

**TEA LEAF DISEASE IDENTIFICATION USING MACHINE LEARNING
BY**

Apurbo Kumar Paul
ID: 181-15-11248

Mishala Akter Mitale
ID: 181-15-11236

Sinthia Afrin
ID: 181-15-10701

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering

Supervised By
Moushumi Zaman Bonny
Assistant Professor
Department of CSE
Daffodil International University

Co-Supervised By
Md. Abbas Ali Khan
Sr. Lecturer
Department of CSE
Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY
DHAKA, BANGLADESH
JANUARY 2021

APPROVAL

This Project/internship titled “**TEA-LEAF DISEASE IDENTIFICATION USING MACHINE LEARNING**” submitted by **Apurbo Kumar Paul**, ID No: **181-15-11248** , **Mst.Mishala Akter**, ID No: **181-15-11236** , and **Sinthia Afrin**, ID No: **181-15-10701** , to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **2 January 2022**.

BOARD OF EXAMINERS



Dr. Touhid Bhuiyan

Professor and Head

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University

Chairman



Moushumi Zaman Bonny

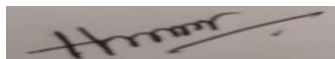
Assistant Professor

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University

Internal Examiner



Md. Mahfujur Rahman

Senior Lecturer

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University

Internal Examiner



Dr. Md Arshad Ali

Associate Professor

Department of Computer Science and Engineering

Hajee Mohammad Danesh Science and Technology University

External Examiner

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Moushumi Zaman Bonny, Assistant Professor, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised to:



Moushumi Zaman Bonny

Assistant Professor

Department of CSE

Daffodil International University

Co-Supervised by:

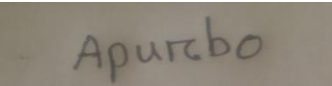


Md. Abbas Ali Khan

Senior Lecturer, Department of CSE

Daffodil International University

Submitted by:

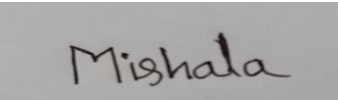


Apurbo Kumar Paul

ID: 181-15-11248

Department of CSE

Daffodil International University

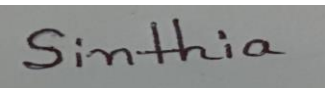


Mst. Mishala Akter

ID: 181-15-11236

Department of CSE

Daffodil International University



Sinthia Afrin

ID: 181-15-10701

Department of CSE

Daffodil International University

ACKNOWLEDGEMENT

First of all, I would like to express my heartfelt thanks and thanks to Almighty God for the divine blessing that allowed me to successfully complete last year's project.

We really grateful and wish our profound our indebtedness to **Mst.Moushumi Zaman Bonny**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of —*Machine Learning* to carry out this project. His endless patience ,scholarly guidance ,continual encouragement , constant and energetic supervision, constructive criticism , valuable advice ,reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to **Professor Dr. Touhid Bhuiyan**, Head, Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would also like to express our sincere thanks to the other members and staff of Panchagarkh Kazi & Kazi Tea Estate Limited and for their kind help in completing our project.

At the end of the course we would like to thank all my teammate from Daffodil International University who participated in this discussion. Lastly, we must respect the unwavering support of our parents and the hearts of our patients.

ABSTRACT

Tea plants have a lengthy lifespan, which makes disease management challenges. Tea leaves are commonly afflicted by bacterial, viral, or fungal infections, which significantly decrease tea production. Identifying disease tea leaves is critical to sustaining worldwide tea demand for a vast population. Early disease detection can assist in reducing losses. However, disease tea leaf detection was limited to image backgrounds and image-capture conditions. In the field of disease tea leaf identification, the convolutional neural network (CNN)-based model is the most popular topic of research. However, existing some CNN-based models have low recognition rates on independent datasets and are limited to learning large-scale network parameters. We proposed a novel CNN-based model to identify disease tea leaf in this study by decreasing network parameters. A number of CNN-based models, namely InceptionV3, MobileNetV2, MobileNet, Vgg16, Xception, and InceptionResNetV2, are trained to distinguish between healthy tea leaves and diseased tea leaves using a new dataset of 2000 tea leaf images. Out of a total of 100% data, we used 75% of image data for training and 25% of image data for testing. By using MobileNet we have achieved the highest accuracy which is 98%. To increase the image's overall accuracy, we used the data augmentation approach. In comparison to other approaches, the combination of transfer learning and data augmentation produces the highest efficiency.

TABLE OF CONTENTS

CONTENTS	PAGE
Board of examiners	ii
Declaration	iii
Acknowledgements	iv
Abstract	v
CHAPTER	
CHAPTER 1: INTRODUCTION	1-4
1.1 Introduction	1
1.2 Motivation	2
1.3 Objective	2
1.4 Research Questions	2
1.5 Expected Outcome	3
1.6 Layout of the Report	3-4
CHAPTER 2: BACKGROUND	5-8
2.1 Introduction	5
2.2 Related Works	5-7
2.3 Research Summary	7
2.4 Scope of the problem	8
2.5 Challenges	8

CHAPTER 3: RESEARCH METHODOLOGY	9-26
3.1 Introduction	9
3.2 Research Subject & Instrumentation	9-10
3.3 Data Collection Procedure	10-15
3.4 Proposed Methodology	16-17
3.4.1 Data Preprocessing	17
3.4.2 Data Augmentation	17-18
3.4.3 Convolutional Neural Network	18-25
3.4.4 Train the Model	25-26
3.5 Implementation Requirements	26
Chapter 4: EXPERIMENTAL RESULTS AND DISCUSSION	27-39
4.1 Introduction	27
4.2 Experimental Setup	27-28
4.3 Experimental Results & Analysis	28-38
4.4 Discussion	39
CHAPTER 5: IMPACT ON SOCIETY, ENVIRONMENT AND SUSTAINABILITY	40-41
5.1 Impact on Society	40
5.2 Impact on Environment	40
5.3 Ethical Aspects	41
5.4 Sustainability Plan	41

CHAPTER 6: SUMMARY, CONCLUSION AND FUTURE WORKS	42-43
6.1 Summary of the Study	42
6.2 Conclusions	42
6.3 Future Works	43
APPENDICES	44
REFERENCES	45-46

LIST OF FIGURES

FIGURES	PAGE
Figure 1.1: A small part of the raw dataset	16
Figure 1.2: Block diagram of proposed methodology	19
Figure 1.3: A typical CNN architecture	20
Figure 1.4: VGG16 architecture	21
Figure 1.5: InceptionV3 architecture	22
Figure 1.6: MobileNetV2 architecture	23
Figure 1.7:: InceptionResNetV2 architecture	24
Figure 1.8: MobileNet architecture	25
Figure 1.9: (a)MobileNetV2 model accuracy; (b)MobileNetV2 model loss	29
Figure: 2.1: Confusion matrix of MobileNetV2 model	30
Figure: 2.2: (a)MobileV2 model accuracy; (b) MobileV2 model loss	31
Figure: 2.3: Confusion matrix of MobileV2 model	31
Figure: 2.4: (a)InceptionV3 model accuracy; (b) InceptionV3 model loss	32
Figure: 2.5: Confusion matrix of InceptionV3 model	33
Figure:2.6:(a) InceptionResNetV2 model accuracy; (b) InceptionResNetV2 model loss	34
Figure: 2.7: Confusion matrix of InceptionResNetV2 model	34
Figure:2.8: (a)Xception model accuracy; (b) Xception model loss	35
Figure: 2.9: Confusion matrix of Xception model	36
Figure 3.1: (a)Vgg16 model accuracy; (b) Vgg16 model loss	37
Figure 3.2: Confusion matrix of Vgg16 model	37
Figure 3.3: Difference caauracy of difference model	39

LIST OF TABLES

TABLES	PAGE NO
Table 1.1: Experimental Result of different neural network model	28
Table 1.2: Classification report of MobileNet model	30
Table 1.3: Classification report of MobileNetV2 model	32
Table 1.4: Classification report of InceptionV3 model	33
Table 1.5: Classification report of InceptionResNetV2 model	35
Table 1.6: Classification report of Xception model	36
Table 1.7: Classification report of vgg16 model	38

CHAPTER 1

Introduction

1.1 Introduction

Tea is one of Bangladesh's most consumed beverages. The majority of Bangladeshis begin their day with a cup of tea. Bangladesh has become a significant tea-producing country. In total, there are currently 172 commercial tea gardens in the nation. The tea-producing districts are Maulvibazar, Habiganj, Sylhet, Chittagong, Panchagarh, Brahmanbaria, and Rangamait. Almost the whole region of Sylhet is devoted to lawns, and Srimangal is known as Bangladesh's tea capital. Bangladesh's tea is the country's second-largest export crop. Bangladesh's alternative bills amount to 1% of the country's GDP. Tea production in Bangladesh is severely hampered by a variety of pests and diseases, including a variety of insects, mites, nematodes, bacteria, algae, fungus, weeds, and other diseases that may be triggered by the region's peculiar circumstances. Bangladesh is a largely agricultural-dependent country. Agriculture is something that over 75% of the population agrees on immediately or indirectly. Every year, around 20% to 30% of the tea leaves are missing because of serious diseases. Farmers in the area use their eyes and previous expertise to determine the identification of tea diseases. Many times, specialists must be consulted to analyze the tea leaves when local farmers are unable to identify diseases; this approach is not only time-consuming but also costly, and it is extremely difficult to detect plant disease by glancing at the leaf's surface. To avoid misunderstandings with other ulcerative diseases, it's critical to correctly identify the disease. The concept of machine mastering by image approach will undoubtedly fix this issue. We can utilize deep learning to read every aspect of an image and physically extract capabilities from it using a convolutional neural network (CNN). We have a proclivity to use a complex strategy known as "Transfer Learning," which is extremely beneficial in deep learning.

1.2 Motivation

Tea is an essential source of vitamins A, C, E, and numerous B vitamins, and is regarded as one of the most important sources of these vitamins in the human body. Tea leaf disease is a severe problem in the tea industry. Along with the speed with which it spreads through the air and the microorganisms that cause it, the fact that the majority of farmers are unaware of many leaf diseases is a major source of concern. It is particularly concerning in Bangladesh, where the majority of farmers are uneducated. They still don't know much about leaf disease. As a result, they are experiencing a significant loss in their business each year. Rapid and precise diagnosis is required to manage and prevent the spread of the tea leaf disease. It is essential to be aware of the causes and symptoms of these diseases in order to properly care for tea trees. That's why we will carry out this task.

1.3 Objective

Our goal is to utilize image processing to detect the disease of a tea leaf in a very efficient and effective way and to create a system that can be used by any tea garden farmer to detect the disease and discover a very straightforward solution to a specific problem. Our goal with this study is to reduce the amount of time it takes to diagnose tea leaf diseases, lower the cost of identification, and boost the output of healthy tea leaves.

1.4 Research Question

It's incredibly difficult to diagnose disease from tea leaf images. When we first start working, we have a lot of questions. At that time, we had several unanswered questions.

- Which platform will be ideal for implementing our solutions?
- What kinds of data should we gather for this project?
- How is this work beneficial to people?
- Is it feasible to get a high accuracy rate?
- How effectively can we make use of it?

1.5 Expected Outcome

We want to create a model that effectively assists people all around the world who have lost a lot of money due to tea leaf disease. It can anticipate over 98% of every input. We have certain objectives for the project's conclusion. The following are some of them:

- Shorten the time it takes to diagnose disease tea leaf.
- Lower the price of diagnosis.
- Increase the production of tea.
- Correctly diagnose all leaf disease images.

1.6 Layout of the Report

The first chapter provided an overview of the project, including an introduction, motivation, and goal. We come up with some research questions and outline some of the predicted outcomes of our study. In this section, we go over the whole report layout of our thesis.

In chapter two, we explored the history of our project, which included relevant application and problem analysis work as well as certain project difficulties.

In chapter three, we'll go through the theory behind our proposed methodology. But first, we'll go through certain methods like collection of data, preprocessing, augmentation, and so forth. We'll use visuals and mathematical formulae to help you understand convolutional neural networks. We'll go over the model layers, testing, and training methods.

The performance of our suggested pre-trained model is compared to that of other prominent transfer learning models, such as InceptionV3, MobileNetV2, and InceptionRestNetV2, among others, in Chapter 4. Finally, we will wrap up the outcome discussion for our suggested transfer learning model, as well as for all other models.

The fifth chapter provided a summary of the research, as well as some recommendations for future research and our findings. We also state some of our limits and issues, but this chapter concludes by demonstrating the area's high future potential. This chapter is responsible for distributing the overall project summary of our study research.

CHAPTER 2

Background

2.1 Introduction

Deep learning has overcome problems in image processing and computer vision better than anything else. Even while some work has been done to describe how deep neural networks learn, they are still dark boxes. Although this topic is tough to learn, it is an advantage to us since it allows us to solve many complicated issues quickly. As a result, the goal of this project is to learn how to create a neural network using Tensorflow and Keras models.

2.2 Related Works

The Convolutional Neural Network (CNN) and Image Classification are two of the most commonly used deep learning algorithms. Despite the fact that tea leaf diseases are common, the image processing technique only performs a few associated jobs. Identified apple leaf disease using AlexNet. The dataset used contained 13,689 images of apple leaves and achieved an accuracy of 97.62%. [1]. G. Geetharamani and A. Pandian used Deep CNN (DCNN) to identify leaf diseases in 39 classes. The accuracy rate reached 96.46% [2]. Meanwhile, K. P. Ferentinos used 87,848 images to detect and diagnose plant diseases using the CNN model. Its success rate reached 96.53%. [3]. A. T. Sapkal and U. V. Kulkarni diagnosed leaf disease automatically with an accuracy of 93.85%, using two feature extraction techniques, namely Gray Level Covariance Matrix (GLCM) and Alexnet.[4]. Shaveta Malik, Tapas Kumar, and Amiya Kumar Sahoo created a model (FAST-PCA NN) that can identify or diagnose the disease tea leaves automatically with 86% accuracy. The authors also demonstrated how FAST, PCA, and NN provide tea producers and researchers with an effective,

quick, and automated technique for detecting leaf disease in this study. The study is divided into multiple sections; in the first, they attempt to use segmentation, which helped them improve the image and detect distinct edges. In the second part, features are extracted from the infected leaf image using feature descriptions such as HOG (Histogram of Gradient), FAST (Features from Accelerated Segment Test), and machine learning algorithms are used to classify the infected and healthy leaf images and find the best classification accuracy using the classifier. They also illustrate how the feature descriptors perform well after using PCA since it decreases dimensionality and produces better results. However, in our research, we used a CNN Transfer Learning model to extract features from pictures and suggested an improved image optimization approach called Data Augmentation to enhance the image.[5].

To diagnose brain cancers, Arshia Rehman and Saeeda Naz suggested a VGG16 transfer learning architecture. The fine-tuned VGG16 architecture achieved the best accuracy of 98.69% in their specified research in terms of classification and detection. In our model, we also combine VGG16 with additional transfer learning methods. To extend the samples of the dataset and limit the danger of over-fitting, data augmentation methods are applied to the MRI slices for generalization of results.[6]. In our dataset, we also employ the data augmentation approach. The solution offered by Ammar Adl and Neama Younan was 92.8 % accurate. In this research, they used a Gaussian filter to reduce noise in photos, then used the GrabCut method to separate the backdrop from the item (*Ichthyophthirius multifiliis* parasite). To reduce noise in our model, we employ the data augmentation strategy. Then, using the GrabCut method, they isolated the backdrop to retrieve only the item (*Ichthyophthirius multifiliis* parasite). To reduce noise in our model, we employ the data augmentation strategy. Then, using the ORB algorithm for feature extraction, they extracted 4.0 features for the Ich disease, allowing the machine to understand what is infected and what is not. They next employed classification algorithms to train and assess the images, achieving excellent accuracy with the Logistic Regression Classifier.[7]. However, in our thesis, we employed the

CNN transfer learning model, which can automatically extract features from data. Jeong-Seon Park, Myung-Joo Oh, and Soonhee Han also demonstrated various picture feature extraction approaches. Using several feature extraction approaches, they achieved the greatest accuracy of 90% and were able to categorize microscopic images of fish disease.[8]. Although much work has been done in the field of leaf disease detection, it remains tough and difficult. And only a few works have been done in this difficult field, but in this thesis, we attempt to tackle this difficult issue in an effective manner by employing the deep learning approach.

2.3 Research Summary

The purpose of this project is to use Tensorflow and Keras, an open-source dataflow and machine learning framework, to develop an image classification Convolutional Neural Network (CNN) using transfer learning for the classification of the leaf picture. A few years ago, machine learning techniques were first utilized for image-based categorization. Convolutional Neural Networks may be used to educate robots to comprehend images. There are several projects in this field that are being carried out in various ways. We can acquire better insights by looking at the background of machine learning and related research. After testing, questioning, analyzing, and comparing all relevant studies on tea leaf disease detection, we arrived at a variety of results. To begin with, none of them employ transfer learning, which is pretty outstanding, especially when the dataset is tiny, and many of them do not employ data augmentation techniques to improve photos, which are superior to other image improving approaches. The second factor is accuracy rate; most projects accomplish up to 96% of the claimed accuracy rate. We can now claim that there have been some studies on tea leaf disease detection, and we presented a novel technique that can categorize tea leaf diseases more effectively and reliably.

2.4 Scope of the Problem

There is a lot of room for the issue in every phase of our task. It is impossible to identify difficulties before we begin a project, but we do so as the project proceeds. The most challenging aspect of our research was data collection. We must change the data in some way after collecting it and before preparing it. Data gathering, the most challenging aspect of working is programming syntax, model selection, and so on.

2.5 Challenges

In order to complete this project, we had to overcome a number of difficulties. Data collection was the most difficult aspect of our project. This is the first time in our country that we have to devote much more effort to collecting tea leaf photographs. We had a more difficult time gathering data during the outbreak. During pandemics, we go to Kazi & Kazi Tea Estate Ltd. in Panchagarh to collect data. It was the most difficult period. The next great problem is to identify or create a model with several layers. Defining the training set, preventing overfitting, choosing the number of epochs, and batch size are more difficult for us. A similar leaf image was also a project challenge. To look at, several leaf images are the same. It is really difficult to locate the sickness simply by looking around the outside of the body. It was quite difficult to deliver precise findings at the time. We are conducting research to find a solution to this problem. After that, we need a strong GPU to train this large number of parameters, since training a model with a GPU takes relatively few times, while training a model without a GPU takes a very long time.

CHAPTER 3

Research Methodology

3.1 Introduction

In chapter three, we will detail our entire working technique. Our working technique has been broken down into numerous phases. First, we'll go through data collection and pre-processing. Then we talk about how we use the Transfer Learning model of the Deep Learning idea, which is the most recent and widely used approach for image categorization. In the last section, we will go over the training and test sets, input and output, for each convolutional layer, implementation requirements, and so forth. We shall provide a perfect example to clarify the concept.

3.2 Research Subject & Instrumentation

The study subject provides a clear notion of our research field. The goal of this study is to distinguish between diseased and healthy leaf types. In this section, we present and design our model, collect perfect data, prepare and train our model, discuss our findings, and finally put our model to work. We present a transfer learning model capable of picture classification utilizing both Tensorflow and Keras. We also make use of packages such as numpy, matplotlib, and others. Our job is carried out on the Windows platform. We'll need certain instruments to run this experiment on any platform. The following instruments and requirements are listed:

1. Python Programming Language
2. Google Colab

3. Tensorflow and KerasAPI

We chose Python because it is free and simple to use, and it can be used for sophisticated algorithm readability in machine learning applications. In addition, we use Google Colaboratory, a free online cloud-based Jupyter notebook environment that we can simply employ to accomplish our project.

3.3 Data Collection Procedure

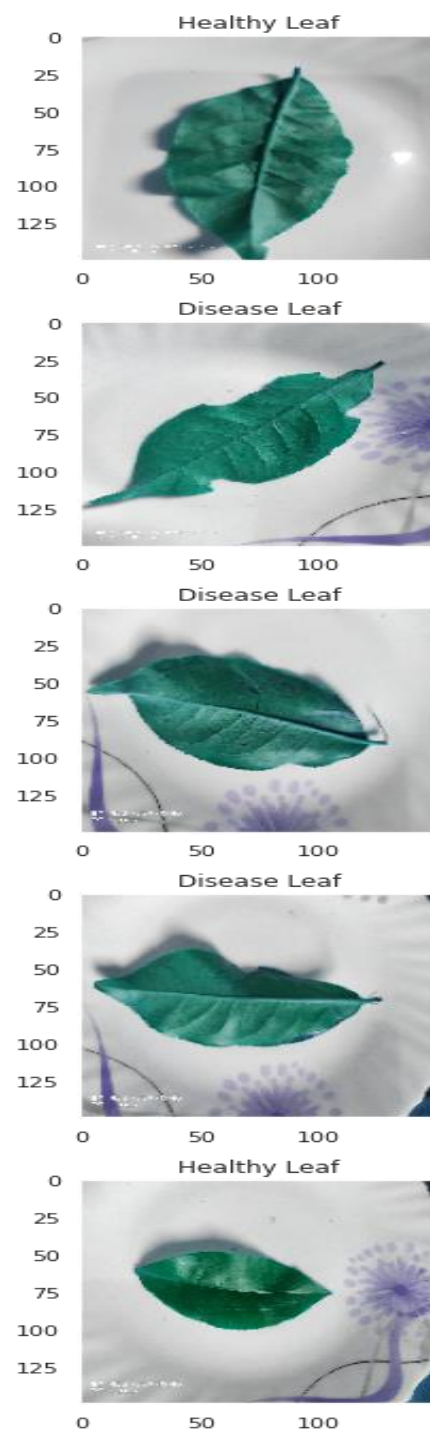
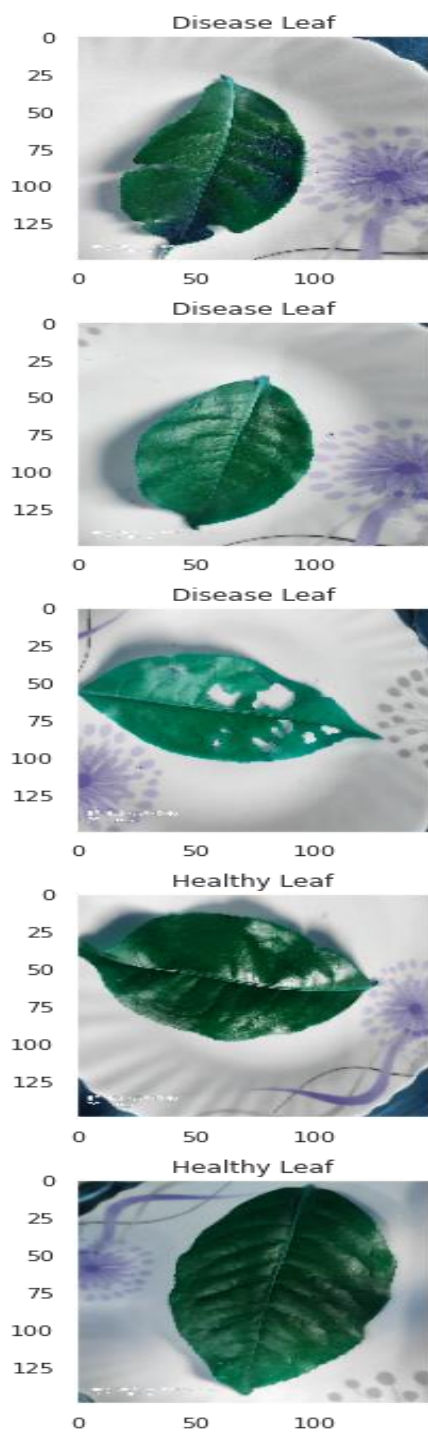
Because the detection of tea leaf diseases by images is not yet a well-studied topic, we could not find any datasets on any of the sites. In reality, there is no accurate dataset on this issue. For this research, we had to develop a new, authentic collection of leaf disease images. So we decided to visit Kazi & Kazi Tea Estate Ltd. in Panchagarh to gather information. They provided us with some tea-related facts.

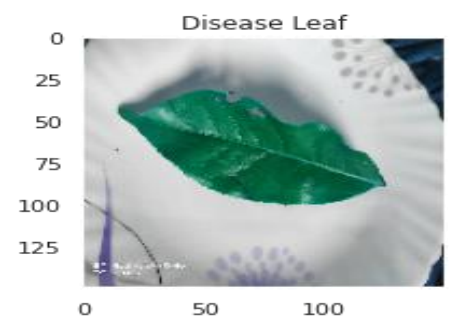
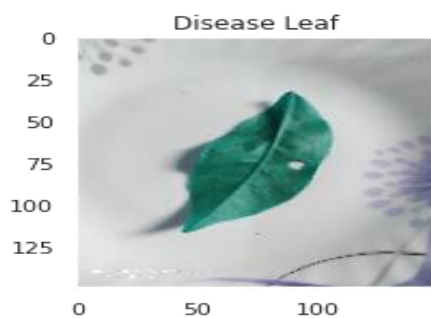
We chose a variety of leaf photos. All photographs are in JPG and PNG format, with different resolutions. There are a total of 2000 images, divided into two categories. The one on the left is a diseased leaf, while the one on the right is a healthy leaf.

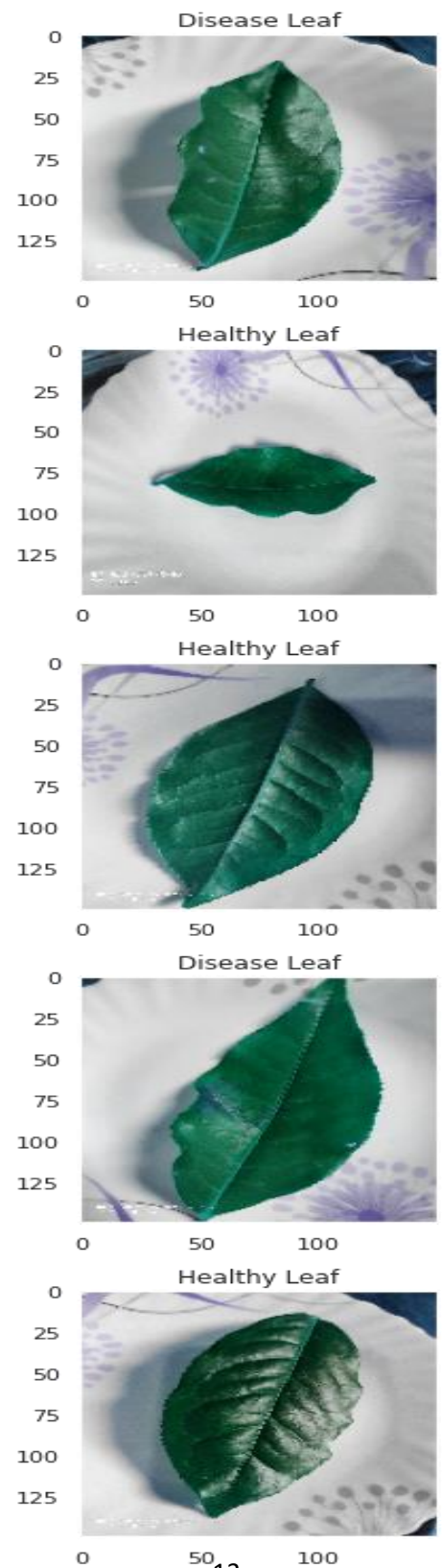
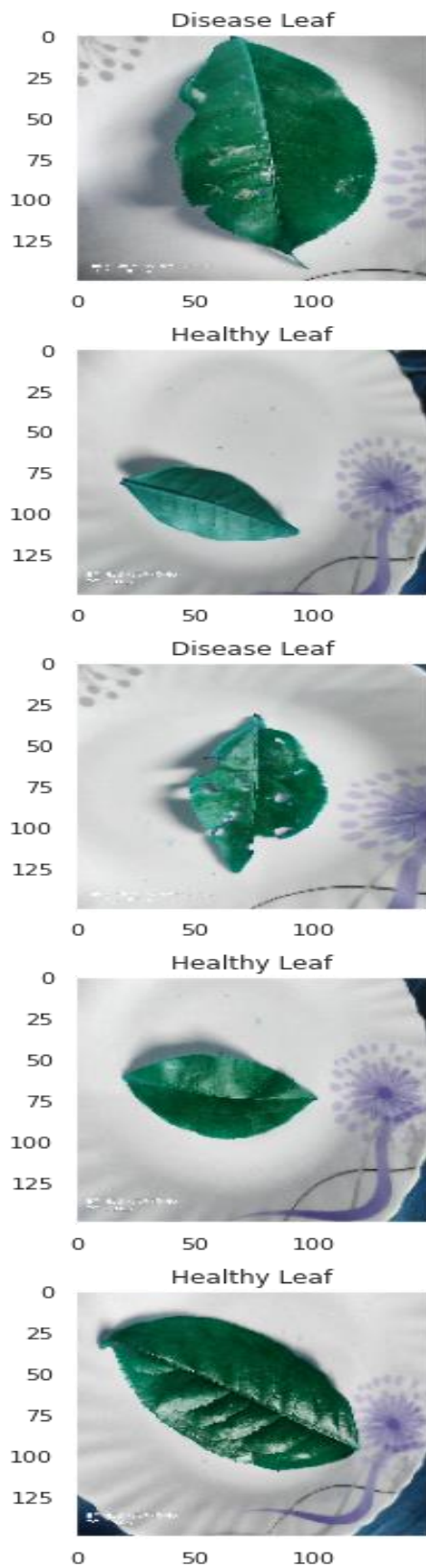
There are a total of 2,000 leaf photos in our collection. Among the 2000 images, 1000 are of diseased leaves and 1000 are of healthy leaves. 75% of the 2000 data points were utilized for training, while the remaining 25% were used for testing. The initial stage in developing our thesis was to collect that data.

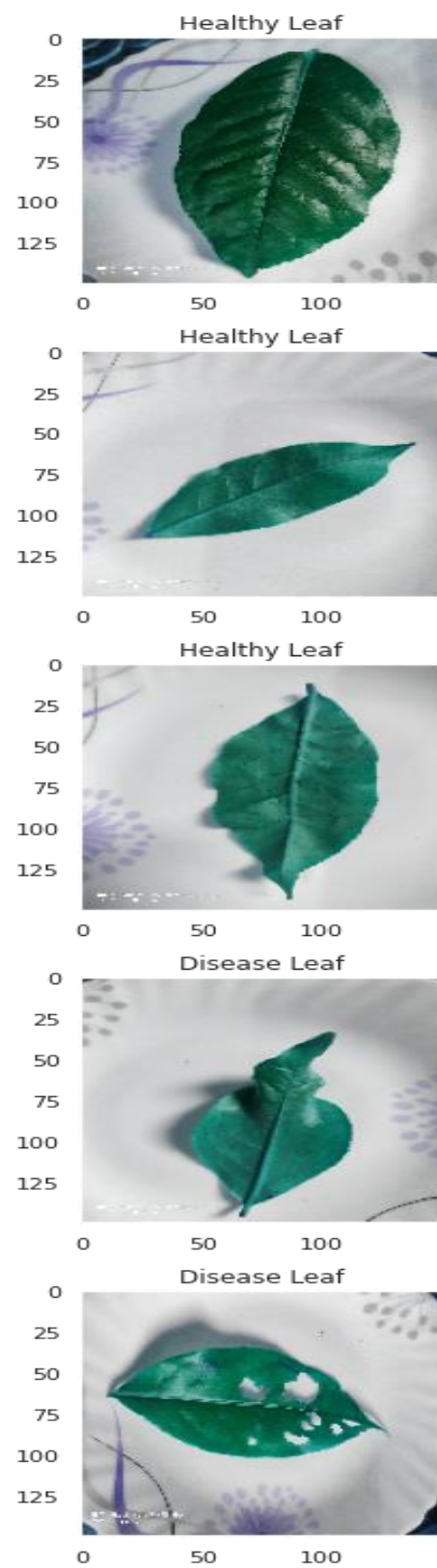
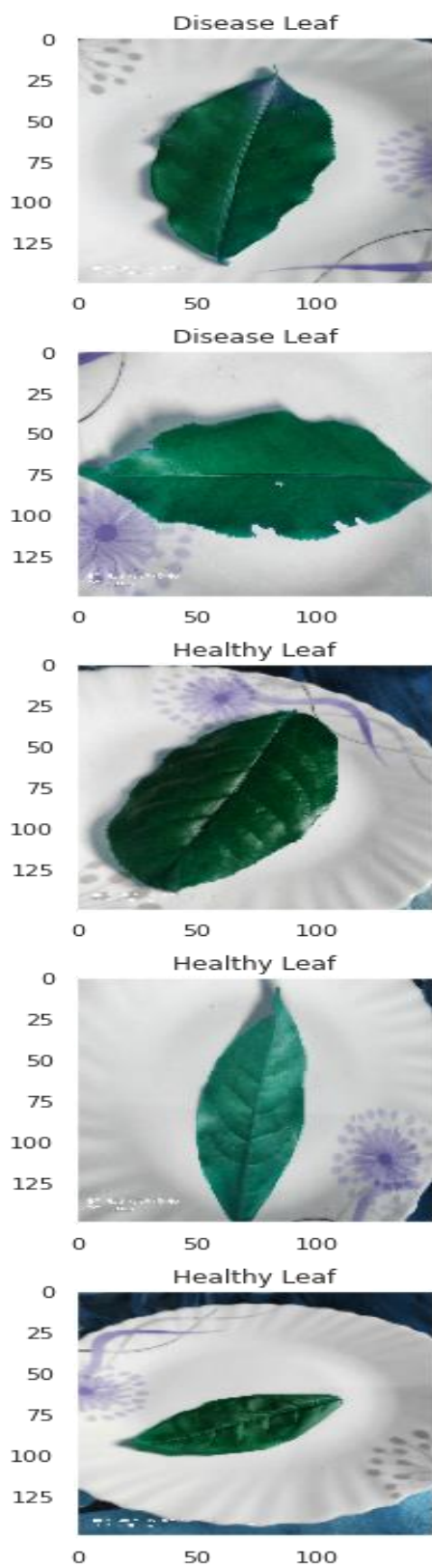
We simply collected raw data from the tea garden.

The raw dataset of tea leaf disease images is shown in the figure below.









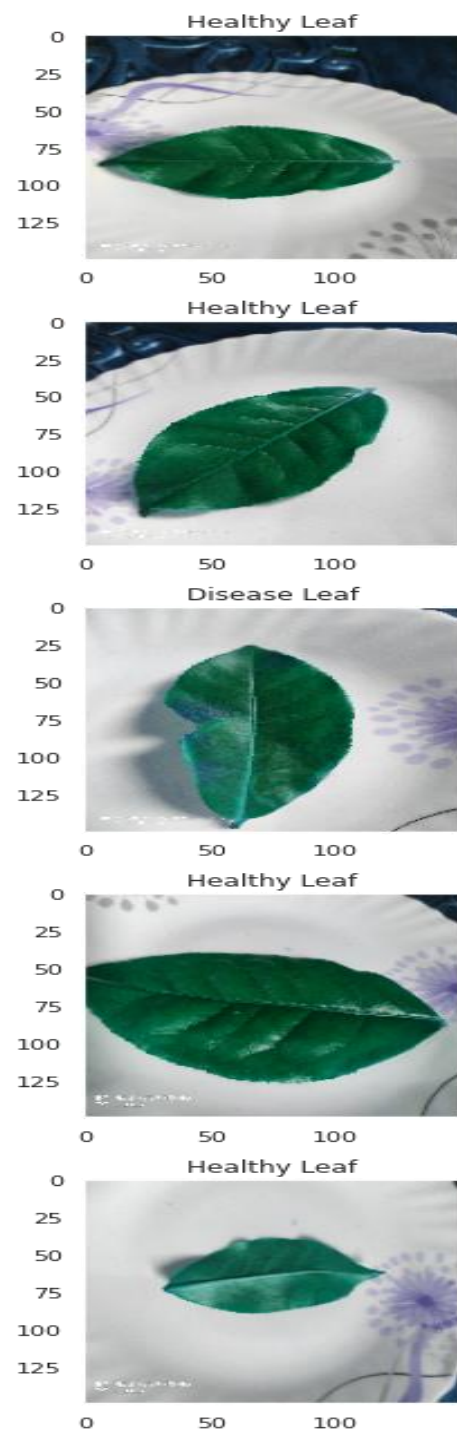
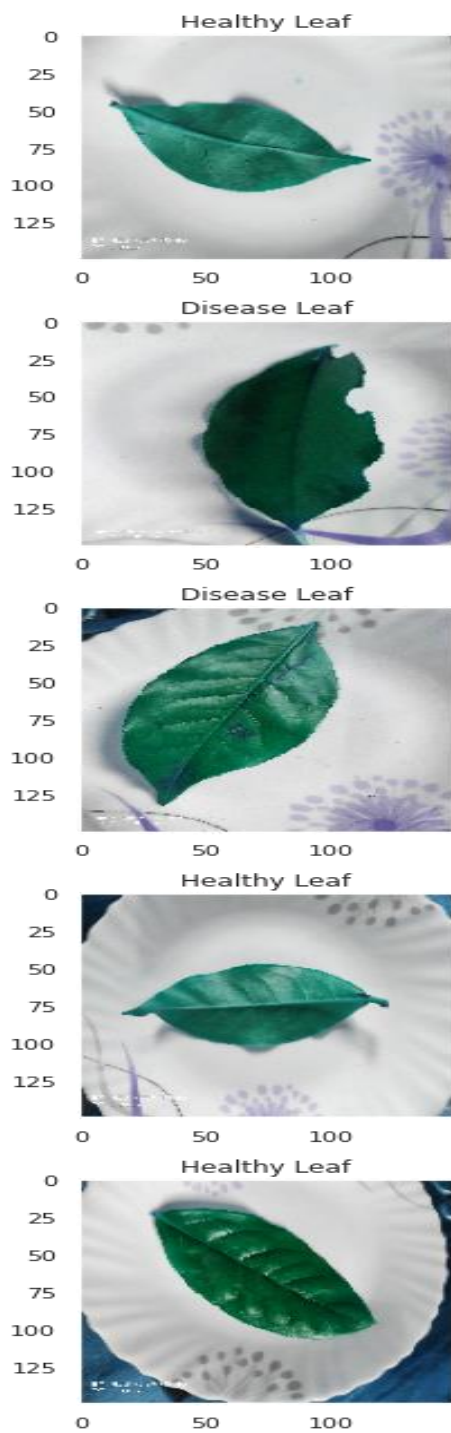


Figure 1.0: A small part of the raw dataset

3.4 Proposed Methodology

To properly create the model, the entire procedure is divided into a number of sub-processes. The full procedure is depicted in Figure 1.2.

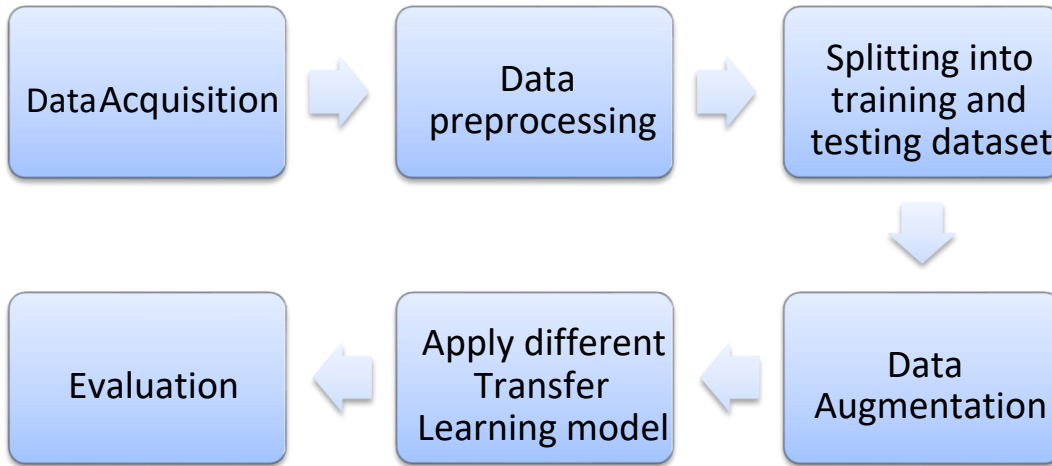


Figure 1.1: Block diagram of proposed methodology

Because we have so many varied shapes and sizes of photos, we must first gather the data and then process the dataset. After that, we divided our dataset into train and test sets. Then, to improve our images, we employ the data augmentation approach. Then we use a variety of transfer learning methods. We apply six different types of transfer learning algorithms in this case.

These are as follows:

- 1) VGG16
- 2) Xception
- 3) InceptionV3
- 4) MobileNetV2
- 5) InceptionResNetV2
- 6) MobileNet

We use the same parameter in all of these models. We utilize GlobalAveragePooling2D in all of our models to eliminate extraneous features while keeping the vital features in our photos. Using this, it selects only the average value from the pooling window. Dropout is used in our model to prevent overfitting. In this case, we use a dropout value of 0.2. Because this is a binary classification, we utilize a sigmoid optimizer. We also set layer.trainable to false in order to freeze the layer. By setting this to false, all of the layer's weights are converted from trainable to non-trainable. Before training the model, we employ a label encoder. We trained all transfer learning models using the Adam optimizer and the binary crossentropy loss function to lower the loss function as much as possible, and we applied it to 75% of the training set and 25% of the test set. We obtained the best results from MobileNet after applying those transfer learning techniques. Other transfer learning methods also produce satisfactory results. Then we assess our model. In the next chapter, we will go deeper into the entire process, from preprocessing through the use of several transfer learning techniques. The evaluation phase will be shown in the next chapter.

3.4.1 Data Preprocessing

First and foremost, we collect data. However, the data was disorganized. Images differ in size and resolution from one another. As a result, we must prepare them before sending them to model training. Data processing is critical for model training and improving accuracy since it lowers computer costs, among other things. Our photos are resized to 150x150 pixels. The data is then shuffled to avoid overfitting our model. In addition, we organized our data into an array list. We reduce RGB values by dividing by 255 to normalize our data. Then, to improve our picture data, we employ some sophisticated approaches.

3.4 .2 Data Augmentation

Data augmentation is a technique that may be used to dynamically increase the size of a training dataset by introducing new updated model versions into the dataset. This

strategy increases the diversity of data accessible for deep learning training models without necessitating the collection of fresh data. It improves the model's performance and decreases classification loss. We also employ the augmentation approach to improve the quality of our data. These are as follows:

- Rotate the picture by a factor of 40.
- Zoom in the range of 0.01
- width shift in the range of 0.2
- height shift in the range of 0.2
- horizontal flip as True

This data augmentation strategy is used by our suggested model to determine the optimal accuracy.

3.4.3 Convolutional Neural Network

Convolutional Neural Network (CNN) is a deep learning algorithm that accepts input photos and may apply learnable weights and biases to distinct objects in the image to distinguish one from the other. In general, the operation of a CNN is dependent on inputs for pattern recognition. CNN is a type of machine learning network that is widely used to solve image visualization challenges such as classification.[8]. It applies a collection of convolutional filters to the data input to get the network's learning parameters. Pooling layers are used to decrease the number of learning parameters needed and the computing requirements by remaining between convolutional layers. Finally, completely linked layers establish entire connections to the prior layer, and the final output is obtained.

Figure 1.3 depicts a high-level overview of CNN architecture.

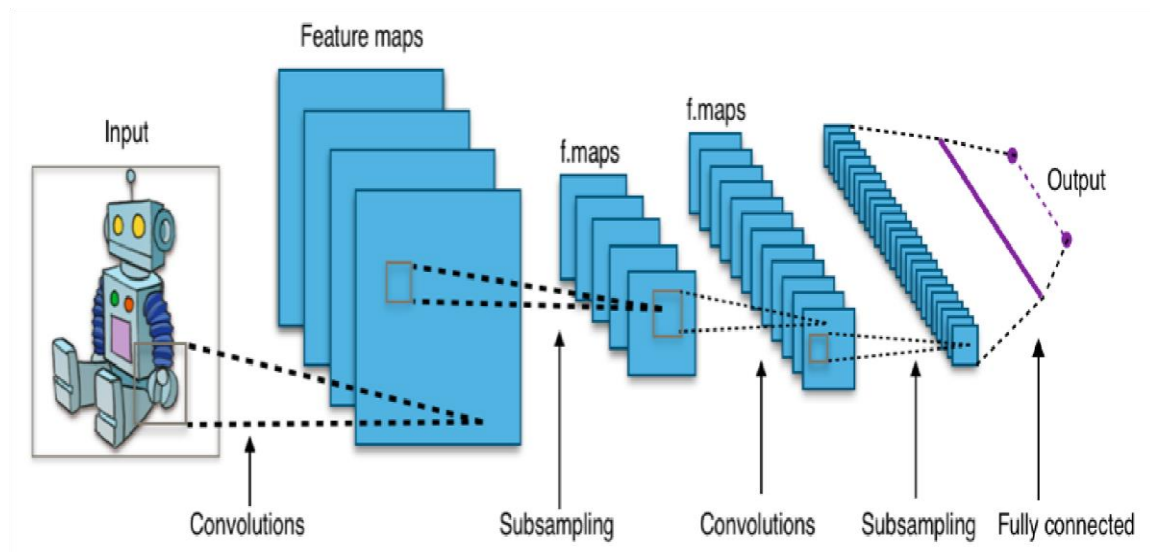


Figure 1.2: A typical CNN architecture

We utilize a transfer learning technique in this study to categorize photos of diseased and healthy leaves. Transfer learning is the process of increasing learning by transferring knowledge from a previously learned, similar activity to a new task. It is particularly useful since it allows the reuse of a previously trained model on a new issue, which is especially useful when the dataset is tiny. Instead of developing a fresh CNN with random parameter initialization, we followed a pre-trained CNN and fine-tuned its parameterization for our specific area of classification. There are several transfer learning models available. However, we only employ six transfer learning techniques in this case.

VGG16: VGG-16 is an Oxford University Visual Geometric Group composition, and VGG-16 is a 16-layer network suggested by the Visual Geometric Group[9]. VGG16 is a CNN convolution architecture that won the ILSVR (Imagenet) competition in 2014. The most remarkable aspect of VGG16 is that, rather than creating a huge number of hyper-parameters, they have focused on having 3x3 filter convolution layers with a stride of 1 and the same padding and max-pooling layer as a 2x2 filter of stride 2. This network is rather large, with parameters of roughly 138 million (approx). We'll go through the VGG-16's design in depth here.

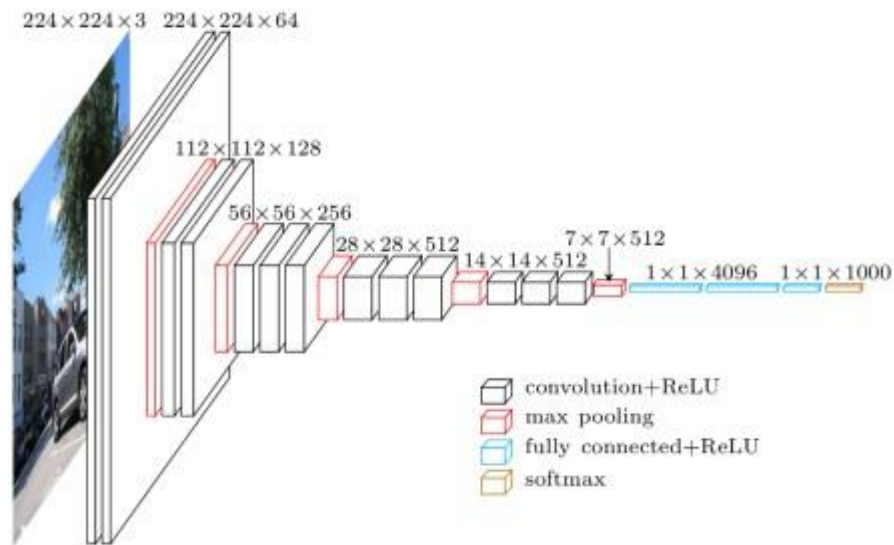


Figure 1.3: VGG16 architecture

The convolution layers are represented by all of the blue rectangles and the non-linear activation function, which is a rectified linear unit in the preceding diagram. In this design, we see it start with a relatively small channel size of 64 and then steadily expand by a factor of 2 after each max-pooling layer until it reaches 512. The architecture is straightforward. We can make this model work for any number of classes by changing the final softmax dense layer unit to whatever number we choose based on the classes we need to categorize.

InceptionV3: Inception-v3 is a convolutional architecture of the Inception family neural network that improves on previous versions by employing Label Smoothing. An auxiliary classifier was employed to factorize 7 x 7 convolutions. This is the third version of Google's Inception Convolutional Neural Network, which was first used in ImageNet Recognition. Inception V3 was learned using a dataset of 1,000 classes from the original

ImageNet dataset, which was trained with over 1 million training pictures. In this section, we'll go over the architecture of the InceptionV3.

The Inception v3 network stacks 11 inception modules in the aforementioned design. Each module, in this case, is made up of pooling layers and convolutional filters using rectified

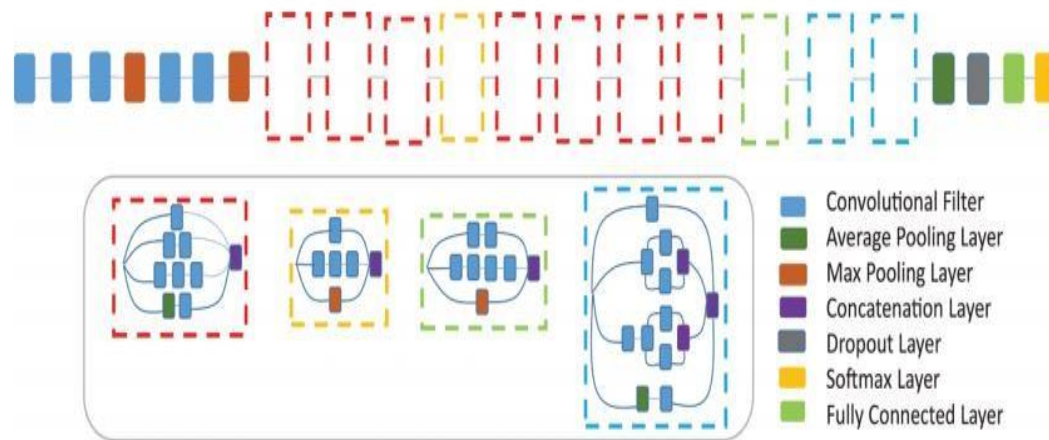


Figure 1.4: InceptionV3 architecture

linear units as activation features [19]. This model input consists of two-dimensional photographs of 16 horizontal brain slices positioned on 4x4 grids obtained during the preprocessing step. Three completely linked layers of sizes 1024, 512, and 3 are appended to the final concatenation layer [19]. A dropout with a rate of 0.6 is performed before the completely linked layers as a regularization technique. The model is pre-trained on the Image-Net dataset and is perfectly alright with a batch size of 8 and a learning rate of 0.0001.

MobileNetV2: MobileNetV2's architecture is entirely based on a reversed residual structure, with the residual block's input and output being thin bottleneck layers as opposed to conventional residual models that use extended input representations. MobileNetV2 filters intermediate expansion layer features using lightweight depth convolutions [11]. This is mostly a V1 improvement that makes it even more effective and productive. This module may be easily deployed in any modern environment

using standard operations and helps our models beat the state of the art using thereof across several performance points [19]. The second new feature of MobileNetV2's construction block is the residual link, which is utilized only when the number of channels coming into the block equals the number of channels leaving it, which is not always the case because the output channels are increased every few blocks. We'll go through the MobileNetV2 architecture in depth here.

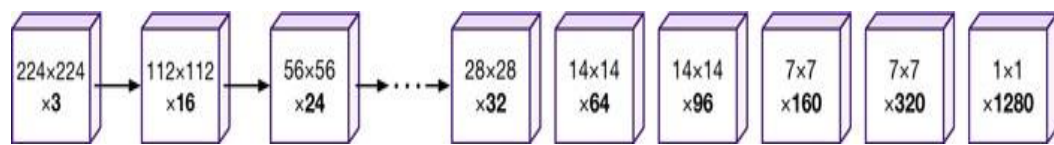


Figure 1.5: MobileNetV2 architecture

Looking at the data as it travels through the network in the above design, we can see how the number of channels remains very minimal between the blocks. The number of channels will grow with time, as is typical for this type of architecture, but the tensors will remain relatively small due to the bottleneck layers that make up the connections between the blocks [11]. The aim behind using low-dimension tensors is to reduce the amount of calculations. We won't be able to extract much information by filtering a low-dimensional tensor using a convolutional layer. MobileNetV2's block design gives us the best of both worlds.

Xception: The Xception architecture has 36 convolutional layers that serve as the network's feature extraction base. Because the Xception architecture has the same number of parameters, the performance increases are attributable to more successful utilization of model parameters rather than higher intensity [13]. In Xception, the updated depth-wise separable convolution, there is NO intermediary ReLU non-linearity. The photos for this model are 299×299 in size by default [14]. We will discuss the architecture of the Xception in details here,

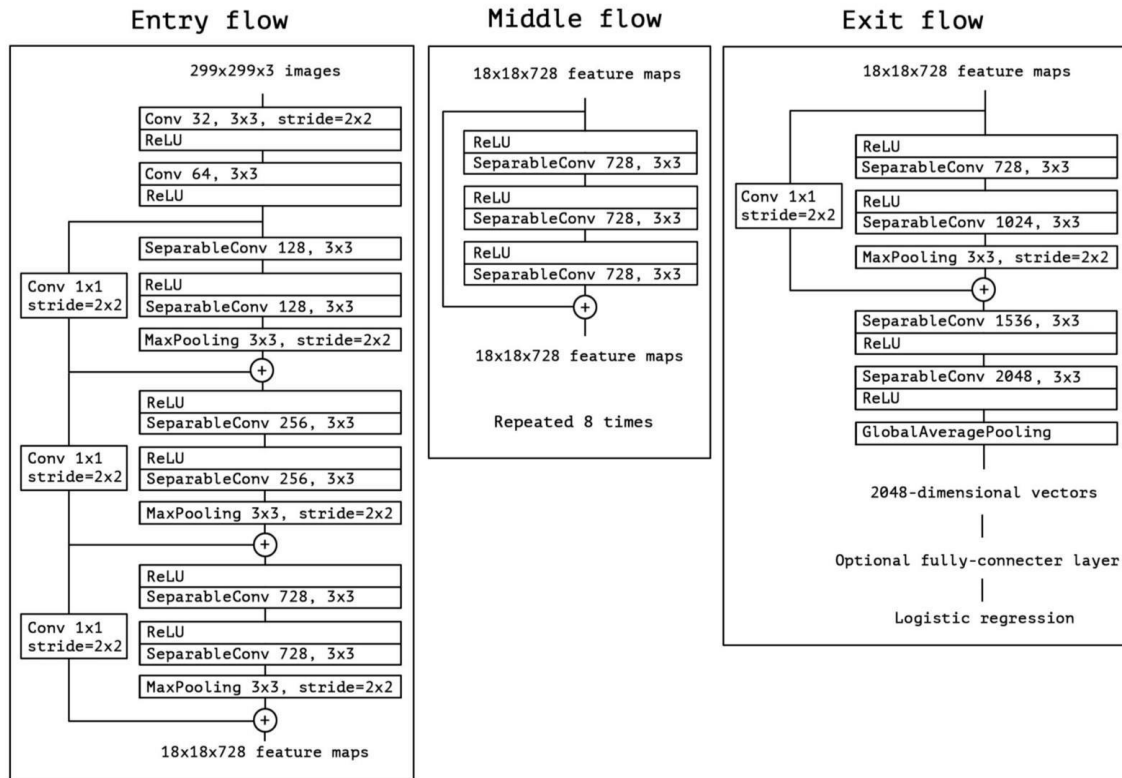


Figure 1.6 : Xception architecture

The design of this convolutional neural network is solely based on depth-wise differentiated convolution layers. The data first passes via the entering flow, then through the middle flow, which is repeated eight times, and lastly, through the exit flow. Batch normalization is followed by convolution and separable convolution layers, which are not shown in the picture above. A depth multiplier of one is utilized for each separable convolution layer. The Separable Convolution layer is the updated depth of separable convolution [14]. Xception is running the non-residual version. The accuracy is substantially higher when employing residual connections, as shown in the preceding image [14].

InceptionResNetV2: In InceptionResNetV2, Inception and ResNet integrate the two designs to improve performance even more. InceptionResNetV2 is a neural convolutional

network that is based on the Inception architecture family but contains residual ties [19]. Residual linkages provide model shortcuts and have assisted researchers in effectively training deeper neural networks, resulting in even better outcomes. It also significantly lowers training time.

In this section, we'll go through the design of the InceptionResNetV2.

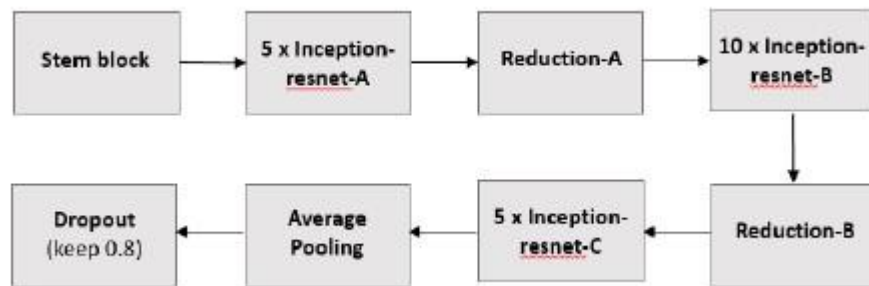


Figure 1.7: InceptionResNetV2 architecture

As far as we know, this CNN model has been trained on over 1 million photos from 1000 categories in ImageNet. Because they are trained on a very large labeled dataset, these deep CNN are capable of learning generic picture properties that are crucial to other image qualities. It is also known that Inception-ResNetV2 models may achieve greater accuracy with less epochs. The input size will also be 299x299 in this case. After everything has been said and done, the model was not trained on a single GPU using Batch Normalization [19].

MobileNet: MobileNet is a model that is primarily based on depth-wise separable convolutions, which were first revealed and then used in Inception models to reduce the amount of computation required in the first few layers [12]. Flattened networks build a network out of completely factorized convolutions and demonstrate the power of extraordinarily factorized networks. In addition to the usage of topological networks,

factorized networks introduce a related factorized convolution. The Xception network gradually learned how to scale up depth-wise separable filters to outperform the Inception V3 networks [13]. Another tiny network is the squeeze net, which employs a bottleneck mechanism to create a very compact network. Arranged transform networks and deep-fried convnets are two further types of reduced computation networks.

We'll go through the MobileNet's design in depth here.

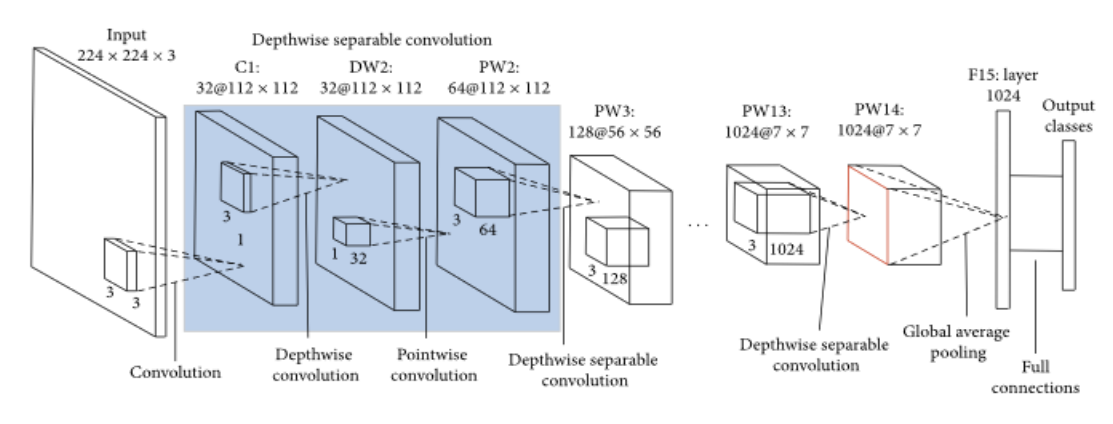


Figure 1.8: MobileNet architecture

MobileNets are constructed using depth-wise separable convolution layers. Each depth-wise separable convolution layer includes both a depth-wise and a point-wise convolution. MobileNets are constructed using depth-wise separable convolution layers. Each depth-wise separable convolution layer includes both a depth-wise and a point-wise convolution. A MobileNet contains 28 levels if depthwise and pointwise convolutions are counted as separate layers. A basic MobileNet contains 4.2 million parameters, which can be lowered further by fine-tuning the width multiplier hyperparameter. The supplied picture has a size of $224 \times 224 \times 3$ [12].

3.4.4 Train the model

After producing data pre-processing and establishing the train and test datasets, we are ready to train our model, which contains 75% of the total dataset. We alter the optimizer,

loss function, number of epochs, and so on to maximize accuracy and minimize loss. We used 128 batch sizes to save memory and speed up the training of our model. To train our models, we use 100 epochs. Before training, we also employed a label encoder. We employ data augmentation in the training process to adjust to the data augmentation. After everything was in place, we began training our models.

3.5 Implementation Requirements

A mandatory listing for this image-based categorization was generated when our CNN approach was completely specified and our model was entirely trained.

- Ram (Minimum 4GB)
- Operating System (Windows 7 or above)
- Hard Disk (Minimum 500GB)
- GPU (Recommended)

We should additionally employ the following development tools:

- Google colab
- Python Environment

CHAPTER 4

Experimental Results and Discussion

4.1 Introduction

In Chapter 4, we explore our model's overall performance evaluation. We will also go through the amount of parameters, the level of precision, and so on. Then we'll evaluate several transfer learning models such as VGG16, Xception, InceptionV3, MobileNetV2, InceptionResNetV2, and MobileNet. For simple underpinning, we utilize several graphs, graphics, a confusion matrix, and classification reports.

4.2 Experimental Setup

Various models were presented in order to discover the optimum model appropriate for the detection of tea leaf diseases. Each model's accuracy, loss, precision, recall, and f1-score were examined after it was applied. The confusion matrix was displayed for all of the models to illustrate the performance of each model. The following formulae are used to calculate the accuracy, precision, recall, and f1-score.

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN})$$

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP})$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN})$$

$$\text{F1 Score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

Where TP denotes True Positive, FP denotes False Positive, TN is True Negative, and FN denotes False Negative.

After that, we apply macro average and weighted average to get the overall worth of these models. The model with the highest accuracy was eventually chosen as the best model.

4.3 Experimental Results & Analysis

Table 1.1 shows an alternative convolutional neural network architecture that we employed in this research, with varying numbers of total parameters, trainable parameters, training data accuracy, training data loss, testing data accuracy, and testing data loss during the model experiences training process.

Table 1.1: Experimental Result of different neural network model

Model	Total Parameters	Trainable Parameters	Train Accuracy	Train Loss	Test Accuracy	Test Loss
MobileNet	3229889	1025	0.9504	0.1227	0.9820	0.0701
MobileNetV2	2259265	1281	0.9300	0.1870	0.9620	0.1290
InceptionV3	21804833	2049	0.9155	0.2205	0.9180	0.2214
InceptionResNetV2	54338273	1537	0.8448	0.3414	0.8740	0.2814
VGG16	14715201	513	0.8345	0.4187	0.8540	0.3751
Xception	23589761	2049	0.6086	0.6649	0.5780	0.6582

Now we'll talk about the graph, the confusion matrix, and the classification report for the various models we employed.

MobileNet

Of all the models we've tried, MobileNet has the best accuracy and lowest loss. Figure 1.9 depicts the model's training and testing accuracy as well as loss.

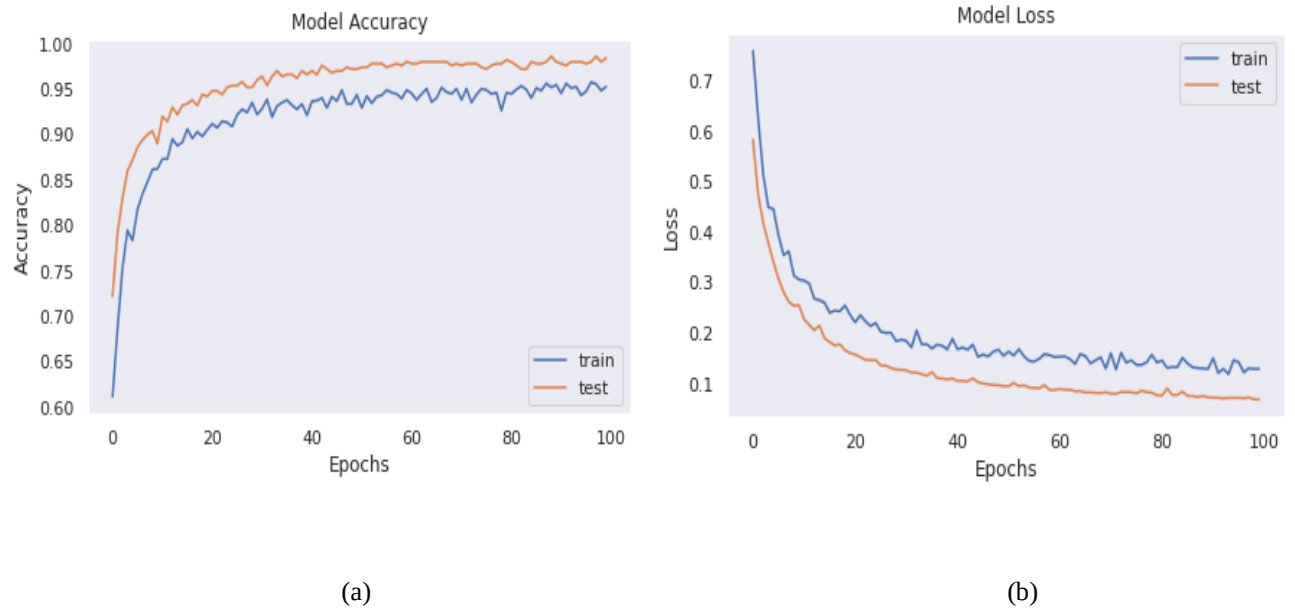


Figure: 1.9: (a) MobileNet model accuracy; (b) MobileNet model loss

The MobileNet model's confusion matrix is shown in Figure 2.1.

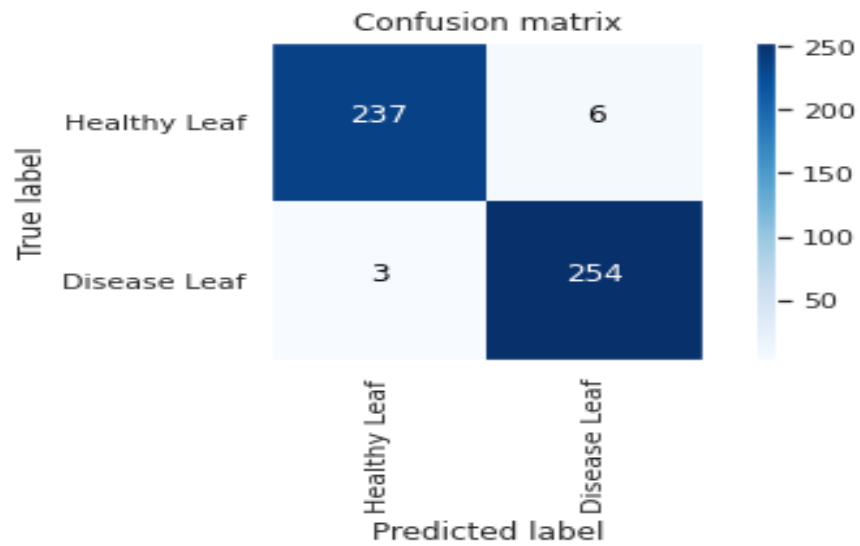


Figure: 2.1: Confusion matrix of MobileNet model

Table 1.2 shows the MobileNet architecture of the neural network we used in this project, including accuracy, recall, f2-score, and support, among other things.

Table 1.2: Classification report of MobileNet model

	Precision	Recall	F1- Score	Support
Healthy Leaf	0.99	0.98	0.98	243
Disease Leaf	0.98	0.99	0.98	257
Accuracy			0.98	500
Macro avg	0.98	0.98	0.98	500
Weighted avg	0.98	0.98	0.98	500

MobileNetV2

Figure 2.2 (a) depicts the model's training and testing accuracy, whereas (b) depicts the model's training and testing loss.

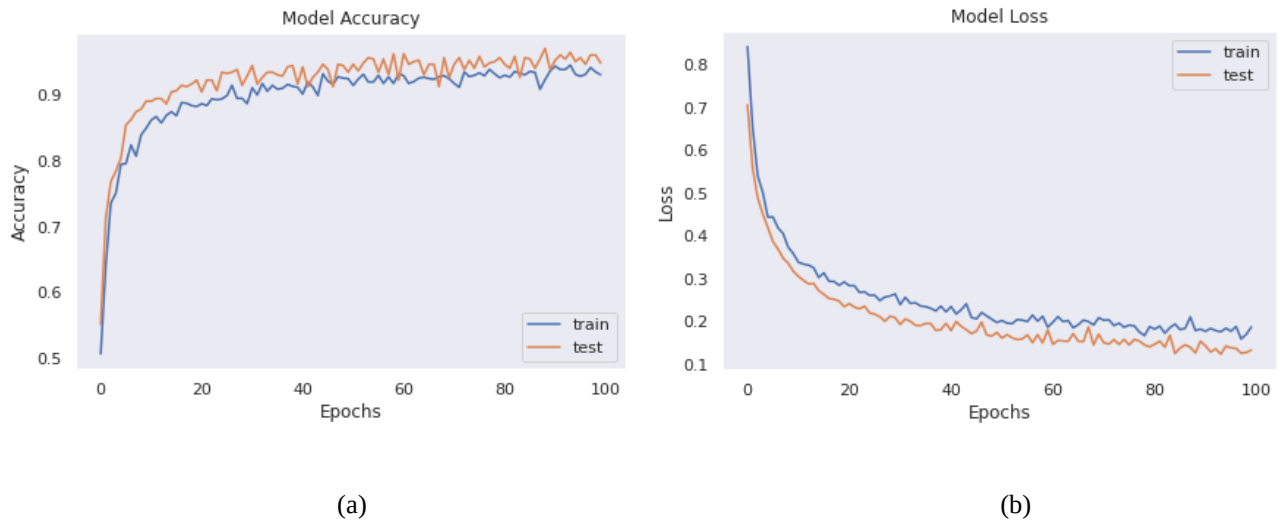


Figure: 2.2: (a) MobileNetV2 model accuracy; (b) MobileNetV2 model loss

The MobileNetV2 model's confusion matrix is shown in Figure 2.3.

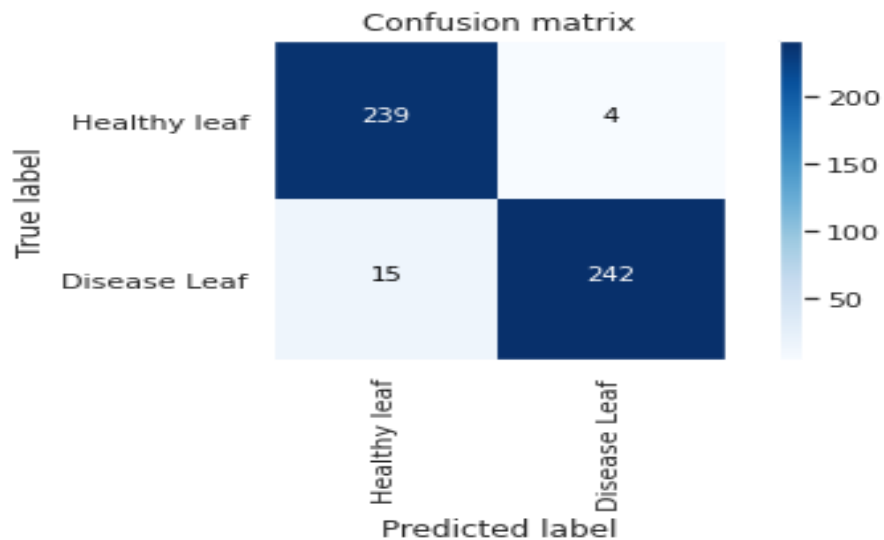


Figure: 2.3: Confusion matrix of MobileNetV2 model

The MobileNetV2 architecture of the neural network that we utilized in this research is shown in Table 1.3, along with accuracy, recall, f2-score, and support.

Table 1.3: Classification report of MobileNetV2 model

	Precision	Recall	F1- Score	Support
Healthy Leaf	0.94	0.98	0.96	243
Disease Leaf	0.98	0.94	0.96	257
Accuracy			0.96	500
Macro avg	0.96	0.96	0.96	500
Weighted avg	0.96	0.96	0.96	500

InceptionV3

Figure 2.4 (a) depicts the model's training and testing accuracy, whereas (b) depicts the model's training and testing loss. .

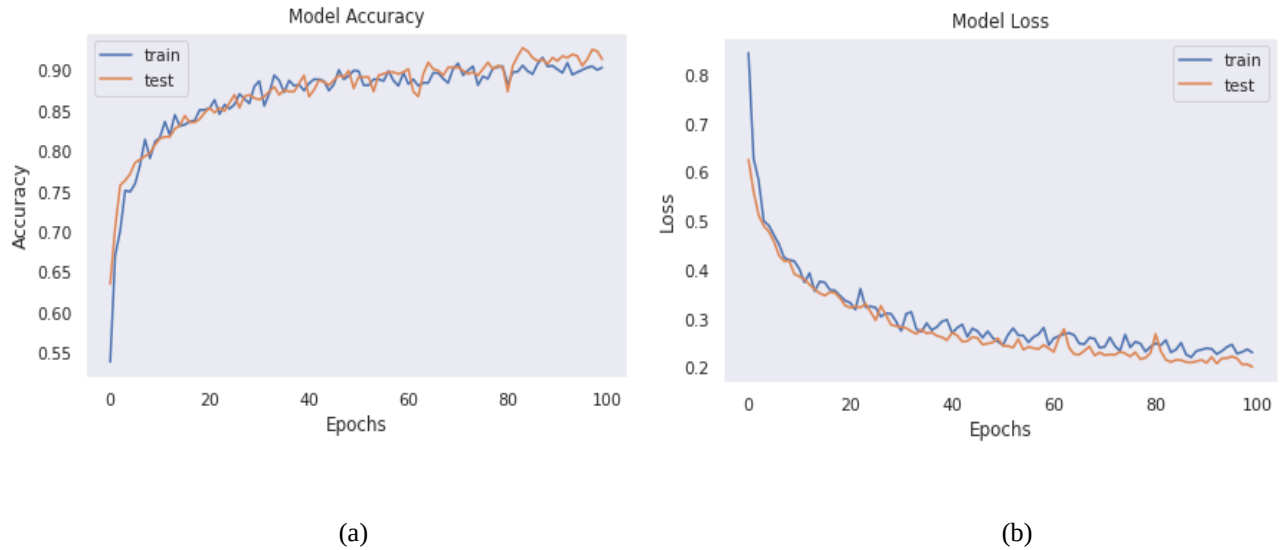


Figure: 2.4: (a) InceptionV3 model accuracy; (b) InceptionV3 model loss

The InceptionV3 model's confusion matrix is shown in Figure 2.5.

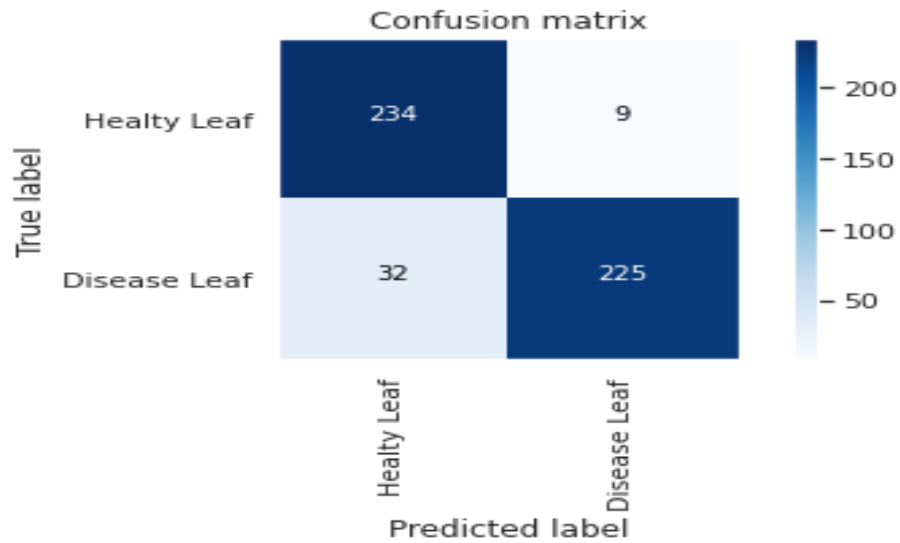


Figure: 2.5: Confusion matrix of InceptionV3 model

Table 1.4 shows the InceptionV3 architecture of the neural network that we utilized in this research, including accuracy, recall, f2-score, and support, among other things.

Table 1.4: Classification report of InceptionV3 model

	Precision	Recall	F1- Score	Support
Healthy Leaf	0.88	0.96	0.92	243
Disease Leaf	0.96	0.88	0.92	257
Accuracy				0.92
Macro avg	0.92	0.92	0.92	500
Weighted avg	0.92	0.92	0.92	500

InceptionResNetV2

Figure 2.6 depicts the model's training and testing accuracy and loss.

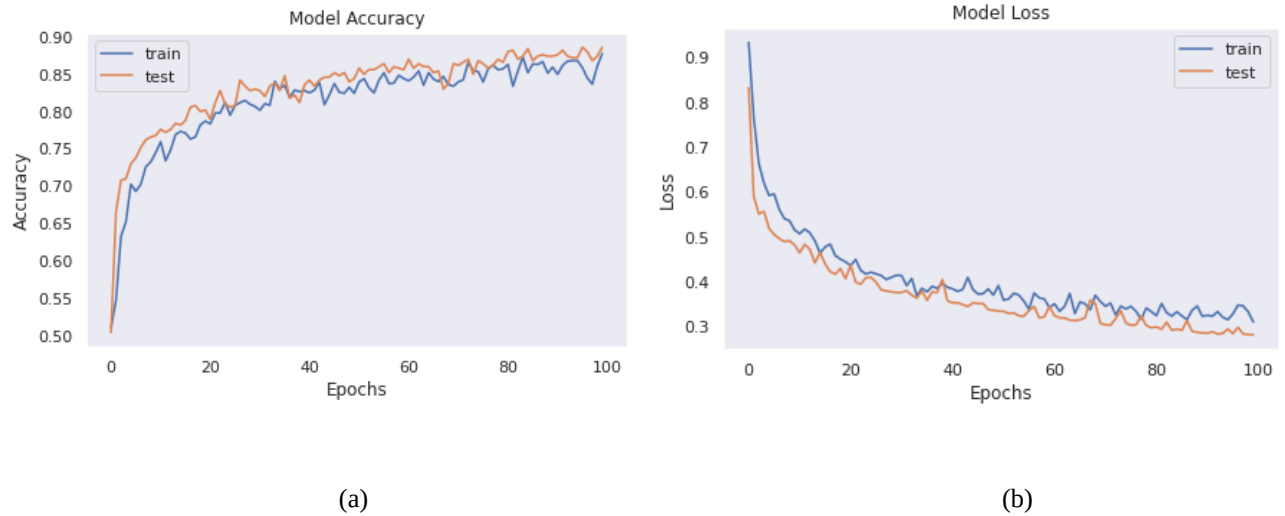


Figure: 2.6: (a) InceptionResNetV2 model accuracy; (b) InceptionResNetV2 model loss

The InceptionResNetV2 model's confusion matrix is shown in Figure 2.7.

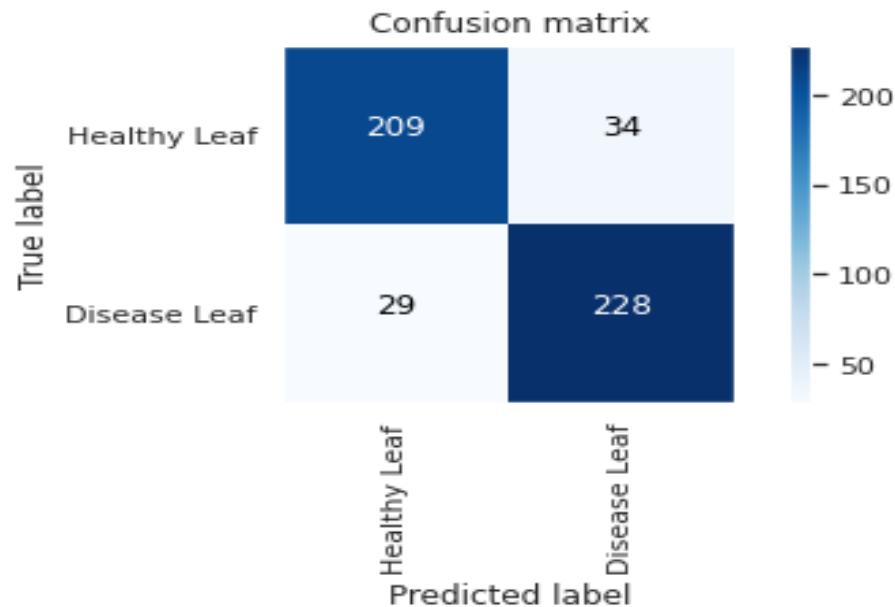


Figure: 2.7: Confusion matrix of InceptionResNetV2 model

Table 1.5 shows the InceptionResNetV2 architecture of the neural network that we utilized in this research, including accuracy, recall, f2-score, and support, among other things.

Table 1.5: Classification report of InceptionResNetV2 model

	Precision	Recall	F1- Score	Support
Healthy Leaf	0.88	0.86	0.87	243
Disease Leaf	0.87	0.89	0.88	257
Accuracy			0.87	500
Macro avg	0.87	0.87	0.87	500
Weighted avg	0.87	0.87	0.87	500

Xception

Figure 2.8 (a) depicts the model's training and testing accuracy, whereas (b) depicts the model's training and testing loss.

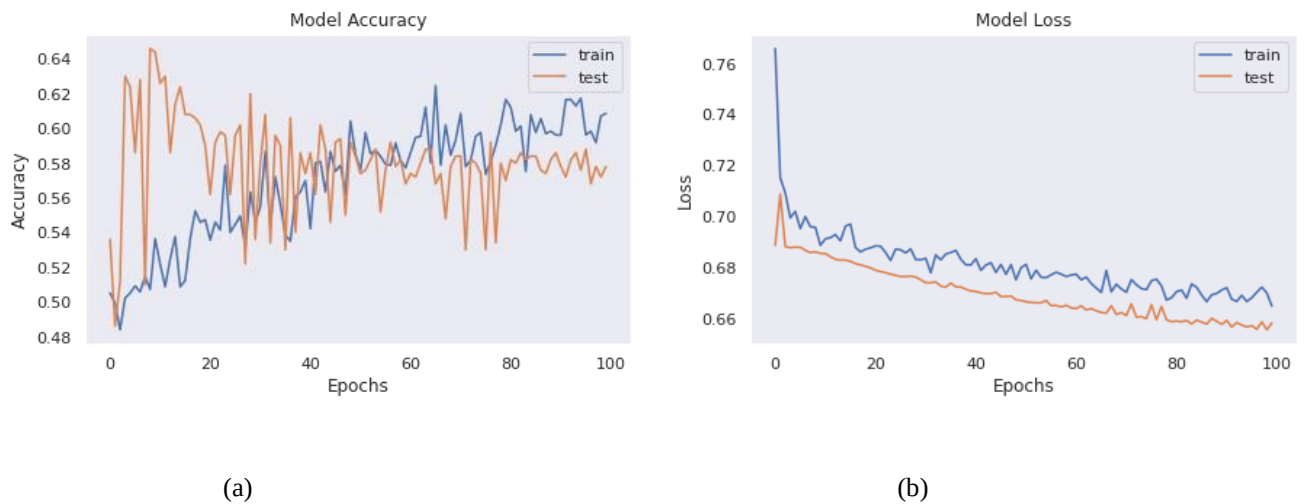


Figure: 2.8: (a) Xception model accuracy; (b) Xception model loss

The Xception model's confusion matrix is shown in Figure 2.9.

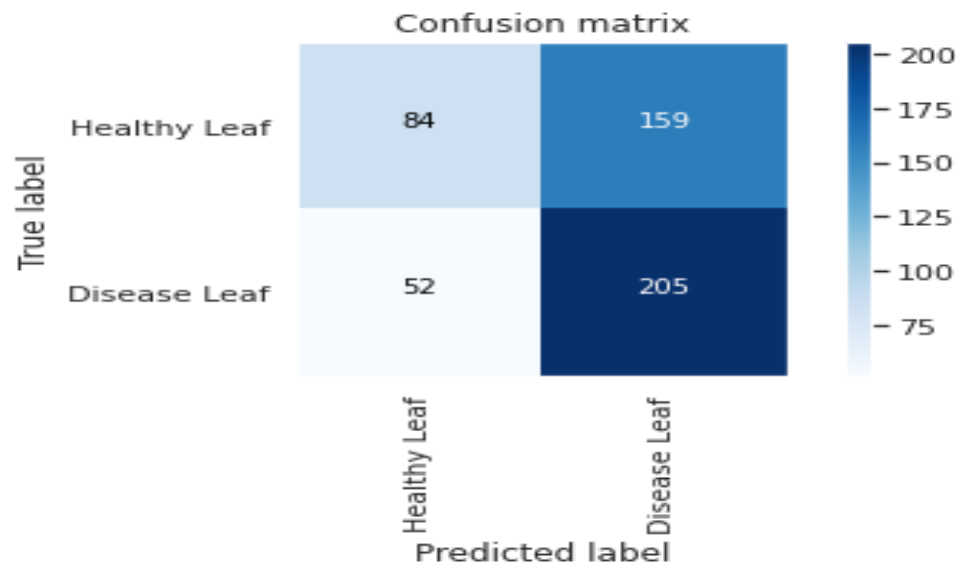


Figure: 2.9: Confusion matrix of Xception model

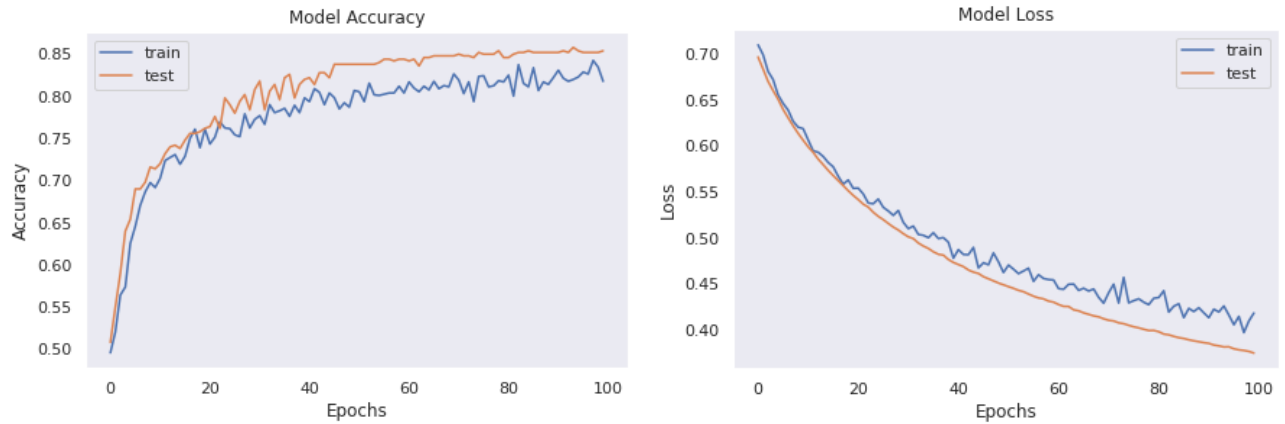
Table 1.6 shows the Xception architecture of the neural network that we used in this project, along with accuracy, recall, f2-score, and support, among other metrics.

Table 1.6: Classification report of Xception model

	Precision	Recall	F1- Score	Support
Healthy Leaf	0.62	0.35	0.44	243
Disease Leaf	0.56	0.80	0.66	257
Accuracy				0.58
Macro avg	0.59	0.57	0.55	500
Weighted avg	0.59	0.58	0.55	500

VGG16

Figure 3.1 (a) depicts the model's training and testing accuracy, whereas (b) depicts the model's training and testing loss.



(a) (b)
Figure 3.1: (a) VGG16 model accuracy; (b) VGG16 model loss

The VGG16 model's confusion matrix is shown in Figure 3.2.

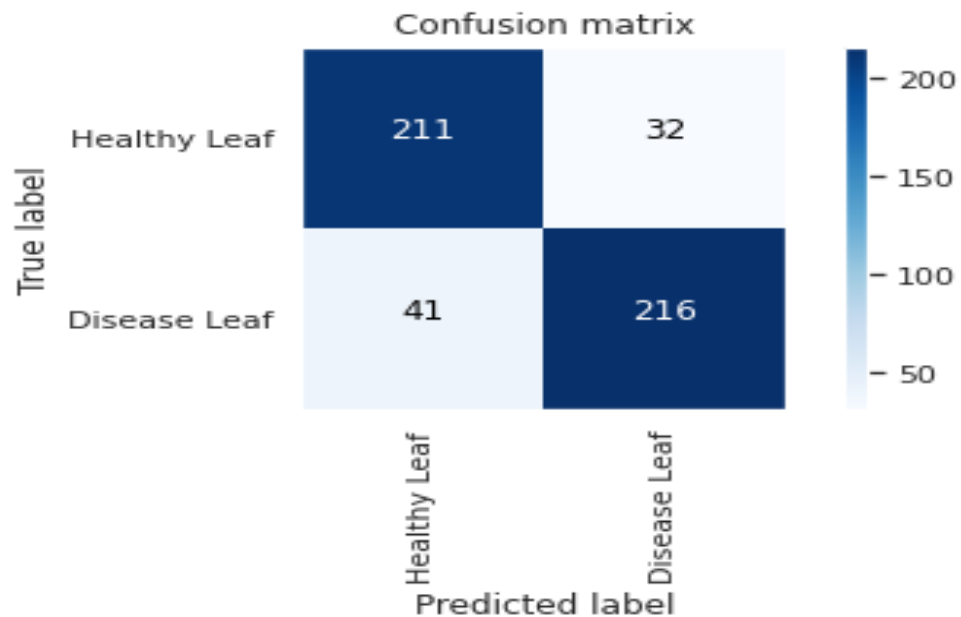


Figure 3.2: Confusion matrix of VGG16 model

Table 4.6 shows the VGG16 architecture of the neural network that we utilized in this research, along with accuracy, recall, f2-score, and support, among other metrics.

Table 1.7: Classification report of VGG16 model

	Precision	Recall	F1- Score	Support
Healthy Leaf	0.84	0.87	0.85	243
Disease Leaf	0.87	0.84	0.86	257
Accuracy			0.85	500
Macro avg	0.85	0.85	0.85	500
Weighted avg	0.85	0.85	0.85	500

4.4 Discussion

We made a chart to compare the accuracy of various architectures.

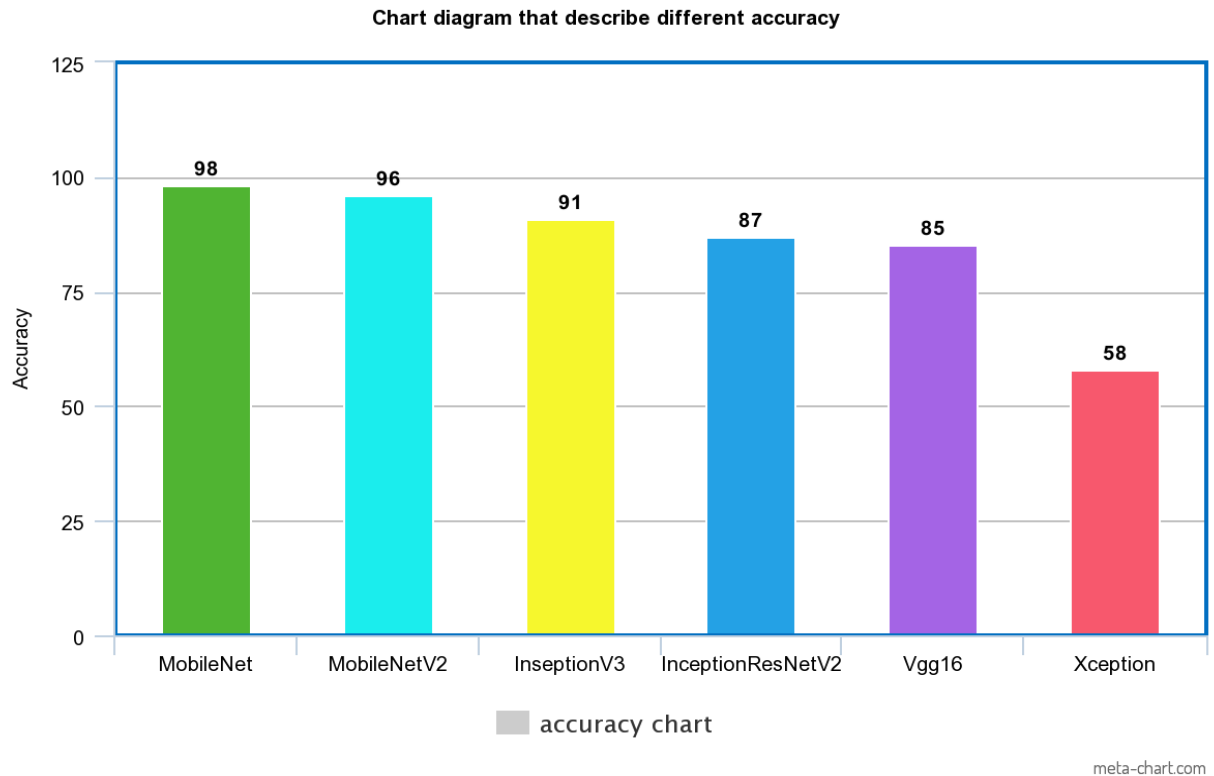


Figure: 3.3: Different accuracy of different model

According to the graph above, the accuracy of MobileNet is 98%, MobileNetV2 is 96 %, InceptionV3 is 91 %, InceptionResNetV2 is 87 %, VGG16 is 85 %, and Xception is 58%. After analyzing the accuracy of other models, we can see that MobileNet generates the greatest accuracy. As a result, our suggested model for identifying disease tea leaf from photos is MobileNet.

CHAPTER 5

Impact on Society, Environment and Sustainability

5.1 Impact on Society

We are all aware that there are several firms and individuals in Bangladesh that make a living from tea cultivation. They have to deal with a number of issues related to typical tea leaf diseases. This research is a boon to those businesses and individuals for whom identifying tea leaf diseases is not always possible. The majority of tea farmers in our nation are unaware of the disease that affects tea leaves. As a result, they are experiencing significant losses in their business. Because there is no available specialist in the remote region who can assist them in this case. Our project may be able assist them by recognizing the tea leaf issues and preventing a significant loss.

5.2 Impact on Environment

In today's society, it is a requirement of completing any sort of employment to ensure that your actions do not negatively impact the environment. Rather, it is strongly advised that you attempt to decrease pollution in the environment. Our activities have the potential to derail this crucial policy. Farmers may minimize garden pollution by implementing our project since, in most situations, farmers don't know the particular problem with the tea leaf, so they employ a variety of pesticides that have a negative impact on the garden environment. As a result of our initiative, we will be able to inform them of the actual situation and enable them to take action that will significantly minimize the use of dangerous chemicals

5.3 Ethical Aspects

There are a few misconceptions regarding the test result when it comes to the ethical aspect of machine learning. Which isn't always the case? Yes, there are a few examples where you may detect some concerns since some of the data they utilized in their projects appears to have been duplicated. However, we are extremely blessed in that we are able to manage all of the authentic data in our project. We get all of our information from 'Kazi & Kazi Ltd.,' which is the greatest source for tea leaf information. As a result, we can safely state that our methodology and initiative are ethically sound.

5.4 Sustainability Plan

The current project's main attribute or trait is its long-term viability. After designing and implementing the original edition of projects, the most important consideration for everyone should be project sustainability. We tried our hardest in our work to make them as long-term as possible. In the near future, we are planning to integrate our methodology into Android and iOS apps. We may also strive to make our initiatives speakable in Bangla so that individuals who don't know how to read or write can benefit from them. And, following the initial publication, the major objective is to improve our model's capability so that it can detect more diseases with greater accuracy.

CHAPTER 6

Summary, Conclusion, And Future Works

6.1 Summary of the Study

Tea leaf disease identification is one of the most important concerns in the tea industry. There are several methods for accomplishing this. However, the majority of them are inept in detecting leaf diseases. We utilized 2000 tea leaf images in this experiment to recognize a diseased leaf as well as a healthy leaf. We used a pre-trained architecture to find the best machine learning model for this task. We picked the MobileNet model since it delivers the greatest accuracy among the architectures we have deployed in this project after analyzing the performance of various models.

6.2 Future Works

In this effort, we focused on identifying tea leaf diseases using several deep CNN architectures. In the future, we will concentrate on identifying many types of leaf diseases. The most difficult aspect of our work was data collecting. We will also update the dataset and add new data in the future. For diverse leaf disease diagnosis, our proposed transfer model outperforms Xception, VGG16, MobileNetV2, InceptionResNetV2, and InceptionV3. However, there are various approaches to upgrade our model for transfer learning in the future. To improve accuracy, we will use additional transfer learning models such as DenseNet, AlexNet ResNet50, and others.

6.3 Conclusions

We used the dataset acquired from the tea garden in this research, therefore this project has offered an opportunity for the people of Asia, particularly Bangladesh and the Indian subcontinent, to have a more accurate forecast of tea leaf disease detection. Despite the fact that the suggested MobileNet model produced 98 percent test accuracy and 0.0701 test loss. In addition, we discussed total parameters, trainable parameters with different graphs, the Confusion Matrix, and the classification report. Our approach strategy is basic, yet it performs considerably faster and with more accuracy than another sophisticated model. The ultimate purpose of this research is to create and optimize a CNN model for detecting leaf disease. We will continue our research to improve the system's effectiveness.

Appendices

```
le=LabelEncoder()  
y_train = np.argmax(y_train, axis=1)  
y_test = np.argmax(y_test, axis=1)  
y_train=le.fit_transform(y_train)  
y_test=le.fit_transform(y_test)  
  
from sklearn.metrics import classification_report, confusion_matrix  
import itertools  
  
Y_pred = model.predict(x_test)  
from sklearn import metrics  
print(metrics.accuracy_score(y_test, Y_pred>0.5))
```

REFERENCES

- [1] B. Liu, Y. Zhang, D. He, and Y. Li, “Identification of apple leaf diseases based on Deep Convolutional Neural Networks,” *Symmetry*, vol. 10, no. 1, p. 11, 2018.
- [2] G. Geetharamani and A. Pandian, “Identification of plant leaf diseases using a nine-layer deep Convolutional Neural Network,” *Computers & Electrical Engineering*, vol. 76, pp. 323–338, 2019.
- [3] K. P. Ferentinos, “Deep learning models for plant disease detection and diagnosis,” *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, 2018.
- [4] A. T. Sapkal and U. V. Kulkarni, “Comparative study of leaf disease diagnosis system using texture features and deep learning features,” *International Journal of Applied Engineering Research*, vol. 13, no. 19, pp. 14 334–14 340, 2018.
- [5] Shaveta Malik, Tapas Kumar, Amiya Kumar Sahoo, —Image processing techniques for identification of fish disease, I 2017 IEEE 2nd International Conference on Signal and Image Processing (ICSIP), Vol. 15, No. 5, August 2017.
- [6] Arshia Rehman, Saeeda Naz, Muhammad Imran Razzak, Faiza Akram and Muhammad Imran, —A Deep Learning-Based Framework for Automatic Brain Tumors Classification Using Transfer Learning I, *Circuits Systems and Signal Processing (Circ Syst Signal Process)*, DOI: 10.1007/s00034-019-01246-3.
- [7] Jeong-Seon Park, Myung-Joo Oh, and Soonhee Han, —Fish Disease Diagnosis System Based on Image Processing of Pathogens’ Microscopic Images, I 2007 Frontiers in the Convergence of Bioscience and Information Technologies, 11-13 October. 2007.
- [8] Long Nguyen, Dongyun Lin, Zhiping Lin and Jiuwen Cao,— Deep CNNs for microscopic image classification by exploiting transfer learning and feature concatenation I, 2018 IEEE International Symposium on Circuits and Systems (ISCAS), May 2018.
- [9] The Architecture and Implementation of VGG-16, available at << <https://medium.com/towardsartificial-intelligence/the-architecture-and-implementation-of-vgg-16-b050e5a5920b> >>, last accessed on 05 October, 2021 at 12:32 AM.
- [10] Yiming Ding, Jae Ho Sohn, Michael G. Kawczynski, —A Deep Learning Model to Predict a Diagnosis of Alzheimer Disease by Using 18F-FDG PET of the Brain I, *RSNA*. Published online, Vol. 290, No. 2, pp. 456 – 464, Nov 06 2018 .

- [11] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov and Liang-Chieh Chen , —MobileNetV2: Inverted Residuals and Linear Bottlenecks, 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, DOI: 10.1109/CVPR.2018.00474.
- [12] MobileNet version 2, available at << <https://machinethink.net/blog/mobilenet-v2/> >> , last accessed on 05 October 2021 at 02:50 PM
- [13] Francois Chollet ,—Xception: Deep Learning with Depthwise Separable Convolutions, 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017.
- [14] Review: Xception — With Depthwise Separable Convolution, Better Than Inception-v3 (Image Classification), available at << <https://towardsdatascience.com/review-xception-with-depthwise-separableconvolution-better-than-inception-v3-image-dc967dd42568> >>, last accessed on 05 October, 2021 at 5:40 PM
- [15] Z. Q. Zhao, P. Zheng, S. T. Xu, and X. Wu, “Object detection with deep learning: A review,” IEEE Transactions on Neural Networks and Learning Systems, pp. 1–21, 2019.
- [16] S. Zhang, X. Wu, Z. You, and L. Zhang, “Leaf image based cucumber disease recognition using sparse representation classification,” Computers and Electronics in Agriculture, vol. 134, no. March, pp. 135–141, 2017.
- [17] S. P. Mohanty, D. P. Hughes, and M. Salathe, “Using deep learning for image-based plant disease detection,” Frontiers in Plant Science, vol. 7, pp. 1–10, 2016.
- [18] G. Geetharamani and A. Pandian, “Identification of plant leaf diseases using a nine-layer deep Convolutional Neural Network,” Computers & Electrical Engineering, vol. 76, pp. 323–338, 2019.
- [19] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, “Inception-v4, inception-ResNet and the impact of residual connections on learning,” in Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, California, USA, Feb. 04–09, 2017.
- [20] B. C. Karmokar, M. S. Ullah, M. K. Siddiquee, and K. M. R. Alam, “Tea leaf diseases recognition using neural network ensemble,” International Journal of Computer Applications, vol. 114, no. 17, pp. 27–30, 2015.

Plagiarism Report

aprb

ORIGINALITY REPORT

14%

SIMILARITY INDEX

10%

INTERNET SOURCES

11%

PUBLICATIONS

%

STUDENT PAPERS

PRIMARY SOURCES

1

dspace.daffodilvarsity.edu.bd:8080

Internet Source

3%

2

Sandeep Dwarkanath Pande, Pramod Pandurang Jadhav, Rahul Joshi, Amol Dattatray Sawant et al. "Digitization of handwritten Devanagari text using CNN transfer learning – A better customer service support", Neuroscience Informatics, 2022

Publication

1%

3

pubs.rsna.org

Internet Source

1%

4

www.pyimagesearch.com

Internet Source

1%

5

eprints.port.ac.uk

Internet Source

<1%

6

"Soft Computing for Security Applications", Springer Science and Business Media LLC, 2022

Publication

<1%