and actual values—also changes, indicating how well the model is able to extract pertinent data for classification tasks.

The Xception model appears to have potential in categorising blood smear images into categories for vivax, falciparum, and uninfected malaria, based on the training output provided. To enhance consistency and performance across epochs and to solve any possible overfitting or underfitting problems, more research and optimisation would be required. To properly evaluate the model's efficacy and generalisation potential in real-world circumstances, additional testing on untested datasets and comparison with alternative models could be required.

9.8.2 With pre-processing

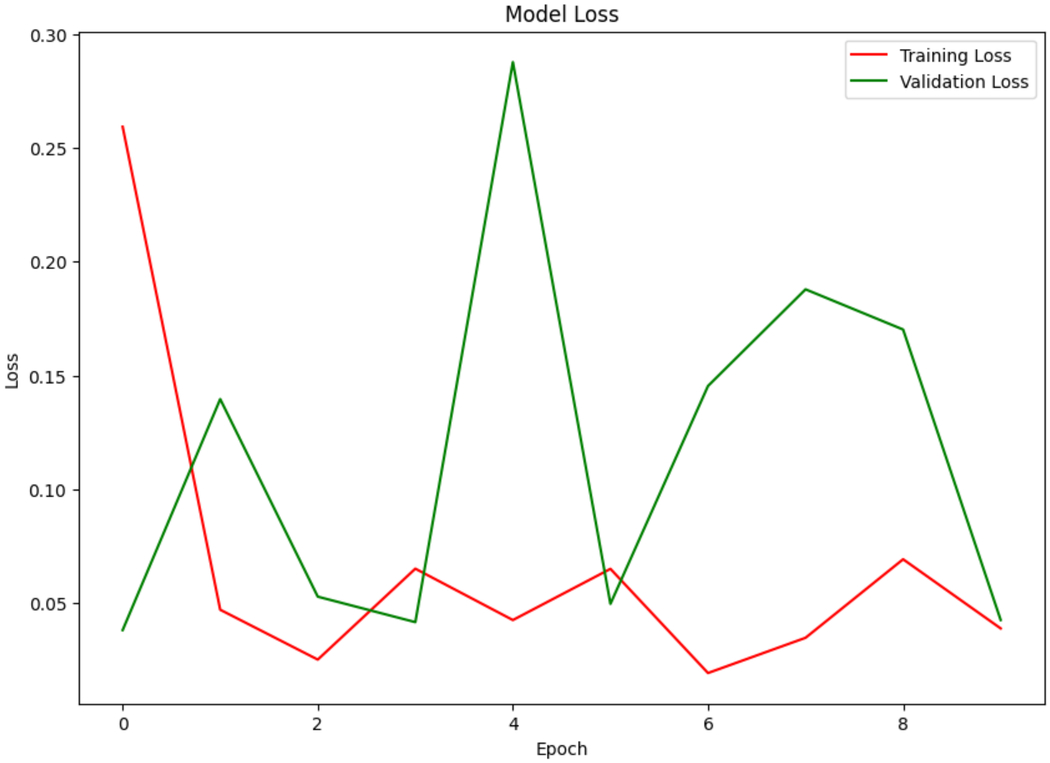


Figure 24 LOSS FUNCTION: XCCETION FOR PARASITE CLASSIFICATION WITH PREPROCESSING

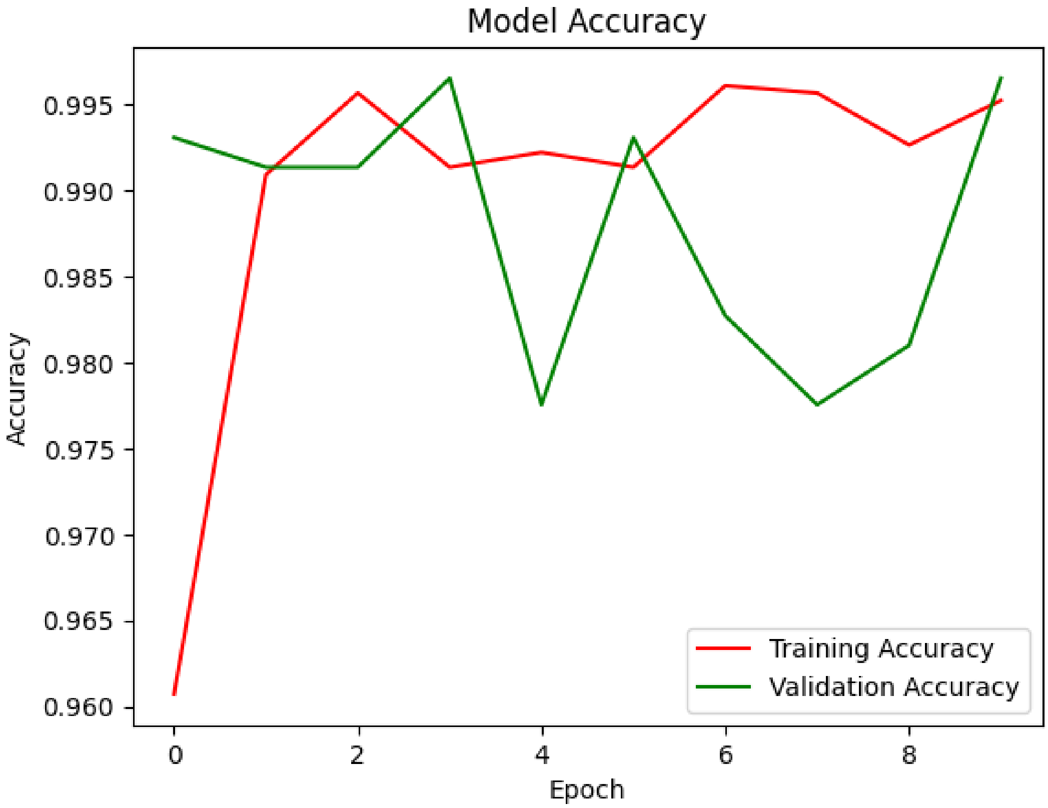


Figure 25 ACCURACY FUNCTION: XCCETION FOR PARASITE CLASSIFICATION WITH PREPROCESSING

The performance of an Xception model trained to categorise blood smear images into three groups—uninfected, falciparum malaria, and vivax malaria—is demonstrated in the training output that has been supplied. The model is trained iteratively over ten epochs, gaining the ability to identify patterns and features in the input images and use them to provide precise predictions. The model shows a great initial performance level in the first epoch, achieving a training accuracy of 96.07% with a corresponding loss of 0.2592.

There are variations in the performance measures between epochs as training advances. While some epochs indicate small reductions, others show improvements. By the end of the period, the model has successfully learned to differentiate between the three classes, as evidenced by its higher training accuracy of 99.53%. As training progresses, the associated loss significantly drops to 0.0389, indicating that the model's predictions get more precise and closer to the genuine labels.

The model consistently shows high accuracy across epochs on the validation set. Throughout training, the validation accuracy maintains a high degree of accuracy, starting at 99.31% in the first epoch and reaching 99.65% by the end of the training period. The model's ability to attain constant accuracy on the validation set suggests that it generalises effectively to new data. Throughout training, the validation loss—a measure of the difference between expected and actual values—remains very small, indicating that the model is successful in gathering pertinent data for classification tasks.

The training output supplied indicates that the Xception model is capable of accurately classifying blood smear images into three categories: vivax, falciparum, and uninfected. A strong tool for the detection and classification of malaria parasites, the model effectively learns discriminative features and patterns present in the dataset, as evidenced by the notable increase in accuracy and decrease in loss measures throughout epochs. To completely evaluate the model's performance and generalisation skills in real-world circumstances, more testing on untested datasets could be required.

9.9 Parasite Classification: ResNet152

9.9.1 Without pre-processing

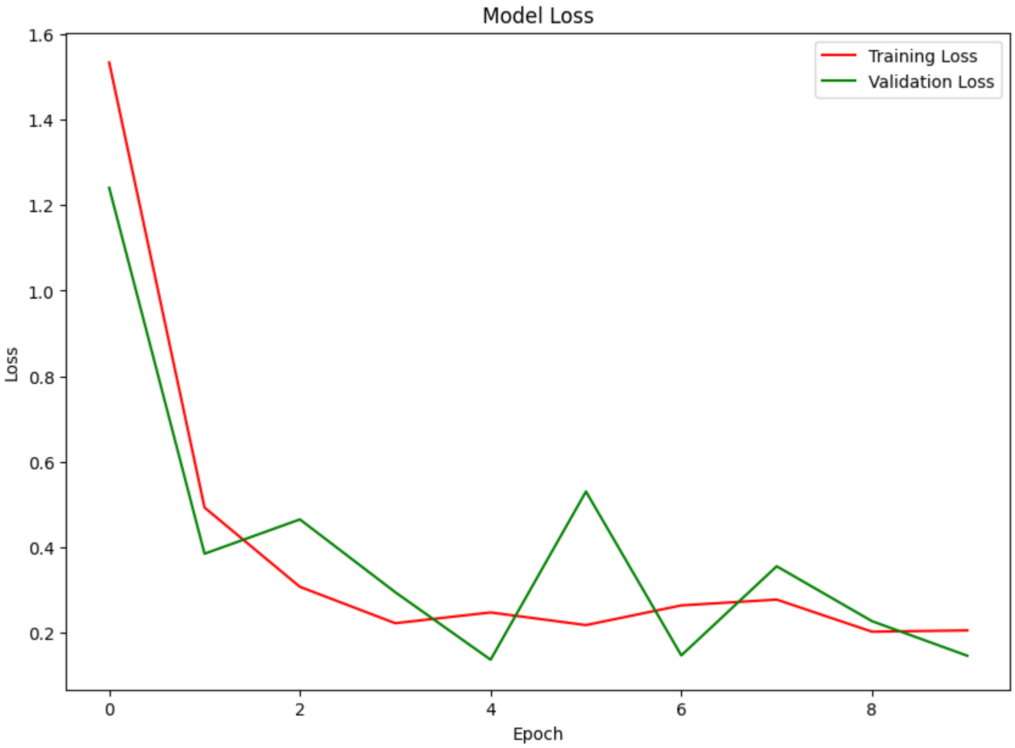


Figure 26 LOSS FUNCTION: RESNET152 FOR PARASITE CLASSIFICATION WITHOUT PREPROCESSING

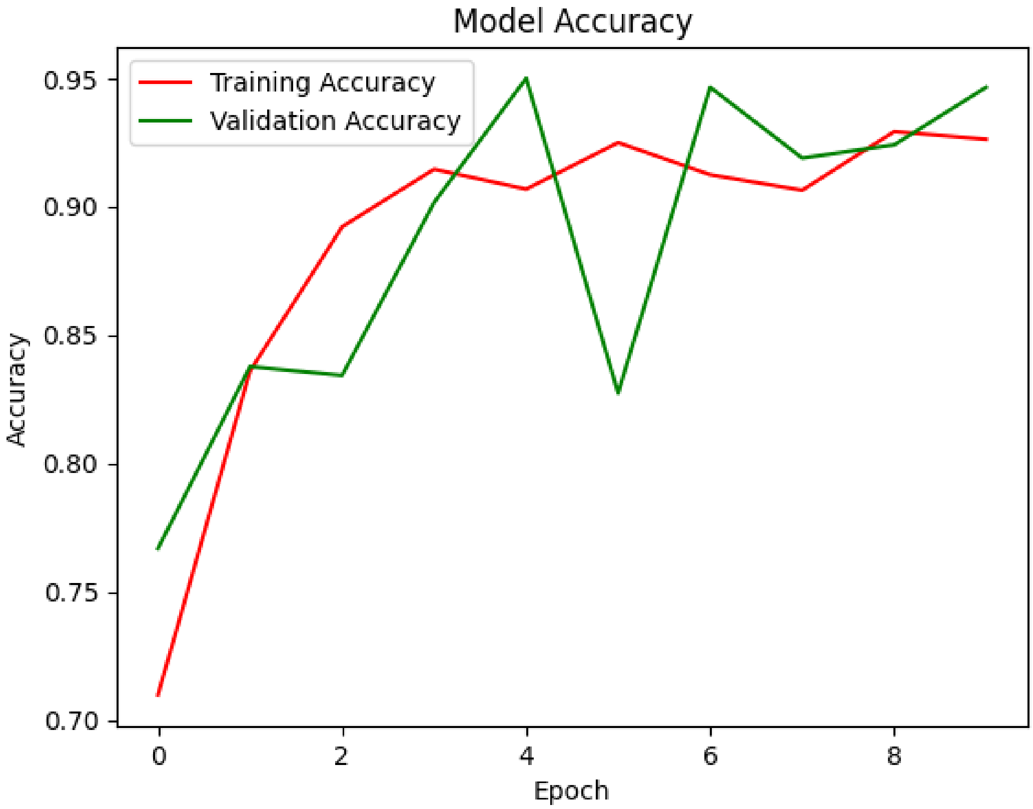


Figure 27 ACCURACY FUNCTION: RESNET152 FOR PARASITE CLASSIFICATION WITHOUT PREPROCESSING

The performance of a ResNet152 model trained to categorise blood smear images into three groups—uninfected, falciparum malaria, and vivax malaria—is demonstrated by the training output that is presented. The model is trained iteratively over ten epochs, gaining the ability to identify patterns and features in the input images and use them to provide precise predictions. The model shows a moderate beginning performance level in the first epoch, achieving a training accuracy of 70.98% with a corresponding loss of 1.5331.

There are variations in the performance measures between epochs as training advances. While some epochs indicate small reductions, others show improvements. The model has a greater training accuracy of 92.62% by the final epoch, suggesting that it has successfully acquired the ability to discriminate between the three classes. As training progresses, the associated loss significantly drops to 0.2059, indicating that the model's predictions get more precise and closer to the genuine labels.

The model consistently shows high accuracy across epochs on the validation set. Throughout training, the validation accuracy maintains a high degree of accuracy, starting at a mild 76.68% in the first epoch and reaching 94.65% by the end of the training period. The model's ability to attain constant accuracy on the validation set suggests that it generalises effectively to new data. Throughout training, the validation loss—a measure of the difference between expected and actual values—remains very small, indicating that the model is successful in gathering pertinent data for classification tasks.

The ResNet152 model appears to be very successful at classifying blood smear images into categories for vivax, falciparum, and uninfected malaria, based on the training output that was provided. A strong tool for the detection and classification of malaria parasites, the model effectively learns discriminative features and patterns present in the dataset, as evidenced by the notable increase in accuracy and decrease in loss measures throughout epochs. To completely evaluate the model's performance and generalisation skills in real-world circumstances, more testing on untested datasets could be required.

9.9.2 With pre-processing

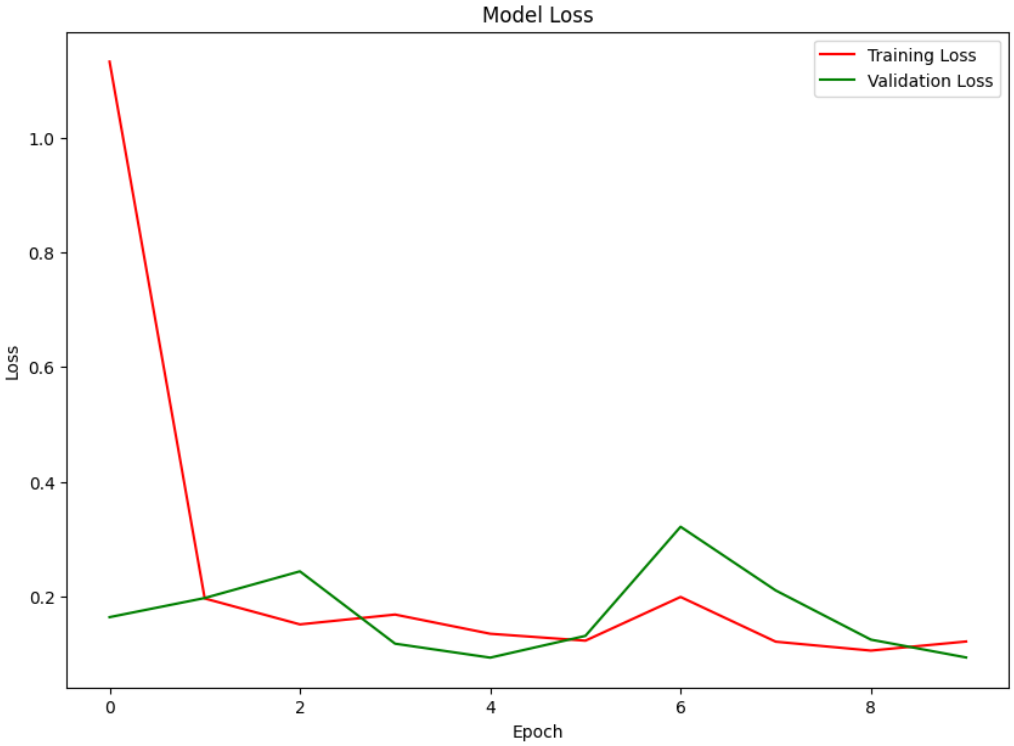


Figure 28 LOSS FUNCTION: RESNET152 FOR PARASITE CLASSIFICATION WITH PREPROCESSING

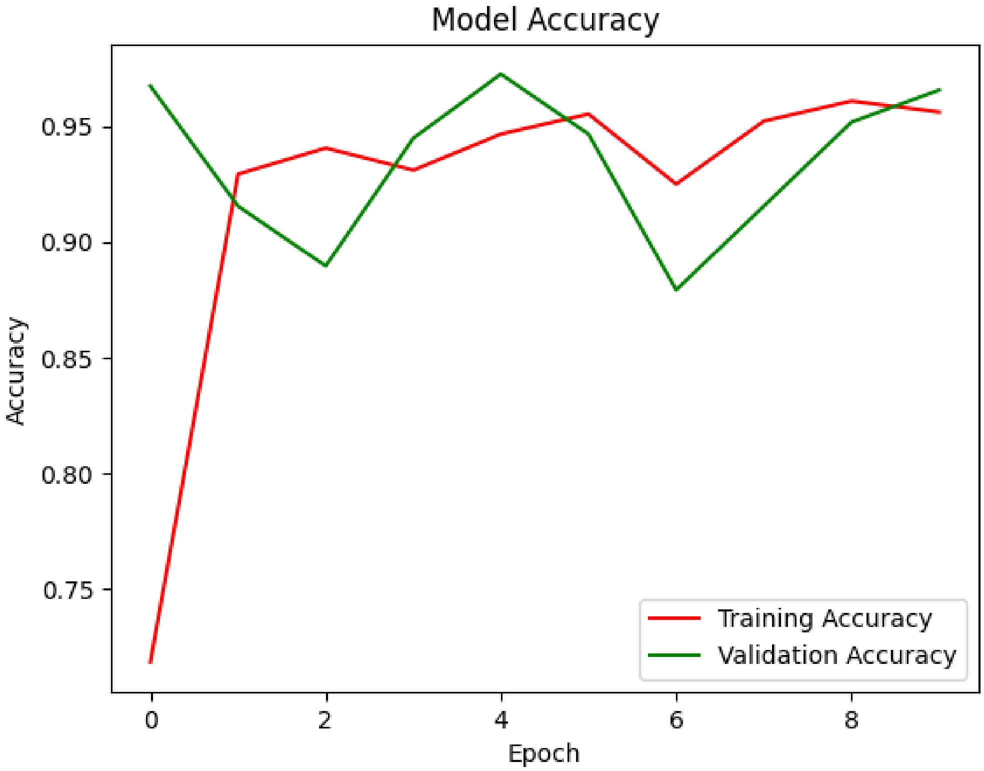


Figure 29 ACCURACY FUNCTION: RESNET152 FOR PARASITE CLASSIFICATION WITH PREPROCESSING

The performance of a ResNet152 model trained to classify blood smear images into three categories—uninfected, falciparum malaria, and vivax malaria—is demonstrated by the training output that is presented. The model is trained iteratively over ten epochs, gaining the ability to identify patterns and features in the input images and use them to provide precise predictions. The model performs moderately at first, achieving a training accuracy of 71.85% in the first epoch with a corresponding loss of 1.1326.

The performance indicators exhibit changes over epochs as training advances, which are indicative of the model's learning process. Over time, the accuracy increases steadily, peaking at 95.60% at the last epoch, while the loss continually declines, bottoming out at 0.1216. These patterns imply that as training goes on, the model efficiently picks up on differentiating between the three classes, with fewer mistakes and more accuracy.

The model consistently shows excellent accuracy on the validation set during training, ranging from 87.91% to 97.24% over epochs. The validation loss, which ranges from 0.0938 to 0.3217, likewise stays rather low, demonstrating the model's strong ability to generalise to new data. All things considered, these findings indicate that the ResNet152 model performs admirably on both the training and validation sets when it comes to classifying blood smear images into the designated categories.

In summary, the training output provided suggests that the ResNet152 model is a reliable tool for the detection and categorization of malaria parasites. The model may attain high accuracy and low loss because to its capacity to acquire discriminative features and patterns from the input images. This makes it appropriate for use in real-world medical diagnostic and research applications. To completely evaluate the model's performance and generalisation skills in real-world circumstances, more testing on more datasets could be required.

9.10 Hybrid Model for Feature Extraction with Ensemble Learning for Classification

9.10.1 Feature Extraction on Unprocessed image data

9.10.1.1 Random Forrest

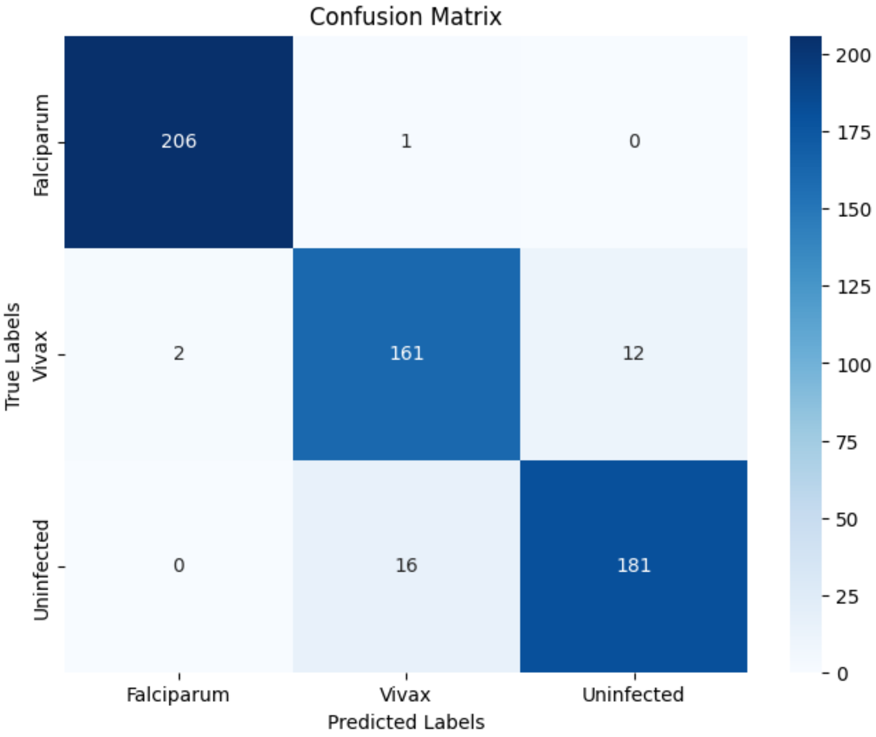


Figure 30 CONFUSION MATRIX: RANDOM FORREST WITH UNRPOCESSED DATA

Accuracy	Precision	Recall
94.64594127806563	94.6541325691426	94.64594127806563

TABLE 1 PERFOMANCE METRICS: RANDOM FORREST WITH UNPROCEESSED DATA

Strong performance is shown across a variety of evaluation criteria by the Random Forest model, which was trained on feature extraction using InceptionResNetV2 for the classification of malaria data into uninfected, falciparum, and vivax.

Based on the obtained accuracy of roughly 94.65%, it can be concluded that the model performs well in accurately categorising the blood smear images. This high accuracy shows that the model has been able to make effective predictions on unseen data by learning significant patterns and features from the extracted representations produced by InceptionResNetV2.

At roughly 94.65%, precision—a metric that quantifies the proportion of accurately predicted positive observations to all anticipated positives—is likewise high. This suggests that the model is quite likely to be accurate when it predicts a certain class (uninfected, falciparum, or vivax), reducing false positives.

Comparably, the recall score is roughly 94.65%, which is the ratio of properly anticipated positive observations to all observations made during the actual class. This shows that the model can detect cases of uninfected, falciparum, and vivax malaria from the dataset and successfully captures the true positives for each class.

All things considered, these assessment metrics point to the Random Forest model—which was trained on feature extraction using InceptionResNetV2—being a strong and trustworthy tool for categorising malaria data. Its capacity to distinguish between infected and uninfected blood smear images with good accuracy, precision, and recall suggests that it may find useful in the diagnosis and study of malaria.

9.10.1.2 AdaBoost

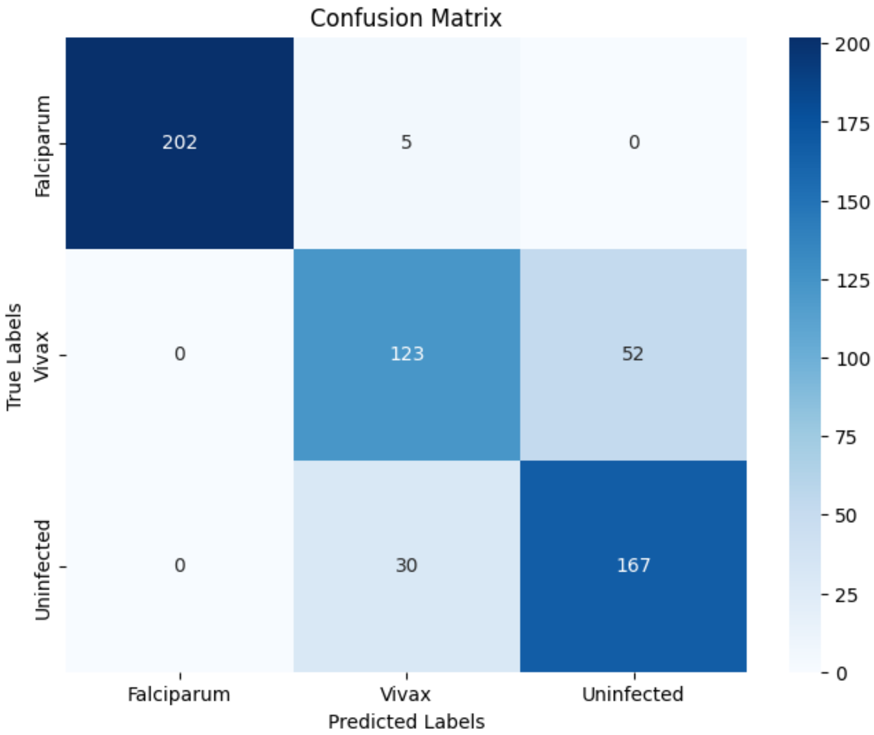


Figure 31 CONFUSION MATRIXADABOOST WITH UNRPOCESSED DATA

Accuracy	Precision	Recall
84.97409326424871	85.22589317917572	84.97409326424871

TABLE 2 PERFOMANCE METRICS: ADABOOST WITH UNPROCEESSED DATA

When it comes to classifying malaria data into three categories—uninfected, falciparum, and vivax—the Adaboost model, which was trained on feature extraction using InceptionResNetV2, performs admirably across important evaluation criteria.

The model shows the capacity to accurately categorise a considerable number of the blood smear images into their respective categories, with an accuracy of about 84.97%. This shows that the model is able to make good predictions on unseen data since it has acquired important patterns and features from the extracted representations created by InceptionResNetV2.

The precision ratio, which calculates the percentage of accurately predicted positive observations to all expected positives, is approximately 85.23%. This suggests that the model is reasonably accurate in predicting a particular class (uninfected, falciparum, or vivax), hence reducing false positives.

In a similar vein, the recall score is roughly 84.97%, which is the ratio of properly predicted positive observations to all observations in the actual class. This shows that the model can detect cases of uninfected, falciparum, and vivax malaria from the dataset and successfully captures the true positives for each class.

All things considered, even though the Adaboost model may not perform as well as some other models, its accuracy, precision, and recall scores show that it is still a trustworthy classifier for this particular task. Additional refinement and optimisation could possibly improve its performance even more.

9.10.2 Feature Extraction on image data preprocessed with sharpening and noise reduction

9.10.2.1 Random Forrest

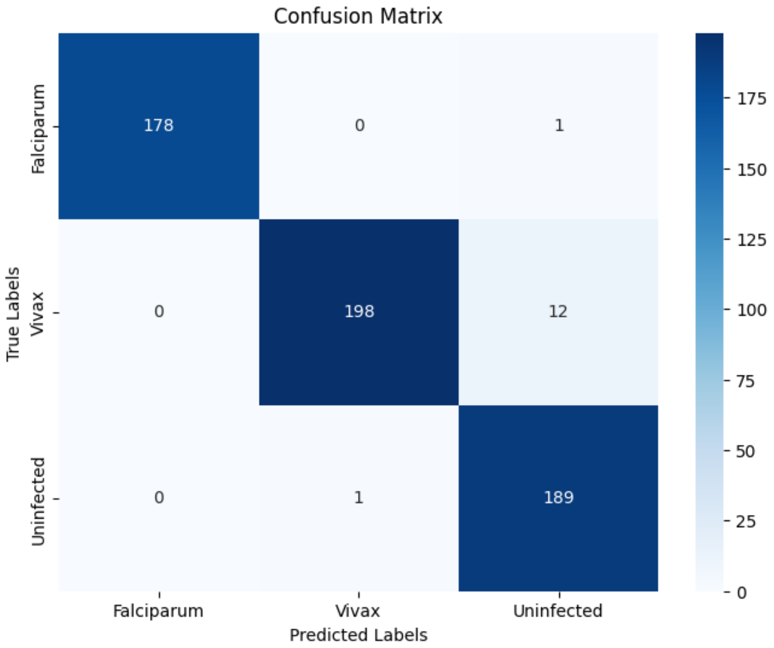


Figure 32 CONFUSION MATRIX: RANDOM FORREST WITH SHARPENING AND NOISE REDUCTION

Accuracy	Precision	Recall
97.58203799654577	97.7058723395187	97.58203799654577

TABLE 3 PERFOMANCE METRICS: RANDOM FORREST WITH SHARPENING AND NOISE REDUCTION

The Random Forest model, which was trained on feature extraction using InceptionResNetV2, performs remarkably well when it comes to dividing malaria data into three categories: uninfected, falciparum, and vivax.

The model shows a high degree of precision in its predictions, indicating its ability to distinguish between the various classes in the dataset, with an accuracy of almost 97.58%. This high accuracy shows that the characteristics that InceptionResNetV2 extracted are suitable for classification and help the model make sensible judgements.

Precision is approximately 97.71%, which is the percentage of accurately predicted positive instances out of all instances projected as positive. This suggests that the model has a low false positive rate, which means that it is very likely to be accurate when it predicts a particular class (uninfected, falciparum, or vivax).

Furthermore, the recall score is almost 97.58%, which is the percentage of properly predicted positive events among all actual positive instances. This shows that the model can detect cases of uninfected, falciparum, and vivax malaria from the dataset and successfully captures the true positives for each class.

Overall, the resilience and efficacy of the Random Forest model in identifying malaria data are indicated by its high accuracy, precision, and recall scores. Its effectiveness indicates that the characteristics recovered by InceptionResNetV2 are appropriate for this classification task, which makes it a dependable option for the detection of malaria parasites.

9.10.2.2 AdaBoost

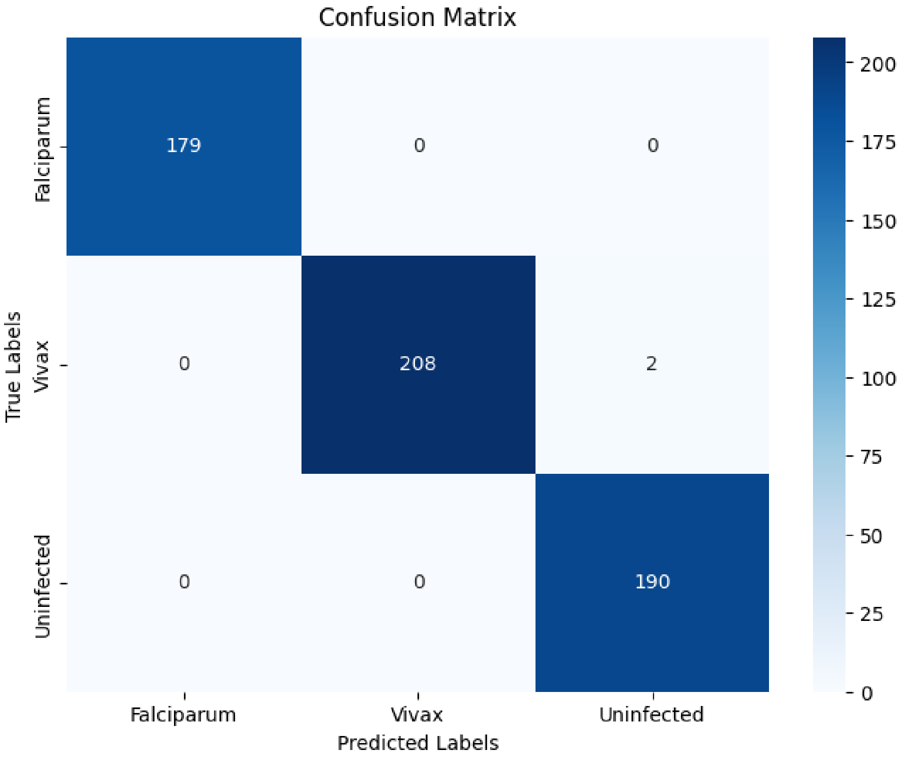


Figure 33CONFUSION MATRIX: ADABOOST WITH SHARPENING AND NOISE REDUCTION

Accuracy	Precision	Recall
99.65457685664939	99.65817501439264	99.65457685664939

TABLE 4 PERFOMANCE METRICS: ADABOOST WITH SHARPENING AND NOISE REDUCTION

The AdaBoost model, which was trained on feature extraction using InceptionResNetV2, performs remarkably well when it comes to dividing malaria data into three categories: uninfected, falciparum, and vivax.

The model has a very high degree of precision in its predictions, demonstrating its ability to reliably distinguish between the various classes within the dataset, with an accuracy of almost 99.65%. Given this high accuracy, it is likely that the characteristics that InceptionResNetV2 extracted are very useful and enhance the prediction ability of the model.

Precision, defined as the percentage of accurately predicted positive instances among all positively predicted instances, is approximately 99.66%. This suggests that the model has an incredibly low false positive rate, which means that it almost always forecasts the proper class (uninfected, falciparum, or vivax) when it makes a prediction.

Moreover, the recall score is roughly 99.65%, which is the percentage of correctly predicted positive events among all actual positive instances. This shows that the model can correctly identify cases of uninfected, falciparum, and vivax malaria from the dataset, indicating that it catches the true positives for each class.

The remarkable accuracy, precision, and recall scores of the AdaBoost model demonstrate its overall robustness and efficacy in identifying malaria data. Its results highlight how well the features recovered by InceptionResNetV2 fit this classification challenge, which makes AdaBoost an extremely dependable option for detecting malaria parasites.

9.10.3 Feature Extraction on image data preprocessed with Gray World Normalization

9.10.3.1 Gray World Normalization

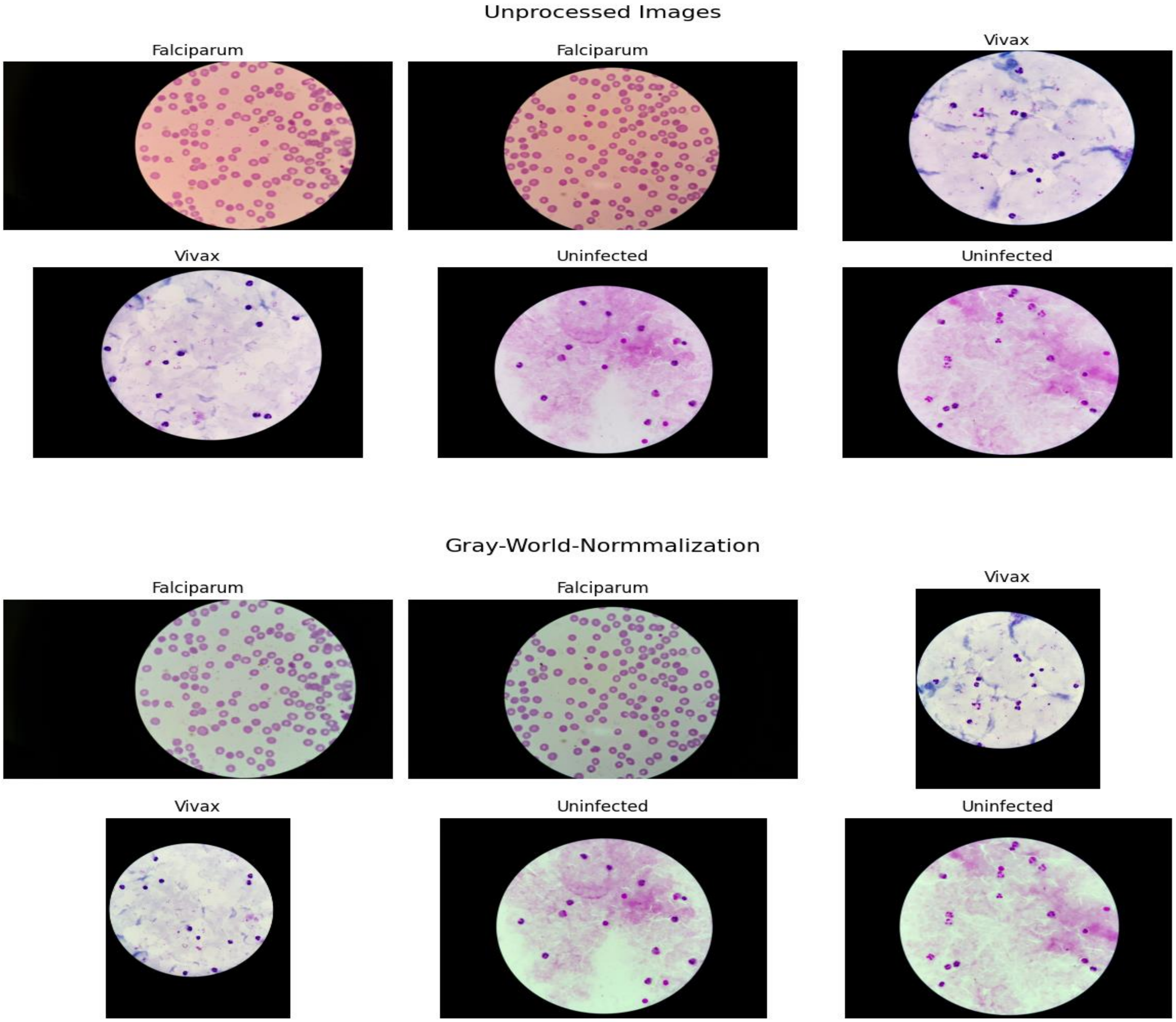


Figure 34 GRAY WORLD NORMALIZATION

The technique known as Grey World Normalisation (GWN) is used to improve the colour balance of photographs, especially when diverse lighting conditions are present or different sources of illumination are employed to record the image. Applying GWN to blood smear pictures for malaria dataset analysis has a number of noteworthy benefits.

First of all, GWN lessens the effect that changes in illumination have on the colour look of blood smear photographs. Shadows, reflections, artificial lighting, and other factors may cause the colours of the parasites and blood cells to appear differently in these photographs since they may have been taken under various lighting sets or conditions. GWN modifies the image colour balance to guarantee that, independent of the lighting conditions in which they were taken, all samples retain accurate and consistent colour representation.

Second, GWN lessens the impacts of colour cast by normalising the colour distribution of blood smear images. When specific colours predominate in an image as a result of ambient elements or the source of illumination, this is known as colour cast. Colour cast can alter the look of blood cells and parasites in blood smear images, making it difficult to identify and categorise them with accuracy. By eliminating or minimising colour cast, GWN produces photos with more balanced and natural colour tones, making it easier to analyse and classify malaria parasites.

Additionally, GWN enhances the general contrast and observability of features in blood smear pictures. Through the process of equalising colour distribution throughout the image, GWN improves the definition and clarity of key structures such background components, parasites, and blood cells. The enhanced contrast facilitates the detection and classification of parasites by automated analysis methods, resulting in more dependable diagnostic results.

Additionally, GWN enhances the general contrast and observability of features in blood smear pictures. Through the process of equalising colour distribution throughout the image, GWN improves the definition and clarity of key structures such background components, parasites, and blood cells. The enhanced contrast facilitates the detection and classification of parasites by automated analysis methods, resulting in more dependable diagnostic results.

9.10.3.2 Random Forrest

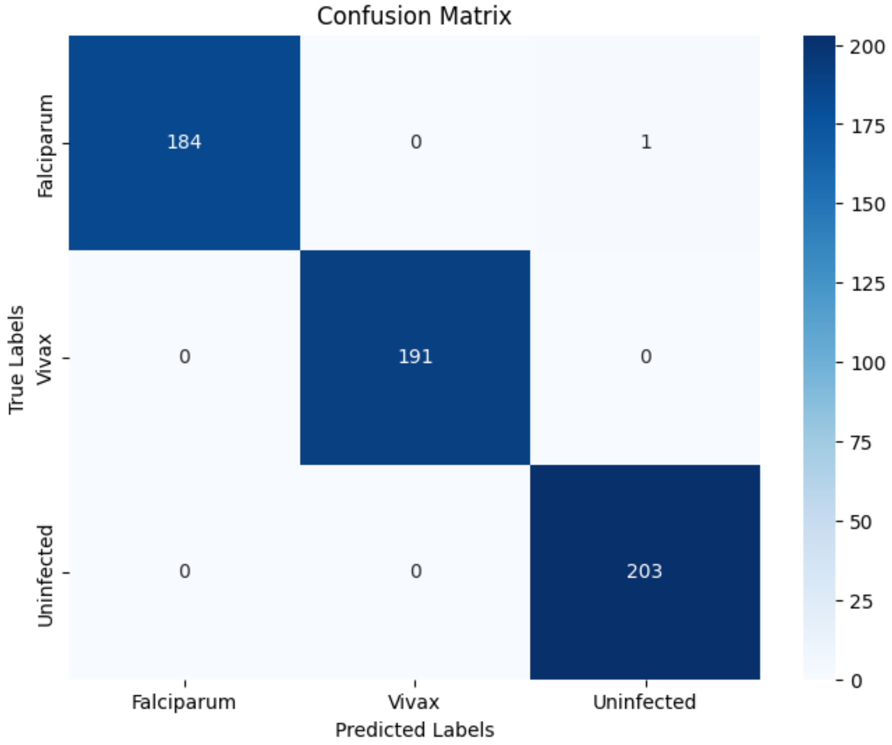


Figure 35 CONFUSION MATRIX: RANDOM FOREST WITH GRAY WORLD NORMALIZATION

Accuracy	Precision	Recall
99.8272884283247	99.82813505367605	99.8272884283247

TABLE 5 PERFOMANCE METRICS: RANDOM FORREST WITH GRAY WORLD NORMALIZATION

With excellent accuracy, precision, and recall scores, the random forest model trained on feature extraction using InceptionResNetV2 exhibits remarkable performance in classifying malaria data into three categories: uninfected, falciparum, and vivax.

The model achieves almost perfect classification accuracy on the dataset, with an accuracy of 99.83%. This suggests that the model performs a very good job of reliably classifying blood smear images into groups based on whether they are vivax, falciparum, or uninfected. The random forest algorithm's robustness and efficacy in capturing the intricate correlations between picture attributes and class labels are demonstrated by the high accuracy score.

Additionally, the 99.83% precision and recall scores show outstanding performance in terms of reducing false positives and false negatives. Recall is the ratio of genuine positive predictions to all actual positives in the dataset, whereas precision is the ratio of true positive forecasts to all positive predictions. The model appears to be able to efficiently identify true positive occurrences of each class while minimising false positive and false negative mistakes, based on the excellent precision and recall scores.

Overall, the random forest model that was trained using InceptionResNetV2 for feature extraction performs exceptionally well when it comes to dividing malaria data into three categories: uninfected, falciparum, and vivax. It is a useful tool for malaria diagnosis and research because of its high accuracy, precision, and recall scores, which show how well it can distinguish between various classes of blood smear images.

9.10.3.3 AdaBoost

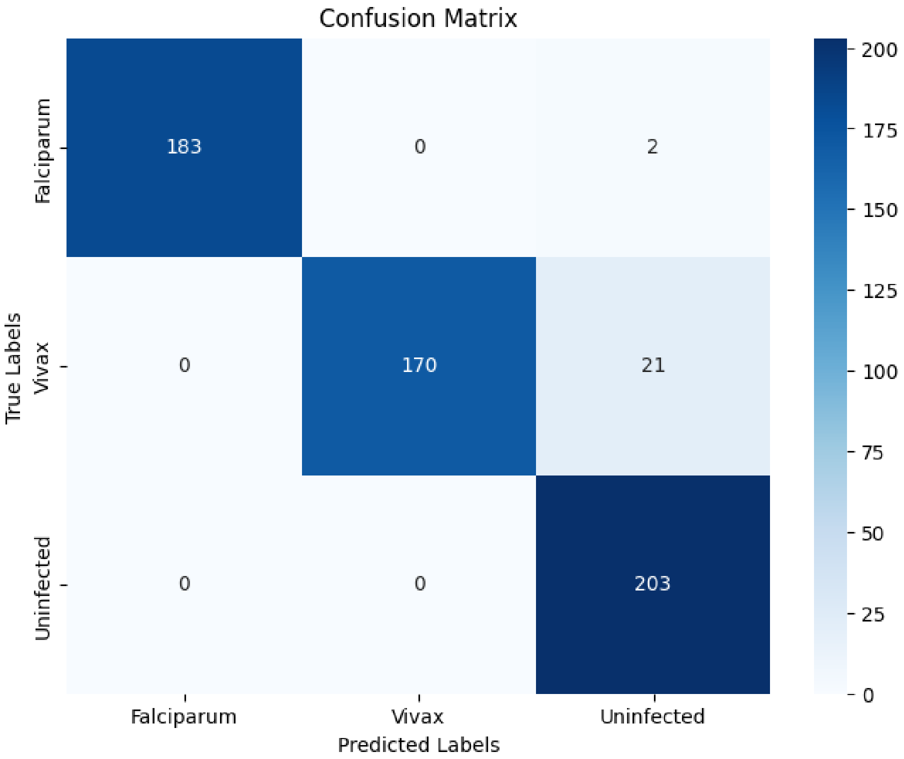


Figure 36 CONFUSION MATRIX: ADABOOST WITH GRAY WORLD NORMALIZATION

Accuracy	Precision	Recall
96.02763385146805	96.43190120286731	96.02763385146805

TABLE 5 PERFOMANCE METRICS: ADABOOST WITH GRAY WORLD NORMALIZATION

The high accuracy, precision, and recall scores of the AdaBoost model, which was trained on feature extraction using InceptionResNetV2, demonstrate its impressive ability in classifying malaria data into three categories: uninfected, falciparum, and vivax.

The model shows good overall classification performance with an accuracy of 96.03%, showing that it can accurately classify most of the blood smear images into the appropriate classes. This degree of precision indicates that the model successfully captures the discriminative characteristics required to differentiate between various groups of cells that are either uninfected or infected with malaria.

Furthermore, the model achieves a high ratio of accurately predicted positive instances, or true positives, to the overall number of positive predictions, or false positives and true positives, as indicated by the precision score of 96.43%. This implies that the model is dependable in detecting cases of each malaria class and has a low rate of false positive predictions.

In a similar vein, the model's 96.03% recall score indicates that it can accurately identify most true positives (positive instances) out of all actual positive instances in the dataset. This suggests that the model minimises false negatives by accurately capturing the presence of each class in the sample without missing many examples.

To summarise, the AdaBoost model, which was trained on feature extraction using InceptionResNetV2, exhibits strong performance in accurately, precisely, and recallably classifying blood smear images into three categories: uninfected, falciparum, and vivax. Its capacity to precisely recognise cells infected with malaria can greatly aid in the diagnosis and investigation of malaria.

Chapter 10

<

Conclusion and Future Work

10.1 Parasite Detection

Model	Accuracy	Loss	Validation Accuracy	Validation Loss
CNN	97.31%	7.63%	97.52%	6.94%
VGG-19	98.37%	4.96%	97.08%	7.34%

TABLE 7 MODELS FOR PARASITE DETECTION

Promising findings are obtained when comparing the CNN and VGG-19 models for malaria parasite detection. This comparison shows how well both architectures identify blood smear images. In terms of accuracy, the VGG-19 model performs marginally better than the CNN model, with an astounding accuracy of 98.37% as opposed to 97.31% for CNN. This implies that VGG-19's deeper architecture, with its several convolutional layers, performs better in classification because it is better at learning complex features seen in the photos. Furthermore, VGG-19 shows a lower loss value of 4.96% as opposed to CNN's loss of 7.63%, suggesting that it minimises the difference between predicted and actual values and converges more successfully throughout training.

Both models have excellent validation accuracies of 97%, suggesting their robustness in generalising to unknown data, despite the modest performance discrepancy. It is noteworthy, nonetheless, that the CNN model outperforms VGG-19 in terms of validation accuracy, achieving 97.52% as opposed to 97.08% for VGG-19. This implies that the CNN model may generalise slightly better to unknown data, even though VGG-19 may perform better overall on the training set. Overall, the findings demonstrate the efficacy of convolutional neural networks in the detection of malaria parasites, with both CNN and VGG-19 models exhibiting noteworthy performance metrics. Specific criteria like training time, processing resources, and the intended balance between accuracy and generalisation capacity may influence which of the two architectures is used.

10.2 Parasite Classification

Model	Pre-processing	Accuracy	Loss	Validation accuracy	Validation loss
VGG-19	no	97.88%	6.52%	83.59%	39.37%
InceptionV3	no	99.14%	2.39%	84.28%	14.21%
Xception	no	97.32%	13.26%	77.55%	19.58%
ResNet152	no	92.62%	20.05%	94.65%	14.66%
VGG-19	yes	99.44%	2.28%	97.75%	8.80%
InceptionV3	yes	99.91%	0.41%	100%	0
Xception	yes	99.53%	3.89%	99.65%	4.26%

ResNet152	yes	95.60%	12.16%	96.55%	9.38%

TABLE 7 MODELS FOR PARASITE CLASSIFICATION

The table provides an extensive comparison of multiple deep learning models, with and without pre-processing, for the classification of blood smear images into different malaria detection categories. These models include VGG-19, InceptionV3, Xception, and ResNet152. The models perform at different levels when pre-processing is not used. InceptionV3 has the best accuracy at 99.14%, closely followed by VGG-19 at 97.88%. For the majority of models, there is a discernible decline in validation accuracy, suggesting possible overfitting problems. The models may have learned to classify the training data too well, which would have limited their ability to generalise on unseen data, according to this disparity between training and validation accuracy.

Pre-processing strategies are found to significantly improve the models' performance, with notable gains seen in all architectures. Specifically, InceptionV3 attains flawless validation accuracy with pre-processing, delivering an astounding 99.91% accuracy. In a similar vein, VGG-19 and Xception show notable advancements as well, achieving validation accuracies of 99.65% and 97.75%, respectively. Pre-processing helps to reduce overfitting to some extent, but as the significant variations in training and validation accuracy show, overfitting is still occasionally seen. In order to address overfitting and guarantee that the models generalise well to new data, this emphasises the necessity of cautious regularisation techniques and hyperparameter tuning. It also emphasises the significance of finding a balance between model complexity and generalisation ability in malaria detection tasks.

10.3 Feature Extraction on image data preprocessed with Gray World Normalization

Model	Pre-processing type	Accuracy	Precision	Recall
Random Forrest	Unprocessed	94.64594127806563	94.6541325691426	94.64594127806563
Random Forrest	Sharpening and noise reduction	97.58203799654577	97.7058723395187	97.58203799654577
Random Forrest	Gray world normalization	99.8272884283247	99.82813505367605	99.8272884283247
AdaBoost	Unprocessed	84.97409326424871	85.22589317917572	84.97409326424871
AdaBoost	Sharpening and noise reduction	99.65457685664939	99.65817501439264	99.65457685664939
AdaBoost	Gray world normalization	96.02763385146805	96.43190120286731	96.02763385146805

TABLE 7 MODELS FOR HYBRID MODEL FOR FEATURE EXTRACTION WITH ENSEMBLE LEARNING FOR CLASSIFICATION

In order to categorise malaria picture data into uninfected, falciparum, and vivax categories, the table presents an insightful comparison of the performance of several models using various preprocessing strategies on feature extraction using InceptionResNetV2. Random Forest accomplishes an accuracy of 94.65% without any pre-processing, indicating that it can categorise the images fairly effectively. However, the accuracy increases dramatically to 97.58% when sharpening and noise reduction techniques are applied, demonstrating the usefulness of these preprocessing techniques in improving model performance. Moreover, the utilisation of grey world normalisation yields an impressive accuracy of 99.83%, underscoring its efficacy in augmenting classification accuracy.

There is a similar tendency when looking at AdaBoost. The model's accuracy is 84.97% without pre-processing and significantly increases to 99.65% with sharpening and noise reduction. The accuracy after grey world normalisation is very high at 96.03%. These findings highlight how crucial pre-processing methods are for enhancing machine learning models' functionality, particularly when working with image data. The models' capacity to accurately classify malaria images into multiple categories is further indicated by the high precision and recall scores obtained from all models and pre-processing methods, indicating their potential for practical use in malaria diagnosis and detection.

In order to reduce the possibility of overfitting in machine learning models, feature extraction is essential, particularly when working with complicated datasets like picture collections. The models can take advantage of the learnt representations of features from large datasets like ImageNet by employing pre-trained convolutional neural networks (CNNs) like InceptionResNetV2 for feature extraction. These pre-trained models are already capable of recognising both higher-level features, such as object pieces and structures, and lower-level features, such as edges, textures, and forms.

By using feature extraction, we can avoid starting from zero while training a deep neural network on the target dataset, which frequently calls for a significant quantity of labelled data and CPU power. Alternatively, we can extract pertinent features from the input photos using the pre-trained CNN. Without overfitting to the training set, these extracted features provide useful representations of the input data that capture its salient aspects.

Because extracted features are more likely to be significant across datasets, feature extraction improves the models' ability to generalise to previously unseen data. The capacity to generalise helps lower the chance of overfitting, a phenomenon in which the model fails to acquire significant patterns and instead becomes accustomed to the noise and outliers present in the training data. Thus, even with minimal training data, feature extraction helps the models attain improved accuracy and robustness, which makes them more appropriate for real-world uses like malaria picture categorization.

10.4 Future Scope

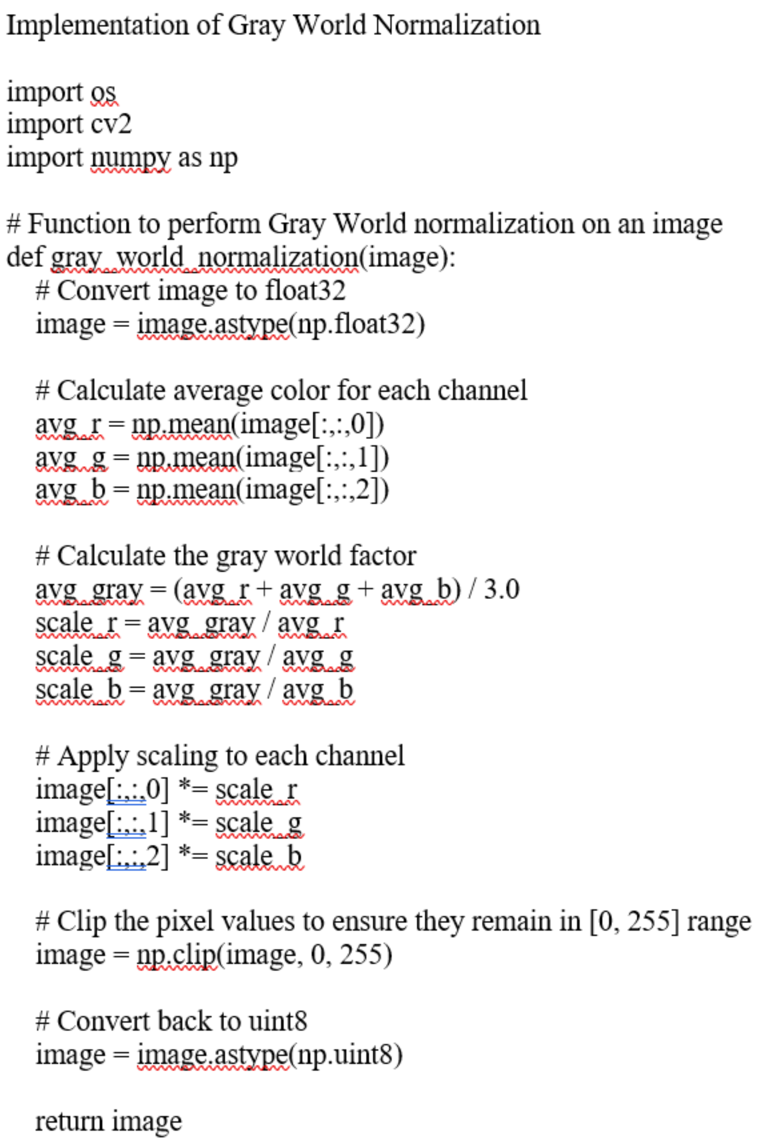
There are a lot of exciting opportunities ahead for improving the current frameworks in the field of malaria parasite categorization. Using federated learning approaches to leverage the architecture to accept greater geographical variants of the malaria virus is one possible direction. Federated learning allows the pooling of various datasets while maintaining data privacy by decentralising the model training process over multiple geographical sites. This method not only helps the model better capture regional variations in malaria but also promotes international cooperation among healthcare facilities.

Moreover, creating a real-time application to implement the paradigm is a crucial project for the future. A real-time software would make it easier and faster to diagnose malaria, enabling both individuals and medical professionals to make informed decisions on time. An application like this might function alone or interface with current healthcare systems to provide real-time feedback on blood smear images for the diagnosis of malaria. Furthermore, integrating functions such as image collection and processing directly on mobile devices will improve the solution's scalability and accessibility, potentially revolutionising the diagnosis of malaria in areas with limited resources. All things considered, these upcoming initiatives represent a determined attempt to use technology to support global malaria preventive and control programmes that are more successful.

Appendices

<

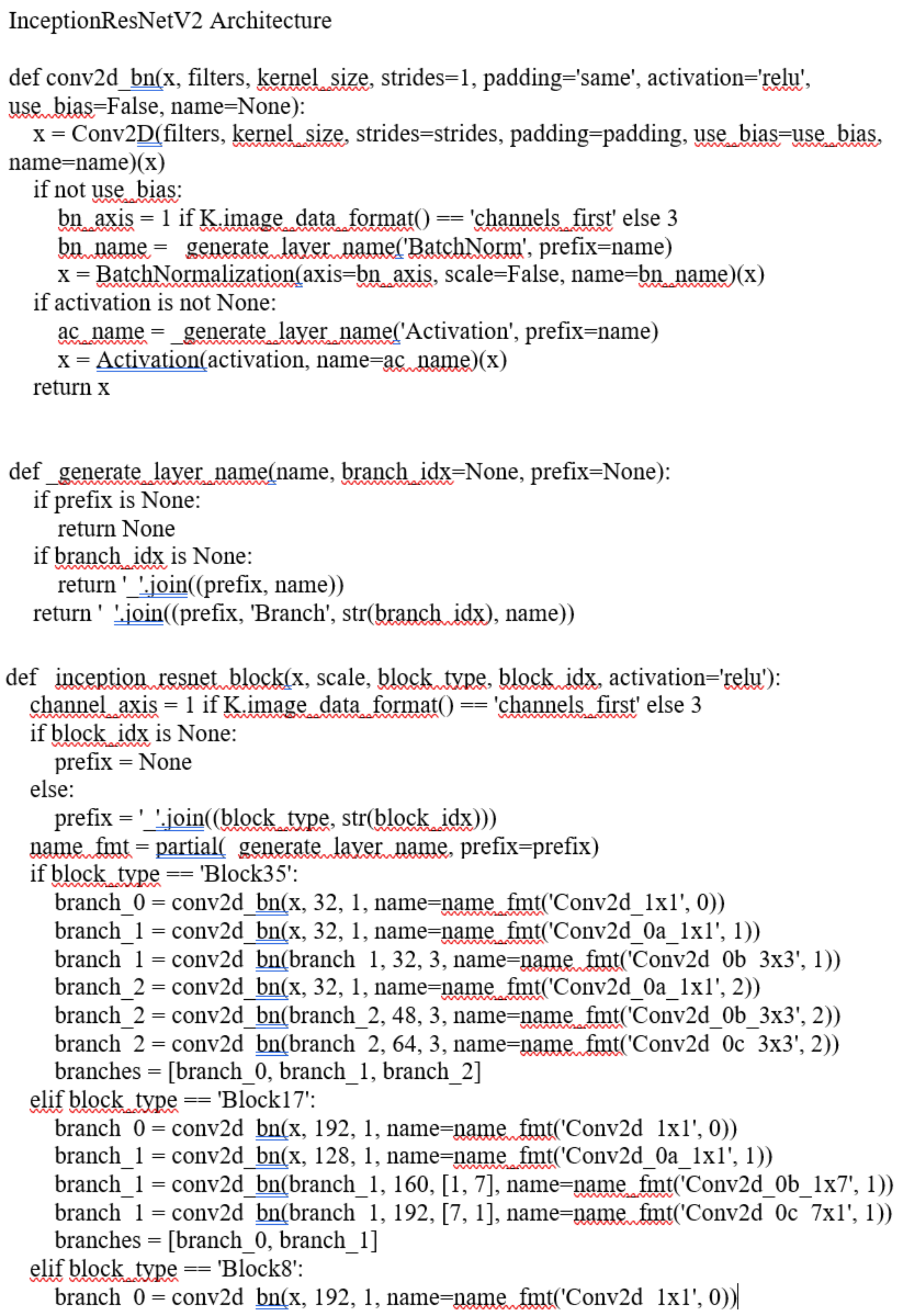
Appendix 1



Implementation of Gray World Normalization

```
import os
import cv2
import numpy as np

# Function to perform Gray World normalization on an image
def gray_world_normalization(image):
    # Convert image to float32
    image = image.astype(np.float32)

    # Calculate average color for each channel
    avg_r = np.mean(image[:,:,0])
    avg_g = np.mean(image[:,:,1])
    avg_b = np.mean(image[:,:,2])

    # Calculate the gray world factor
    avg_gray = (avg_r + avg_g + avg_b) / 3.0
    scale_r = avg_gray / avg_r
    scale_g = avg_gray / avg_g
    scale_b = avg_gray / avg_b

    # Apply scaling to each channel
    image[:,:,0] *= scale_r
    image[:,:,1] *= scale_g
    image[:,:,2] *= scale_b

    # Clip the pixel values to ensure they remain in [0, 255] range
    image = np.clip(image, 0, 255)

    # Convert back to uint8
    image = image.astype(np.uint8)

    return image
```

Appendix 2



InceptionResNetV2 Architecture

```
def conv2d_bn(x, filters, kernel_size, strides=1, padding='same', activation='relu',
use_bias=False, name=None):
    x = Conv2D(filters, kernel_size, strides=strides, padding=padding, use_bias=use_bias,
name=name)(x)
    if not use_bias:
        bn_axis = 1 if K.image_data_format() == 'channels_first' else 3
        bn_name = _generate_layer_name('BatchNorm', prefix=name)
        x = BatchNormalization(axis=bn_axis, scale=False, name=bn_name)(x)
    if activation is not None:
        ac_name = _generate_layer_name('Activation', prefix=name)
        x = Activation(activation, name=ac_name)(x)
    return x


def _generate_layer_name(name, branch_idx=None, prefix=None):
    if prefix is None:
        return None
    if branch_idx is None:
        return '_'.join((prefix, name))
    return '_'.join((prefix, 'Branch', str(branch_idx), name))


def _inception_resnet_block(x, scale, block_type, block_idx, activation='relu'):
    channel_axis = 1 if K.image_data_format() == 'channels_first' else 3
    if block_idx is None:
        prefix = None
    else:
        prefix = '_'.join((block_type, str(block_idx)))
    name_fmt = partial(_generate_layer_name, prefix=prefix)
    if block_type == 'Block35':
        branch_0 = conv2d_bn(x, 32, 1, name=name_fmt('Conv2d_1x1', 0))
        branch_1 = conv2d_bn(x, 32, 1, name=name_fmt('Conv2d_0a_1x1', 1))
        branch_1 = conv2d_bn(branch_1, 32, 3, name=name_fmt('Conv2d_0b_3x3', 1))
        branch_2 = conv2d_bn(x, 32, 1, name=name_fmt('Conv2d_0a_1x1', 2))
        branch_2 = conv2d_bn(branch_2, 48, 3, name=name_fmt('Conv2d_0b_3x3', 2))
        branch_2 = conv2d_bn(branch_2, 64, 3, name=name_fmt('Conv2d_0c_3x3', 2))
        branches = [branch_0, branch_1, branch_2]
    elif block_type == 'Block17':
        branch_0 = conv2d_bn(x, 192, 1, name=name_fmt('Conv2d_1x1', 0))
        branch_1 = conv2d_bn(x, 128, 1, name=name_fmt('Conv2d_0a_1x1', 1))
        branch_1 = conv2d_bn(branch_1, 160, [1, 7], name=name_fmt('Conv2d_0b_1x7', 1))
        branch_1 = conv2d_bn(branch_1, 192, [7, 1], name=name_fmt('Conv2d_0c_7x1', 1))
        branches = [branch_0, branch_1]
    elif block_type == 'Block8':
        branch_0 = conv2d_bn(x, 192, 1, name=name_fmt('Conv2d_1x1', 0))
```

```
branch_1 = conv2d_bn(x, 192, 1, name=name_fmt('Conv2d_0a_1x1', 1))
```

```
        branch_1 = conv2d_bn(branch_1, 224, [1, 3], name=name_fmt('Conv2d_0b_1x3', 1))
        branch_1 = conv2d_bn(branch_1, 256, [3, 1], name=name_fmt('Conv2d_0c_3x1', 1))
        branches = [branch_0, branch_1]
    else:
        raise ValueError(str(block_type))

    mixed = Concatenate(axis=channel_axis, name=name_fmt('Concatenate'))(branches)
    up =
conv2d_bn(mixed,K.int_shape(x)[channel_axis],1,activation=None,use_bias=True,name=name_
fmt('Conv2d_1x1'))
    x = Lambda(lambda inputs, scale: inputs[0] + inputs[1] * scale,
               output_shape=K.int_shape(x)[1:],arguments={'scale':
scale},name=name_fmt('ScaleSum'))([x, up])
    if activation is not None:
        x = Activation(activation, name=name_fmt('Activation'))(x)
    return x
      def
      InceptionResNetV2(include_top=True,weights='imagenet',input_tensor=None,input_shape=Non
      e,pooling=None,classes=1000,dropout_keep_prob=0.8):

          if weights not in {'imagenet', None}:
              raise ValueError('Error')

          if weights == 'imagenet' and include_top and classes != 1000:
              raise ValueError('Error')

          # Determine proper input shape
          input_shape =
        obtain_input_shape(input_shape,default_size=299,min_size=139,data_format=K.image_data_f
      ormat(),require_flatten=False,weights=weights)

          if input_tensor is None:
              img_input = Input(shape=input_shape)
          else:
              if not K.is_keras_tensor(input_tensor):
                  img_input = Input(tensor=input_tensor, shape=input_shape)
              else:
                  img_input = input_tensor
          # Stem block: 35 x 35 x 192
          x = conv2d_bn(img_input, 32, 3, strides=2, padding='valid', name='Conv2d_1a_3x3')
          x = conv2d_bn(x, 32, 3, padding='valid', name='Conv2d_2a_3x3')
          x = conv2d_bn(x, 64, 3, name='Conv2d_2b_3x3')
          x = MaxPooling2D(3, strides=2, name='MaxPool_3a_3x3')(x)
          x = conv2d_bn(x, 80, 1, padding='valid', name='Conv2d_3b_1x1')
          x = conv2d_bn(x, 192, 3, padding='valid', name='Conv2d_4a_3x3')
          x = MaxPooling2D(3, strides=2, name='MaxPool_5a_3x3')(x)
```

```
# Mixed 5b (Inception-A block): 35 x 35 x 320
    channel_axis = 1 if K.image_data_format() == 'channels_first' else 3
    name_fmt = partial(_generate_layer_name, prefix='Mixed_5b')
    branch_0 = conv2d_bn(x, 96, 1, name=name_fmt('Conv2d_1x1', 0))
    branch_1 = conv2d_bn(x, 48, 1, name=name_fmt('Conv2d_0a_1x1', 1))
    branch_1 = conv2d_bn(branch_1, 64, 5, name=name_fmt('Conv2d_0b_5x5', 1))
    branch_2 = conv2d_bn(x, 64, 1, name=name_fmt('Conv2d_0a_1x1', 2))
    branch_2 = conv2d_bn(branch_2, 96, 3, name=name_fmt('Conv2d_0b_3x3', 2))
    branch_2 = conv2d_bn(branch_2, 96, 3, name=name_fmt('Conv2d_0c_3x3', 2))
    branch_pool = AveragePooling2D(3, strides=1, padding='same',
name=name_fmt('AvgPool_0a_3x3', 3))(x)
    branch_pool = conv2d_bn(branch_pool, 64, 1, name=name_fmt('Conv2d_0b_1x1', 3))
    branches = [branch_0, branch_1, branch_2, branch_pool]
    x = Concatenate(axis=channel_axis, name='Mixed_5b')(branches)

    # 10x Block35 (Inception-ResNet-A block): 35 x 35 x 320
    for block_idx in range(1, 11):
        x = _inception_resnet_block(x, scale=0.17, block_type='Block35', block_idx=block_idx)
          # Mixed 6a (Reduction-A block): 17 x 17 x 1088
    name_fmt = partial(_generate_layer_name, prefix='Mixed_6a')
    branch_0 = conv2d_bn(x,384, 3, strides=2, padding='valid',
name=name_fmt('Conv2d_1a_3x3', 0))
    branch_1 = conv2d_bn(x, 256, 1, name=name_fmt('Conv2d_0a_1x1', 1))
    branch_1 = conv2d_bn(branch_1, 256, 3, name=name_fmt('Conv2d_0b_3x3', 1))
    branch_1 =
conv2d_bn(branch_1,384,3,strides=2,padding='valid',name=name_fmt('Conv2d_1a_3x3', 1))
    branch_pool =
MaxPooling2D(3,strides=2,padding='valid',name=name_fmt('MaxPool_1a_3x3', 2))(x)
    branches = [branch_0, branch_1, branch_pool]
    x = Concatenate(axis=channel_axis, name='Mixed_6a')(branches)

    # 20x Block17 (Inception-ResNet-B block): 17 x 17 x 1088
    for block_idx in range(1, 21):
        x = _inception_resnet_block(x,scale=0.1,block_type='Block17',block_idx=block_idx)

    # Mixed 7a (Reduction-B block): 8 x 8 x 2080
    name_fmt = partial(_generate_layer_name, prefix='Mixed_7a')
    branch_0 = conv2d_bn(x, 256, 1, name=name_fmt('Conv2d_0a_1x1', 0))
    branch_0 =
conv2d_bn(branch_0,384,3,strides=2,padding='valid',name=name_fmt('Conv2d_1a_3x3', 0))
    branch_1 = conv2d_bn(x, 256, 1, name=name_fmt('Conv2d_0a_1x1', 1))
    branch_1 =
conv2d_bn(branch_1,288,3,strides=2,padding='valid',name=name_fmt('Conv2d_1a_3x3', 1))
    branch_2 = conv2d_bn(x, 256, 1, name=name_fmt('Conv2d_0a_1x1', 2))
    branch_2 = conv2d_bn(branch_2, 288, 3, name=name_fmt('Conv2d_0b_3x3', 2))
    branch_2 =
conv2d_bn(branch_2,320,3,strides=2,padding='valid',name=name_fmt('Conv2d_1a_3x3', 2))
    branch_pool =
MaxPooling2D(3,strides=2,padding='valid',name=name_fmt('MaxPool_1a_3x3', 3))(x)
    branches = [branch_0, branch_1, branch_2, branch_pool]
    x = Concatenate(axis=channel_axis, name='Mixed_7a')(branches)

    # 10x Block8 (Inception-ResNet-C block): 8 x 8 x 2080
    for block_idx in range(1, 10):
        x = _inception_resnet_block(x,scale=0.2,block_type='Block8',block_idx=block_idx)
    x = _inception_resnet_block(x,scale=1.,activation=None,block_type='Block8',block_idx=10)

    # Final convolution block
    x = conv2d_bn(x, 1536, 1, name='Conv2d_7b_1x1')

    if include_top:
        # Classification block
        x = GlobalAveragePooling2D(name='AvgPool')(x)
        x = Dropout(1.0 - dropout_keep_prob, name='Dropout')(x)
        x = Dense(classes, name='Logits')(x)
        x = Activation('softmax', name='Predictions')(x)
    else:
        if pooling == 'avg':
            x = GlobalAveragePooling2D(name='AvgPool')(x)
        elif pooling == 'max':
            x = GlobalMaxPooling2D(name='MaxPool')(x)

    # Ensure that the model takes into account
    # any potential predecessors of `input_tensor`
    if input_tensor is not None:
        inputs = get_source_inputs(input_tensor)
    else:
        inputs = img_input
    # Create model
    model = Model(inputs, x, name='inception_resnet_v2')
```

REFERENCES

[1] Garba, S., Abdullahi, M.B., Bashir, S.A. and Abisoye, O.A., 2022, November. Implementation of Malaria Parasite Detection and Species Classification Using Dilated Convolutional Neural Network. In 2022 5th Information Technology for Education and Development (ITED) (pp. 1-6). IEEE.

[2] Kakkar, B., goyal, M., Johri, P. and Kumar, Y., 2023. Artificial Intelligence-Based Approaches for Detection and Classification of Different Classes of Malaria Parasites Using Microscopic Images: A Systematic Review. Archives of Computational Methods in Engineering, pp.1-20.

[3] Tan, D. and Liang, X., 2023. Multiclass malaria parasite recognition based on transformer models and a generative adversarial network. Scientific Reports, 13(1), p.17136.

[4] Swastika, W., Kristianti, G.M. and Widodo, R.B., 2021, April. Effective preprocessed thin blood smear images to improve malaria parasite detection using deep learning. In Journal of Physics: Conference Series (Vol. 1869, No. 1, p. 012092). IOP Publishing.

[5] Ohdar, K. and Nigam, A., 2023, July. A Robust Approach for Malaria Parasite Identification with CNN Based Feature Extraction and Classification using SVM. In 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT) (pp. 1-6). IEEE.

[6] Wang, C., Dong, Y., Li, C., Oberstaller, J., Zhang, M., Gibbons, J., Pires, C.V., Xiao, M., Zhu, L., Jiang, R.H. and Kim, K., 2023. MalariaSED: a deep learning framework to decipher the regulatory contributions of noncoding variants in malaria parasites. Genome Biology, 24(1), p.231.

[7] Jusman, Y., Aftal, A.A., Tyassari, W., Kanafiah, S.N.A.M., Hayati, N. and Mohamed, Z., 2023, August. Classification of Parasite Malaria in Schizon Stage

with GoogleNet and VGG-19 Pre-Trained Models. In 2023 10th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE) (pp. 219-223). IEEE.

[8] Jusman, Y., Ardiyanto, Y., Kanafiah, S.N.A.M., Mohamed, Z. and Hassan, R., 2022, November. Classification System of Malaria Disease with Hu Moment Invariant and Support Vector Machines. In 2022 2nd International Conference on Electronic and Electrical Engineering and Intelligent System (ICE3IS) (pp. 365-368). IEEE.

[9] Kundu, T.K. and Anguraj, D.K., 2023, March. Using Image Classification-based RBC Extraction for Malarial Parasite Identification in Peripheral Blood Smear. In 2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS) (pp. 1554-1559). IEEE.

[10] Sifat, M.M.H. and Islam, M.M., 2020, June. A fully automated system to detect malaria parasites and their stages from the blood smear. In 2020 IEEE Region 10 Symposium (TENSYMP) (pp. 1351-1354). IEEE.