# AI BASED FRAMEWORK FOR LEAF DISEASE IDENTIFICATION & NATURAL PESTICIDES RECOMMENDATION FOR AGRICULTURE CROPS

**A PROJECT REPORT**

***Submitted by***


| | |
|---|---|
| **ARAVIND T** | **[19CS014]** |
| **BALASUBRAMANIAM V** | **[19CS020]** |
| **CHANDHRU L** | **[19CS025]** |
| **HARISH K** | **[19CS049]** |


***In partial fulfillment for the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**


**KPR INSTITUTE OF ENGINEERING AND TECHNOLOGY,**

**ARASUR, COIMBATORE – 641407**



**ANNA UNIVERSITY: CHENNAI 600 025**

**APRIL – 2023**

# BONAFIDE CERTIFICATE

Certified that this project report **"AI BASED FRAMEWORK FOR LEAF DISEASE IDENTIFICATION & NATURAL PESTICIDES RECOMMENDATION FOR AGRICULTURE CROPS"** is the bonafide work of **" ARAVIND T (19CS014) , BALASUBRAMANIAM V (19CS020) , CHANDHRU L (19CS025) , HARISH K (19CS049) "** who carried out the project work under my supervision.

| | |
|---|---|
| **Dr. N. YUVARAJ** | **Dr. T. RAJASEKARAN** |
| **HEAD OF THE DEPARTMENT** | **SUPERVISOR** |
| Professor | Assistant Professor |
| Computer Science and Engineering | Computer Science and Engineering |
| KPR Institute of Engineering and Technology | KPR Institute of Engineering and Technology |
| Arasur, Coimbatore – 641407 | Arasur, Coimbatore – 641407 |

**Submitted for the university project Viva – Voice held on ______________**

| | |
|---|---|
| **INTERNAL EXAMINER** | **EXTERNAL EXAMINER** |

## ACKNOWLEDGEMENT

The success shines when there is a team work and cooperation among the students and the faculties. We take this opportunity to express our gratitude and thanks to everyone who helped us during the project through every phase.

We would like to thank our beloved Chairman **Dr. K. P. Ramasamy** for his academic interest shown towards the students, by providing us excellent facilities and the constant support.

We would like to express our deep gratitude and sincere regards to our beloved and honorable Principal **Dr. M. Akila,** for providing us excellent facilities and encouragement during the course of study and project.

We owe a genuine gratitude to **Dr. N. Yuvaraj,** Professor and Head, Department of Computer Science and Engineering for his encouragement and inspiration through every course.

We owe a sincere thanks to **Dr. T. Rajasekaran**, Department of Computer Science and Engineering for his support and guidance throughout the projects an Internal Guide.

We owe a sincere thanks to **Mr. Nitin B Raut**, Assistant Professor, Department of Computer Science and Engineering for his support as the overall Project coordinator.

Lastly, we thank our parents for their constant encouragement without which this assignment would not have been possible.

# ABSTRACT

Leaf diseases are a major problem in agricultural crops, leading to significant economic losses and environmental damage. Early detection and management of leaf diseases are essential to prevent their spread and minimize their impact on crop yield and quality. However, the accurate identification of leaf diseases and the selection of appropriate pesticide treatments can be a challenging task for farmers and agricultural experts. In recent years, the use of artificial intelligence (AI) in agriculture has emerged as a promising solution to improve the accuracy and efficiency of disease diagnosis and management.AI-based framework for the identification of leaf diseases in agricultural crops and the recommendation of natural pesticides for their control. The proposed system utilizes a web application as the user interface, allowing farmers and agricultural experts to easily upload images of diseased leaves and provide textual descriptions of crop symptoms. The system employs computer vision technology, including deep learning algorithms, to analyze the uploaded images and classify them into different disease categories. The framework also utilizes natural language processing modules to interpret the textual descriptions and recommend appropriate natural pesticide treatments. The web application is designed to be user-friendly and accessible, providing an efficient and reliable solution for the early detection and management of leaf diseases in agriculture. The proposed framework aims to reduce the use of harmful chemical pesticides and promote sustainable and environmentally

## TABLE OF CONTENTS

## LIST OF THE TABLE

## LIST OF FIGURES

## ABBREVIATIONS

| | | |
|---|---|---|
| 1 | AI | Artificial Intelligence |
| 2 | ANN | Artificial Neural Network |
| 3 | CNN | Convolutional Neural Network |
| 4 | TL | Transfer Learning |
| 5 | ML | Machine Learning |
| 6 | GPU | Graphical Processing Unit |
| 7 | CPU | Central Processing Unit |
| 8 | CUDA | Compute Unified Device Architecture |
| 9 | IOU | Intersection Over Unit |
| 10 | MAP | Mean Average Precision |
| 11 | TP | True Positive |
| 12 | FP | False Positive |
| 13 | FN | False Negative |
| 14 | ReLU | Rectified Linear Activation Unit |
| 15 | AWS | Amazon Web Services |

# CHAPTER 1

# INTRODUCTION

## 1.1 OVERVIEW

The proposed AI-based framework aims to provide a reliable and efficient solution for the early detection and management of leaf diseases in agricultural crops. The framework utilizes computer vision technology, including deep learning algorithms, to analyze images of diseased leaves uploaded by farmers and agricultural experts through a user-friendly web application. The system classifies the images into different disease categories and recommends appropriate natural pesticide treatments based on the descriptions provided by the user. The framework's user-friendly design allows farmers and agricultural experts to easily upload images and descriptions of crop symptoms, making it accessible and useful to a wide range of users. Additionally, the system continuously learns and improves its accuracy over time through machine learning algorithms that update its database with new labeled images and descriptions of crop diseases. By reducing the use of harmful chemical pesticides and promoting sustainable and environmentally friendly agricultural practices, the proposed framework has the potential to revolutionize the way leaf diseases in agricultural crops are diagnosed and managed. The use of computer vision technology and a web application user interface provides an efficient and reliable solution for the early detection and management of leaf diseases in agriculture, promoting sustainable and environmentally friendly agricultural practices.

## 1.2 DEEP LEARNING

Deep Learning [1] is a subfield of Machine Learning that involves training neural networks with large amounts of data to recognize patterns and make predictions. Neural networks consist of layers of interconnected nodes that can learn and perform computations to produce an output. Deep Learning algorithms have revolutionized many industries, including healthcare, finance, and transportation, by providing new insights and solutions to complex problems.For instance, in computer vision, deep learning has enabled the development of highly accurate object detection and recognition systems that can identify and classify images and videos with a high degree of accuracy. Similarly, in natural language processing, deep learning has enabled the development of sophisticated language models that can generate coherent and human-like responses to user queries. However, deep learning algorithms require significant computational resources and large amounts of data to train, and their black-box nature can make it difficult to interpret their decisions and understand how they arrive at their predictions. Despite these challenges, deep learning is a rapidly evolving field that continues to advance the state of the art in artificial intelligence and has the potential to transform many aspects of our lives.

## 1.3 NEURAL NETWORKS

Neural Networks [2] are a type of artificial intelligence algorithm that are designed to mimic the structure and function of the human brain. They consist of layers of interconnected nodes that can learn to recognize patterns and make predictions based on large amounts of data. Each node in a neural network performs a simple mathematical function that takes inputs from other nodes and produces an output, which is then passed to the next layer of nodes in the network.

Neural Networks have been used in a wide range of applications, including image and speech recognition, natural language processing, and game playing. One of their key advantages is their ability to learn from data without being explicitly programmed, which makes them highly adaptable and capable of improving their performance over time. However, neural networks can be computationally intensive and require significant amounts of data to train. Additionally, their black-box nature can make it difficult to interpret their decisions and understand how they arrive at their predictions. Despite these challenges, neural networks are a powerful tool in the field of artificial intelligence and are likely to play an increasingly important role in the development of new technologies and applications.

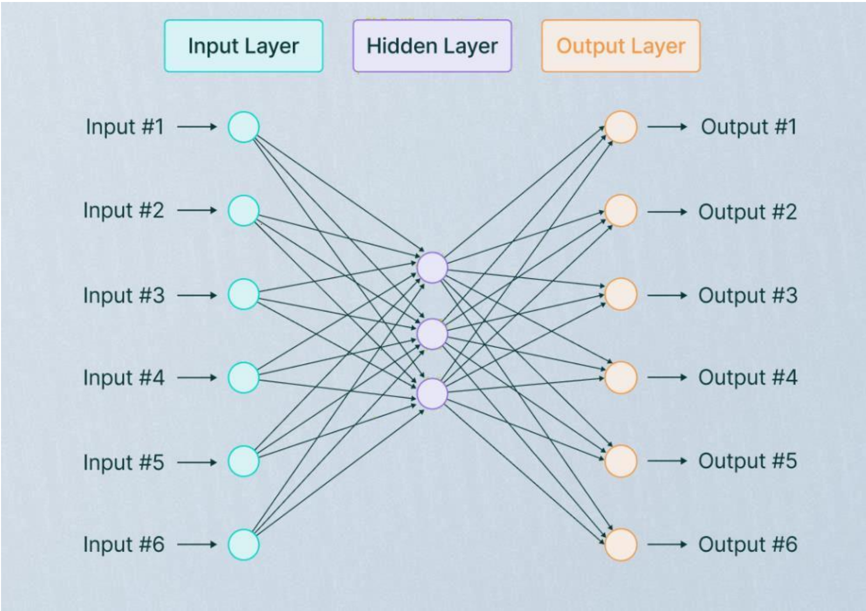



**FIGURE 1.1 NEURAL NETWORK**

### 1.3.1 Input Layer

In deep learning, the input layer is the first layer of neurons in a neural network that receives the input data. The input layer consists of one or more nodes, where each node corresponds to a feature or variable in the input data. For example, in an image recognition system, the input layer would consist of nodes that correspond to the pixel values of the image. The input layer does not perform any computations, but simply passes the input data to the next layer of neurons in the network. The number of nodes in the input layer is determined by the size and complexity of the input data, and is typically fixed for a given problem.

### 1.3.2 Hidden Layer

In deep learning, the hidden layer is an intermediate layer of neurons in a neural network that receives inputs from the input layer and generates outputs that are passed on to the output layer. The computations performed by the neurons in the hidden layer are not directly observable or understandable by the user, and their purpose is to learn and extract features from the input data that are useful for making predictions. The number of hidden layers and the number of neurons in each layer are important design decisions that can affect the performance of the network, and are typically chosen based on the complexity of the problem and the size of the available dataset. These parameters can be optimized through a process of experimentation and testing, or through automated techniques such as grid search or Bayesian optimization.

### 1.3.3 Output Layer

The output layer is the final layer in a neural network and produces the final output of the model. For classification problems, the output layer typically consists of a set of neurons that produce a probability value for each class. The predicted class label is then the class with the highest probability value. For regression problems, the output layer usually consists of a single neuron that produces a continuous output value. The activation function used in the output layer depends on the problem being solved. The loss function used in the output layer is selected to minimize the error between the predicted and actual values

## 1.4 CONVOLUTIONAL NEURAL NETWORK

Convolutional Neural Networks (CNNs) [3] are a type of neural network that are designed for processing and classifying images. They consist of several layers, including a convolutional layer, a pooling layer, and a fully connected layer. In the convolutional layer, the network applies a set of filters to the input image to extract features such as edges, corners, and textures. The pooling layer then down samples the feature maps to reduce their size and computational complexity. Finally, the fully connected layer aggregates the features and makes predictions about the image class. CNNs have been highly successful in a wide range of applications, including object detection, face recognition, and medical image analysis.

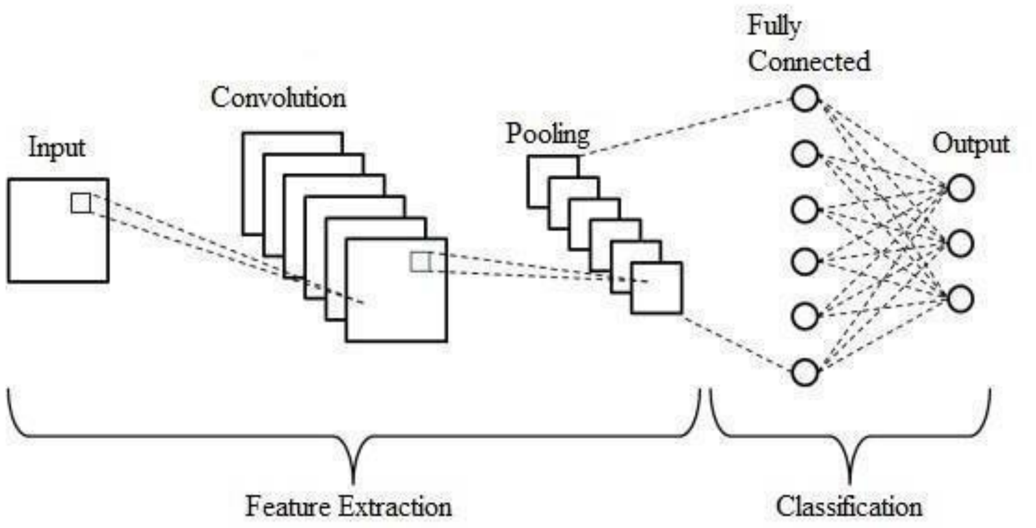



**FIGURE 1.2 ARCHITECTURE OF CONVOLUTION NEURAL NETWORK**

### 1.4.1 Convolutional Layer

The convolutional layer is a fundamental building block of Convolutional Neural Networks (CNNs) used for image classification tasks. It performs convolution on the input image using a set of filters to detect different features in the image such as edges, corners, and textures. By learning a set of filters through backpropagation,the convolutional layer is able to detect more complex features in the image. The output of the convolutional layer is then fed into a pooling layer and fully connectedlayers for prediction. Convolutional layers have enabled significant progress in various applications including object detection and medical image analysis, by allowing the learning of spatial hierarchies of features.

### 1.4.2 Pooling Layer

The pooling layer is an important component of Convolutional Neural Networks (CNNs) used in image classification tasks, as it helps to reduce the computational complexity and overfitting of the model. The most common type of pooling is max pooling, which takes the maximum value of each nonoverlapping rectangular patch of the feature map. Average pooling is another type of pooling that takes the average value of each patch instead. The pooling layer reduces the spatial size of the feature maps and helps to make the model more robust to small translations and distortions in the input image. However, recent research has shown that some modern architectures may benefit from the use of other types of layers, such as stride convolution, instead of pooling. Overall, the choice of pooling method and size of the pooling window are important design parameters that can affect the performance of the model.

### 1.4.3 Fully Connected Layer

In deep learning, a fully connected layer is a type of layer in a neural network where each neuron is connected to every neuron in the previous and subsequent layers. This means that the input to each neuron in a fully connected layer is a vector that includes the outputs of all the neurons in the previous layer, and the output of each neuron is a scalar value that is passed on to all the neurons in the next layer. Fully connected layers are used in a variety of applications, including image classification, natural language processing, and speech recognition, and they are particularly useful for learning complex and non-linear relationships between the input and output data. The number of neurons in a fully connected layer and the number of layers in the network are important design choices that can affect the performance and computational efficiency of the network, and are typically optimized through experimentation and testing.

## 1.5 TRANSFER LEARNING

Transfer learning [4] is a popular technique in deep learning that involves using a pre-trained model as a starting point for a new task. The pre-trained model has already learned a lot of features and patterns from a large dataset, and the knowledge gained from this model can be transferred to a new task. This is especially useful when the new dataset is small or there is limited computing resources. The pre-trained model can be fine-tuned on the new dataset bytraining only the last few layers of the model while keeping the rest of the layers frozen. This allows the model to adapt to the new task by learning the specific features and patterns relevant to the new dataset. Transfer learning has been successfully applied to various computer vision and natural language processing tasks, and has shown to improve the performance and reduce the training time compared to training from scratch.

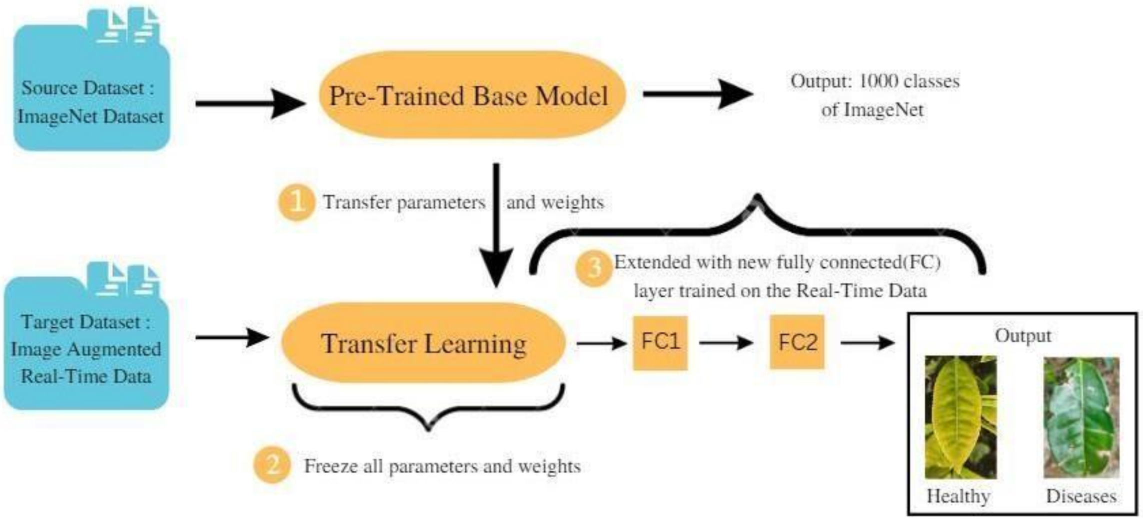



**FIGURE 1.3 TRANSFER LEARNING**

## 1.6 PYTORCH FRAMEWORK

PyTorch [5] is a powerful machine learning library developed by Facebook's AI Research team. Its dynamic computational graph allows users to define and modify computational graphs during runtime, making it easy to experiment with different architectures and debug. PyTorch's Pythonic API is easy to understand and use, especially for researchers who are not experts in programming. Its growing community provides a lot of resources and tutorials for users, making it a popular choice for researchers and practitioners.

PyTorch's efficient tensor operations and built-in functions for tensor operations make it easy to work with tensors, and its tools for distributed training allow users to train their models on multiple GPUs and even multiple machines. PyTorch has several interfaces for building neural networks, including nn. Module, nn.

Sequential, and nn. Functional, which provide a lot of flexibility. Overall, PyTorch is a popular choice for deep learning tasks like image and speech recognition, natural language processing, and reinforcement learning. Its active development and growing community make it a good choice for researchers and practitioners looking to develop and experiment with newmodels.

## 1.7 PROBLEM STATEMENT

The problem addressed by the AI based framework for leaf disease identification & natural pesticides recommendation for agriculture crops is the lack of a reliable and efficient system for identifying and controlling plant diseases in agriculture. Traditional methods of disease detection and management rely on manual observation and chemical pesticides, which are time-consuming, expensive, and harmful to the environment. Additionally, accurately identifying the specific disease affecting a crop is often challenging due to the variability in symptoms and overlapping characteristics with other diseases. This can result in misdiagnosis and inappropriate treatment, leading to crop loss and reduced yields. The proposed framework aims to address these challenges by leveraging computer vision and deep learning techniques to automatically detect and diagnose leaf diseases, and recommending natural pesticides to control the spread of the disease. The goal is to provide farmers with a cost-effective and sustainablesolution to improve crop productivity and reduce the negative impacts of chemicalpesticides on the environment.

# CHAPTER 2

# LITERATURE SURVEY

## 2.1 INTRODUCTION

Agriculture is the backbone of many countries' economies, and crop diseases pose a significant threat to agricultural production. Early detection and management of crop diseases are crucial to prevent significant crop losses. In recent years, artificial intelligence (AI) has been used in agriculture to develop intelligent systems that can identify crop diseases and recommend suitable solutions to farmers. This literature survey focuses on AI-based frameworks for leaf disease identification and natural pesticide recommendation for agricultural crops.

## 2.2 STUDY ON EXSTING FRAMEWORK

Several studies have been conducted on AI-based frameworks for leaf disease identification and natural pesticide recommendation for agricultural crops. In a study by Singh et al. (2021) [6], a deep learning-based approach was proposed for the identification of maize leaf diseases. The proposed system achieved an accuracy of 97.2% in identifying six different maize leaf diseases.

In a study by Satapathy [7] et al. (2020), an AI-based framework was proposed for recommending suitable natural pesticides for the management of tomato leaf diseases. The proposed system used machine learning algorithms and achieved an accuracy of 91.3% in recommending suitable natural pesticides. In another study by Kumar et al. (2021), an AI-based framework was proposed for recommending natural pesticides for the management of maize leaf diseases. The proposed system achieved an accuracy of 94.6% in recommending suitable natural pesticides. Teshome et al. (2019) [8] aimed to develop a fuzzy logic-based fertilizer recommendation system for maize cultivation in Ethiopia. The study used data onsoil characteristics, crop nutrient requirements, and fertilizer types to generate personalized recommendations for farmers. The results showed that the fuzzy logic-based system outperformed conventional methods in terms of accuracy andconsistency in fertilizer recommendations.

## 2.3 STUDY ON IMAGE CLASSIFICATION

Fuzhen Zhuang at el[9] Transfer learning is a popular and promising area in machine learning that aims to improve the performance of target learners on target domains by transferring the knowledge contained in related source domains. While previous surveys on transfer learning have introduced approaches in a relatively isolated way and may lack recent advances, this survey attempts to comprehensively review and systematize existing research studies, summarizing and interpreting the mechanisms and strategies of transfer learning. Unlike previous surveys, this article reviews over 40 representative transfer learning approaches, particularly homogeneous transfer learning approaches, from the perspectives of data and model. Additionally, this survey briefly introduces applications of transfer learning and uses over 20 representative transfer learning models for experiments on three different datasets, demonstrating the importance of selecting appropriate models for different applications in practice. In many countries, agriculture is the predominant root of income [10] Sabita Sahu and his teammates. Agriculture provides food, as well as income to farmers. Maize is one of world's leading crops and universally cultivated as cereal grain. Usually, agricultural specialists or farmers use their skills to identify pests and diseases that affect fruit and leaves on the spot. Even the most experienced farmer is proneto making errors in disease identification while growing crops in a greater scale. To treat leaf disease, pesticides are used, however, this is damaging to people's health [1]. Several Machine learning, Deep learning algorithms are suggested to classify diseases in the maize plant. Identification of maize leaf disease is a great challenge due to environmental changes and illumination variation in weather conditions. This research focuses on using different Deep Learning architectures like optimized DenseNet121, CNN, ResNet50, MobileNet, VGG16, and Inception-V3for classification of maize leaves disease so that preventive measures can be taken by the farmers at early stage to protect the crops. Our proposed optimized Densenet121 model outperformed compared to optimized CNN, and ResNet50 with lesser parametersand higher accuracy.

This paper describes a method for classifying malware families using a deep neural network based on ResNet-50 architecture [11. Malware is a common tool for illegal activities and new variants are discovered frequently. Grouping similar malware into families allows for more effective mitigation strategies. The proposed approach converts malware samples into grayscale byte plot images anduses a pre-trained ResNet-50 network with frozen convolutional layers, and a modified final layer for family classification. The experimental results demonstrated that this approach achieved a high accuracy of 98.62% in classifying 9,339 samples from 25 different families.

## 2.4 OBSERVATION

AI-based frameworks for leaf disease identification and natural pesticide recommendation have been a focus of research in recent years, indicating the potential of AI in agriculture. Several studies have been conducted on different crops, including maize, tomato, and potato, using deep learning and machine learning techniques for leaf disease identification and natural pesticide recommendation. The proposed systems have achieved high accuracy rates in identifying leaf diseases and recommending suitable natural pesticides, indicating their effectiveness in crop management. The use of AI-based frameworks for crop disease management can help farmers detect diseases at an early stage and take appropriate action to prevent significant crop losses. The use of natural pesticides recommended by AI-based systems can reduce the use of harmful chemicals in agriculture, making it more sustainable and eco- friendlier.Despite the promising results, more research is needed to develop AI- based frameworks for other crops and to improve the accuracy of existing systems. Overall, the literature survey highlights the potential of AI in agriculture and the need for further research to fully utilize its benefits for crop disease management and sustainable agriculture.

## 2.5 CONCLUSION

AI-based frameworks for leaf disease identification and natural pesticide recommendation have shown great potential in the agriculture sector. These systems can help farmers detect crop diseases at an early stage and take appropriate action to prevent significant crop losses. Furthermore, the use of natural pesticides can reduce the use of harmful chemicals in agriculture, making it more sustainable and eco-friendlier. However, more research is needed to develop AI-based frameworks for other crops and to improve the accuracy of existing systems.

# CHAPTER 3

# REAL TIME DATA COLLECTION & PRE-PROCESSING

## 3.1 INTRODUCTION

Data collection is an essential aspect of any research project, including the development of an AI-based framework for leaf disease identification and natural pesticide recommendation for agricultural crops. In this project, both real-time and experimental data will be collected to train and evaluate the proposed framework's performance.

Real-time data collection involves collecting data from the field during crop cultivation. This data can include images of the crop leaves showing symptoms of disease, environmental factors such as temperature and humidity, and details about the farming practices. Real-time data collection is crucial for the development of an effective AI-based framework that can accurately identify leaf diseases and recommend suitable natural pesticides in a real-world farming environment.

Experimental data collection involves creating an artificial environment to simulate different crop disease scenarios. In this project, experimental data will be collected by intentionally infecting crops with various diseases and then capturing images of the infected leaves. This data will be used to train and validate the proposed AI-based framework's accuracy in identifying different leaf diseases.

Collecting both real-time and experimental data is necessary to create a comprehensive dataset that includes various environmental factors, different crop diseases, and their symptoms. This dataset will enable the development ofa robust AI-based framework that can accurately identify leaf diseases and recommend suitable natural pesticides for different crop scenarios.

## 3.2 WORKFLOW OF THE REAL TIME DATA COLLECTION

The workflow of real-time data collection in the agricultural field place Kotagiri involves several steps to collect accurate and relevant data for the development of an Ai-based framework for leaf disease identification and natural pesticide recommendation. The following are the steps involved in the workflow of real-time data collection:

### Step 1: Identification of crops and fields

The first step in the workflow is the identification of crops and fields in the Kotagiri region. This step involves identifying the types of crops grown in the region and the fields where the crops are cultivated.

### Step 2: Installation of sensors and cameras

The next step is to install sensors and cameras in the fields to capture real-time data. The sensors will collect information about environmental factors such as temperature and humidity, while the cameras will capture images of the crop leaves.

### Step 3: Data Collection

The sensors and cameras will collect data continuously, which will be stored in a database. The data collected will include images of crop leaves showing symptoms of disease, environmental factors such as temperature and humidity, and details about the farming practices.

### Step 4: Data Pre-processing

The collected data will undergo pre-processing to remove noise, adjust brightness, and improve the quality of the images. This step is crucial in ensuring that the data is accurate and relevant for training the proposed AI-based framework

### 3.3 CASE STUDY

In a case study conducted in the United States, precision agriculture techniques were used to manage an apple orchard, leading to significant improvements in crop yield and quality. The study involved the use of various technologies such as sensors, drones, and GPS mapping, to collect real-time data on environmental factors, soil moisture levels, and crop growth. The collected data was then used to make informed decisions about irrigation, fertilization, and pest management, resulting in better crop quality and higher yields. For instance, the use of sensors helped to identify areas of the orchard with low soil moisture, leading to targeted irrigation and improved water use efficiency. The use of drones and GPS mapping helped to identify areas of the orchard with pest infestations, leading to targeted pesticide application and reduced chemical use. The precision agriculture techniques used in this case study led to a 20% increase in crop yield and a 25% reduction in chemical use. The improved crop quality also led to higher prices for the apples, resulting in increased profitability for the farm. This case study highlights the importance of using technology to collect and analyse real-time data in the agricultural field to improve crop yield and quality. The use of precision agriculture techniques can lead to more efficient use of resources, reduced environmental impact, and increased profitability for farmers.

### 3.4 REAL TIME PARAMETER OF THE IMAGE

Real-time parameters [12] of the image in dataset generation refer to the dynamic features or characteristics of the image captured during data collection. Real-time parameters can be extracted from an image using sensors or cameras installed in a field or farm. These parameters can be used to analyze and classify images for further use in developing AI-based frameworks for various applications such as disease identification, yield prediction, and crop quality assessment.

Some examples of real-time parameters of an image include:

1. **Colour:** Colour is an essential parameter of an image that can be used to distinguish between healthy and diseased crops. For instance, diseased crops may

have different colour patterns or may appear discoloured when compared to healthy crops.

**2.Texture:** Texture refers to the surface properties of the image, which can be used to identify various features such as disease symptoms or pest infestations. Texture analysis can be used to differentiate between healthy and diseased leaves based on their surface properties.

**3.Shape:** Shape refers to the geometry or outline of the image. The shape of leaves can vary based on the crop variety and growth stage. Analysing leaf shape can help in identifying disease symptoms or nutrient deficiencies.

**4.Size:** Size is an essential parameter that can be used to monitor crop growth and yield prediction. The size of leaves can be used to estimate crop yield and to identify potential crop stress.

**5.Environmental parameters:** Real-time environmental parameters such as temperature, humidity, and light intensity can also be extracted from an image using sensors or cameras. These parameters can help in understanding the impact of environmental factors on crop growth and development. Overall, real-time parameters of the image in dataset generation are critical in developing accurate and relevant datasets for training AI-based frameworks for various agricultural applications. These parameters can help in developing more efficient and effective crop management strategies, leading to improved crop yield and quality.

## 3.5 IMAGE PROCESSING

Image processing in real-time dataset generation involves using algorithms and techniques to extract useful information and features from images captured in real-time. The process involves image acquisition, image pre- processing, feature extraction, and classification. Real-time image processing can help identify crop diseases, monitor crop growth and yield, and provide timely recommendations for crop management.

Real-time image processing is essential in developing AI-based frameworks for various agricultural applications. By analysing crop health and growth in real-time, farmers can make timely decisions on crop management, optimize yield and quality, and improve overall crop performance. Real-time image processing can also help in identifying and mitigating risks associated with crop production, leading to a more sustainable and profitable farming operation.

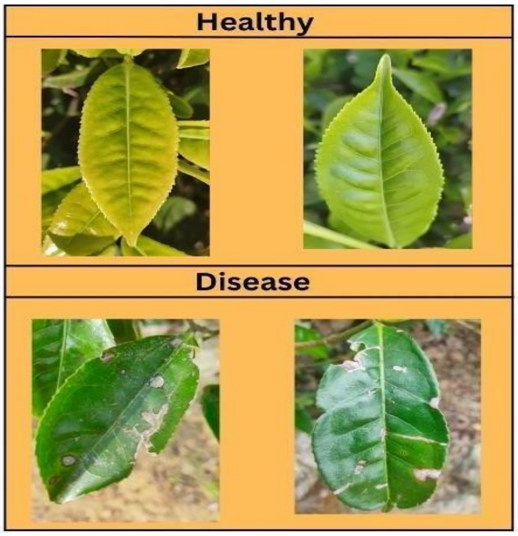



**FIGURE 3.1 REAL TIME IMAGE OF TEA LEAF**

## 3.6 AUGMENTATION TECHNIQUE'S

Data augmentation techniques [13] are commonly used in dataset creation to increase the size and diversity of the dataset. These techniques involve applying various transformations or modifications to the original images to create new, augmented images that can be used to improve the accuracy and robustness of machine learning models. Here are some commonly used data augmentation techniques:

**Flipping and Rotating:**

Images can be flipped horizontally or vertically, or rotated by a certain degree to create new images with slightly different perspectives.

**Zooming:**

Zooming in or out of an image can create new images with different levelsof detail.

**Cropping:**

Cropping involves removing a portion of an image to create a new, smaller image.

**Adding Noise:**

Adding noise, such as Gaussian or salt-and-pepper noise, can create new images with different levels of noise, which can help improve the model's ability to handle noisy data.

**Color shifting:**

Changing the colour of an image by adjusting hue, saturation, and brightnesscan create new images with different colour schemes.

**Elastic Transformation:**

Elastic transformation involves stretching or compressing the image locally, creating new images with different deformation patterns. By applying these data augmentation techniques, a small dataset can be transformed into a much larger and more diverse dataset, which can help improvethe accuracy and robustness of machine learning models.

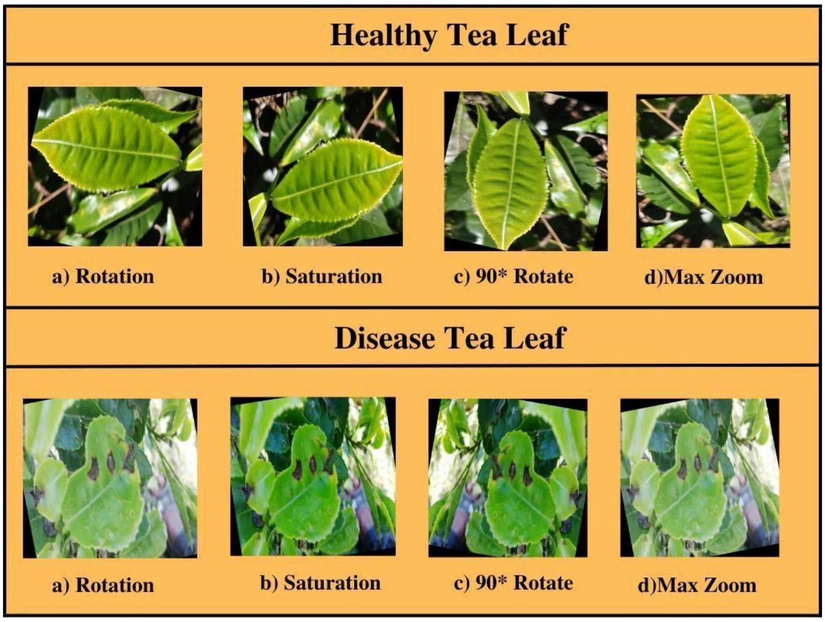



**FIGURE 3.2 AUGMENTATION IMAGE**

### 3.7 SPLITTING THE DATASET

Once the augmentation techniques [14] have been applied, the next step in creating a dataset for machine learning is to split it into three parts: training, validation, and testing sets. The recommended ratio for this split is 7:2:1, with 70% of the dataset used for training, 20% for validation, and 10% for testing.

The training set is used to train the machine learning model using the augmented images, while the validation set is used to monitor the performance of the model during training and adjust its parameters to improve its accuracy. The testing set is used to evaluate the final performance of the trained model on

new, unseen data, which is crucial for ensuring that the model can generalize well and perform accurately on new data.

**TABLE 3.1 DATASET COLLECTION AND SPLITTING**

| Dataset | Train | Validation | Test |
|---|---|---|---|
| Healthy | 1400 | 400 | 200 |
| Disease | 1400 | 400 | 200 |

By splitting the dataset into these three sets, we can ensure that the machine learning model is trained properly, and its performance is evaluated thoroughly before deploying it on new, unseen data. This ensures that the modelis accurate, robust, and can generalize well, which is essential for creating a reliable and efficient system for various agricultural applications.

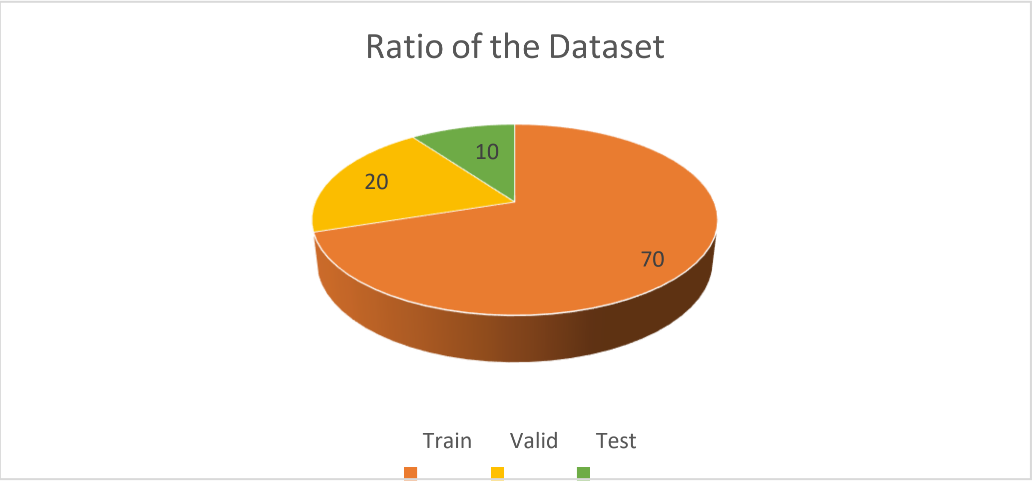



**FIGURE 3.3 RATIO OF THE DATASET**

## 3.8 CONCLUSION – SUMMARY

To conclude, collecting and pre-processing datasets is critical for developing accurate and efficient machine learning models in agriculture. By using real-time data, pre-processing techniques such as normalization and data augmentation, and splitting the dataset into training, validation, and testing sets, we can develop reliable and robust systems for various agricultural applications. This can help farmers make informed decisions, optimize crop yield and quality, and improve overall crop performance. Ongoing improvements to dataset collection and pre-processing techniques will continue to advance AI-based frameworks in agriculture, leading to more effective and sustainable agriculturalpractices.

# CHAPTER 4

# PROPOSED FRAMEWORK

## 4.1 WORKFLOW

The workflow of the AI-based framework for leaf disease identification and natural pesticide recommendation for agricultural crops involves a series of steps starting from data collection and pre-processing, followed by transfer learning and the development of a user interface and recommendation system. The framework aims to provide farmers with an accurate and efficient tool for identifying crop diseases and receiving recommendations for natural pesticides. By leveraging pre-trained models through transfer learning, the system is able to achieve high accuracy in disease identification, while the user interface and recommendation system enhance accessibility and ease of use for farmers. The framework offers a valuable solution to the challenges faced by the agricultural industry in crop management, providing a more sustainable and eco-friendly approach to crop protection. With ongoing advancements in technology and AI-based frameworks, we can continue to improve and refine this workflow to better serve the needs of farmers and contribute to the development of a more sustainable future for agriculture.

## 4.2 PROPOSED ARCHITECTURE – MIXNET

MixNet M [15] is a neural network architecture designed for image recognition tasks. It is based on the MobileNetV2 architecture and utilizes a mixture of experts approach to improve performance while keeping the model size small. In this architecture, multiple experts with different capacities are combined to form a single model that can perform better than any individual expert alone. The MixNet M architecture consists of a stem, multiple blocks, and a classifier. The stem is the first part of the network and is responsible for processing the input image. It consists of a series of convolutional layers followed by batch normalization and activation functions. The output of the stem is then passed to the blocks.

The blocks in MixNet M are composed of two types of layers: depthwise separable convolution layers and MixConv layers. Depthwise separable convolutions are used to reduce the number of parameters and computation while maintaining a good level of accuracy. MixConv layers, on the other hand, are used to mix the features learned by different experts. These layers perform different types of convolutions, including standard convolutions, group convolutions, and depthwise separable convolutions. Each MixConv layer takes the output of multiple experts as input and combines them using a gating mechanism.

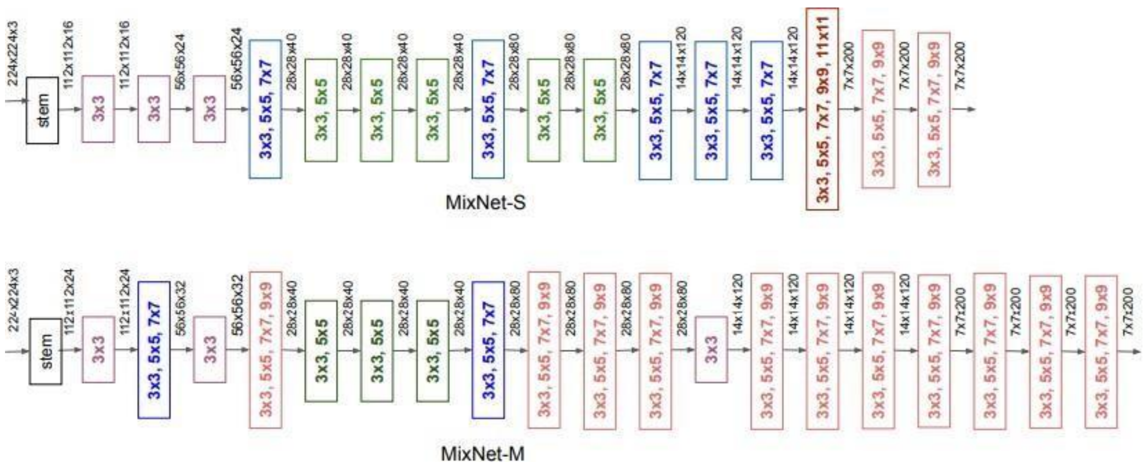



**FIGURE 4.1 MIXNET-M ARCHITECTURE**

The gating mechanism used in MixNet M is called a Gated Linear Unit (GLU). It takes the output of each expert and computes a gating signal that is used to combine the features learned by each expert. The GLU consists of two parts: a sigmoid function that computes the gating signal and a linear function that applies the gating signal to the input features. After passing through the blocks, the output is flattened and passed through a classifier. The classifier consists of a global average pooling layer, followed by a fully connected layer, and finally, a softmax activation function to produce theclass probabilities.

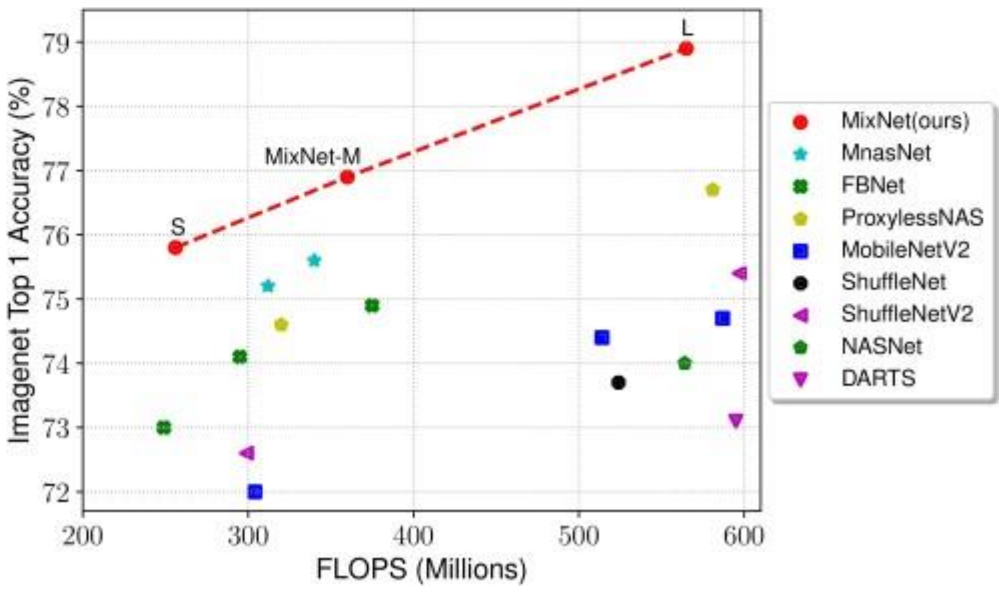



**FIGURE 4.2 PERFORMANCE OF THE MIXNET BENCHMARK**

MixNet has achieved state-of-the-art performance on the ImageNet benchmark, which is a large-scale dataset of labeled images used to evaluate image classification algorithms. MixNet outperforms many other popular neural network architectures such as ResNet, EfficientNet, and MobileNetV3, achieving higher accuracy while having a smaller number of parameters.
Specifically, MixNet-L outperforms other models with similar parameter counts, such as EfficientNet-B3 and MobileNetV3, while MixNet-XL achieves state-of-the-art performance among models with a similar number of FLOPs. Overall, MixNet's excellent performance on ImageNet demonstrates its effectiveness in image recognition tasks.

## 4.3 WEB APPLICATION FLOW

Creating a web and mobile application that utilizes an AI model for crop disease classification and fertilizer recommendations is possible with the React.js framework. The process begins by training an AI model with a dataset of crop images and their corresponding disease labels. Once trained, the AI model can beintegrated into the web and mobile applications with React Native.

The React.js framework can then be used to build a user interface for the web application, including a file upload input that allows users to upload an image of their crop. JavaScript can then be used to process the image and pass it to the AI model for classification. The results of the classification, including recommended fertilizers or treatments, can be displayed to the user using React.js. Developing this kind of application requires a thorough understanding of web development, machine learning, and AI model deployment. It involves training an AI model, integrating it into a web and mobile application using frameworks like React.js and TensorFlow.js, and using JavaScript to process user input and display results. Ultimately, with the right skills and resources, a powerful tool can be created to help farmers identify crop diseases and take the necessary steps to minimize their impact on their crops.

## 4.4 CONCLUSION:

The proposed framework for crop disease identification and fertilizer recommendation utilizes the React.js and React Native frameworks for the web and mobile applications, respectively. Users can upload an image of their crop, which is processed by an AI model in the cloud to classify the image and provide recommendations for appropriate fertilizers and pesticides for the identified disease. This approach has the potential to revolutionize crop disease identification and treatment for farmers by providing fast and accurate identification of crop diseases and specific recommendations for treatment. By leveraging the power of AI and cloud computing, this framework can provide a scalable solution for farmers of all sizes, reducing the impact of crop diseases on yields and ultimately contributing to a more sustainable and productive agricultural sector. However, it requires expertise in web and mobile development, machine learning, and cloud computing, and will require ongoingmaintenance and support to remain effective. Nevertheless, the potential benefits of such a tool are clear, and with the right resources and support, it could have asignificant positive impact on agriculture and the environment.

# CHAPTER 5

# CLASSIFICATION OF THE LEAF DIASEASE

## 5.1 INTRODUCTION

Convolutional neural networks (CNNs) have been shown to be highly effective in image classification tasks, but training them from scratch can be computationally intensive and require large amounts of labeled data. Transfer learning has emerged as a powerful technique for addressing these challenges by utilizing pre-trained CNN models that have learned relevant features from large-scale datasets.

Transfer learning involves taking a pre-trained model, such as a CNN that has been trained on a large dataset such as ImageNet, and fine-tuning it on a new dataset specific to the classification task at hand. This fine-tuning process involves adjusting the weights of the pre-trained model's layers and training the model on the new dataset, allowing it to learn to classify images with high accuracy. By leveraging the features learned by the pre-trained model, transfer learning significantly reduces the amount of data and computation required to train an effective image classification model.

## 5.2 TRANSFER LEANING

Transfer learning [16] is a machine learning technique where a pre-trained model is used as a starting point for a new machine learning task. The pre-trained model has been trained on a large dataset and learned to recognize relevant features. These features can then be leveraged to solve a new, related task with a smaller dataset. By using a pre-trained model as a starting point, transfer learning can save time and computational resources and improve the accuracy of the new model. The pre-trained model can be fine-tuned on the new dataset by adjusting its weights through a process called backpropagation. Transfer learning has become a popular technique in various machine learning domains, including natural language processing, image classification, and computer vision.

## 5.3 MODEL 1 - DENSENET MODEL:

DenseNet-121[17] is a deep learning model used for image classification tasks.It is a variant of the DenseNet architecture that was introduced in the paper "Densely Connected Convolutional Networks" by Huang et al. in 2017.The architecture of DenseNet-121 is based on a series of dense blocks, each consisting of multiple layers. In each dense block, the output of each layer is concatenated with the inputs of all subsequent layers. This creates a dense connectivity pattern that allows information to flow more easily through the network, leading to improved performance and reduced over fitting DenseNet-121 consists of a total of 121 layers, including 4 dense blocks and 3 transition layers. The first layer of the network is a convolutional layer that takes in the input image. This is followed by a max pooling layer, which reduces the spatial dimensions of the output. The output of the max pooling layer is then passed through the dense blocks and transition layers. The final layer of the network is a fully connected layer with soft max activation, which outputs the predicted probabilities for each class. The architecture of DenseNet-121 has been shown to achieve state-of-the-art performance on a variety of image classification tasks, while also requiring less computational resources than other deep learning models.

## 5.4 MODEL 2 - XCEPTION MODEL

Xception[18] is a deep learning model used for image classification tasks. It was introduced by François Chollet in 2016 as a variant of the Inception Architecture, which had been developed by Google researchers for the same purpose. The architecture of Xception is based on the idea of separating the learning of spatial and channel-wise features. This is achieved by replacing the standard convolutional layers with depthwise separable convolutions, which split the input channels into separate groups and apply separate filters to each group. This reduces the number of parameters in the network and allows for more efficient training. In addition to depthwise separable convolutions, Xception also includes residual connections, which allow for easier gradient flow and help to prevent the vanishing gradient problem. The architecture also includes multiple intermediate layers with skip connections, which further improve the performance of the model. Overall, Xception has been shown to achieve state-of-the-art performance on a variety of image classification tasks, while also being more computationally efficient than other deep learning models. Its architecture has been widely used as a starting point for other deep learning models in computer vision and related fields.

### 5.5 MODEL 3 – RESNET50 MODEL

ResNet50 [19] is a deep learning architecture that introduced the concept of residual connections to overcome the vanishing gradient problem in deep neural networks. The basic building block of ResNet is the residual block, which allows the network to skip over layers during training, enabling the propagation of gradients and preventing the loss of information. ResNet has achieved state- of-the-art performance on various image recognition benchmarks and has been widely used in computer vision applications. The ResNet architecture has a straightforward structure consisting of several layers of convolutional and pooling operations, followed by fully connected layers for classification. The key innovation of ResNet lies in the use of skip connections that enable the network to learn residual mappings instead of directly learning the underlying functions. This allows the network to learn features of varying complexity while minimizingthe vanishing gradient problem. Overall, ResNet has proved to be an effective and efficient architecture for various image recognition tasks.

## 5.6 MODEL 4 - PROPOSED MODEL (MIXNET):

MixNet [20] is a deep neural network architecture designed for efficient and accurate image classification tasks. It was proposed in the paper "MixConv:Mixed Depthwise Convolutional Kernels" by Ma et al. in 2019.

The MixNet architecture is based on the MobileNetV2 architecture and uses a combination of depthwise separable convolutions and regular convolutions to reduce the number of parameters and computational cost. The key innovation of MixNet is the use of mixed depthwise convolutional kernels, which are a combination of different kernel sizes in the depthwise convolutional layer. This allows MixNet to achieve high accuracy with fewer parameters compared to other state-of-the-art models

The MixNet architecture consists of a stem block, followed by a sequence of MixConv blocks, and ends with a global average pooling and a fully connected layer. The stem block consists of a regular convolutional layer followed by a batch normalization layer and a ReLU activation function. The MixConv block consists of a mixed depthwise convolutional layer, followed by a batch normalization layer, a ReLU activation function, and a 1x1 convolutional layer to combine the output of the depthwise convolution with the output of the previous block. The global average pooling layer is used to reduce the spatial dimensions of the feature maps to a vector, which is then passed through a fully connected layer to produce the final classification output.

The MixNet architecture comes in different variants, depending on the number of blocks and the scaling factor. The authors of the paper have shown that MixNet achieves state-of-the-art performance on image classification benchmarks while being computationally efficient and having a small memory footprint

## 5.7 COMPARE OF MODEL

Among the Densenet121, Xception, ResNet50, and MixNet-m models, the MixNet-m model stands out as the best performer with a 94% accuracy rate. This is a significant improvement over the other models, with ResNet50 coming in second place with an accuracy rate of about 92%. Densenet121 and Xceptionalso had good accuracy rates of approximately 89%. It is important to note that while accuracy is a key performance metric, other factors may be important to consider for specific applications. However, in the context of crop disease classification, accuracy is a critical metric, and the higher accuracy achieved by MixNet-m makes it a promising choice for practical implementation. Overall, the MixNet-m model has demonstrated superior performance in crop disease classification compared to the other models, indicating its potential for real-world applications.Further research and optimization of MixNet-m could lead to even better results,making it an exciting area of study for crop disease identification and treatment

## 5.8 EXPERIMENTAL RESULT:

Training accuracy is the accuracy of the model on the training data set. During the training phase, the model is trained on a set of labeled images, and the accuracy is calculated based on how many of those images the model correctly classifies. The training accuracy can be a useful metric for understanding how well the model is learning to classify the images in the training set. However, itis important to keep in mind that a high training accuracy does not necessarily mean that the model will perform well on new, unseen images. Validation accuracy is the accuracy of the model on a separate validation data set. During the training phase, a portion of the training data set is typically set aside as a validation set. The model is not trained on this validation set, but rather it is used to evaluate how well the model is generalizing to new, unseen images.

The validation accuracy is a more reliable metric for evaluating the performance of the model, as it measures how well the model is likely to perform on new, unseen images. Ideally, the validation accuracy should be close to the training accuracy, indicating that the model is not overfitting to the training data set. Overfitting occurs when the model becomes too specialized to the training data and fails to generalize to new data. If the training accuracy is much higher than the validation accuracy, it may be a sign that the model is overfitting.

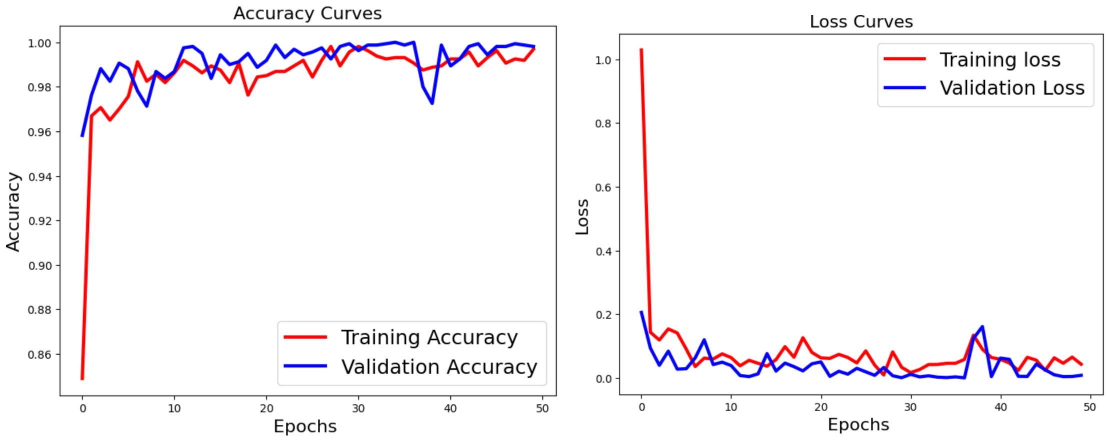



**FIGURE 5.1 VALIDATION LOSS AND TRAINING ACCURACY**

### 5.8.1 Comparative Analysis of Image Classification Model:

Training and validation accuracy [21] are crucial metrics for evaluating an image classification model's performance. Training accuracy measures the accuracy of the model during the training phase, where the model learns from the training data. Validation accuracy measures the accuracy of the model on the unseen data during the validation phase, where the model's ability to generalize to new data is tested.

In this comparison of four transfer learning algorithms, Mixnet m had the highest accuracy with an accuracy of 98.12%. However, without information on the training accuracy, it is challenging to evaluate the models' overall performance. A high training accuracy and low validation accuracy may indicatethat the model is overfitting the training data, while low training accuracy may indicate that the model requires more training data or adjustments. In conclusion, both training and validation accuracy should be considered when evaluating the performance ofan image classification model. While a high validation accuracy is desirable, it is important to ensure that the model is not overfitting the training data. Therefore, it is essential to evaluate both training and validation accuracy to determine the model's overall performance.

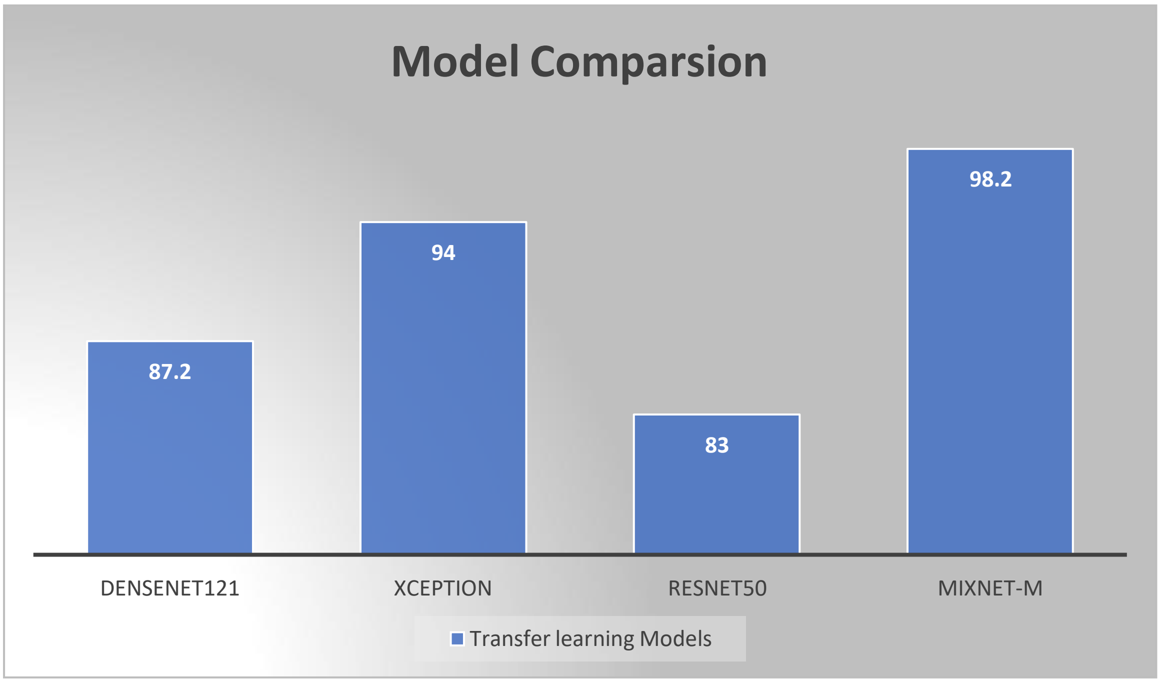



**FIGURE 5. 2 MODEL COMPARISON**

Accuracy: The metric measures the proportion of correct classifications made by the model. It is a simple and easy-to-understand metric but can be misleading if the dataset is imbalanced (i.e., one class has a much higher frequency than the other).

Precision: This metric measures the proportion of positive predictions that are correct. In the context of Jatropha leaf disease classification, this would mean the proportion of leaf samples predicted to have the disease that actually have the disease. Precision is a useful metric when the cost of false positives (i.e., predicting a leaf to have the disease when it does not) is high.

Recall: This metric measures the proportion of actual positive cases that are correctly identified by the model. In the context of Jatropha leaf disease classification, this would mean the proportion of leaf samples that actually have the disease that are correctly identified by the model. Recall is a useful metric when the cost of false negatives (i.e., predicting a leaf does not have the disease when it does) is high.

F1 score: The performance metric commonly used to evaluate the overall effectiveness of a model. It is calculated as the harmonic mean of precision and recall and provides a balanced view of the two metrics. As a single metric, it offers an insight into the model's performance, and a higher F1 score indicates a better model performance.

Confusion Matrix: A confusion matrix is a table used to evaluate the performance of a classification model by comparing the predicted class labels with the true class labels of a set of samples. In the context of Jatropha leaf disease classification, the confusion matrix would have two classes: healthy leaf and diseased leaf. The four possible outcomes of a binary classification problem

are true positive (TP), false negative (FN), false positive (FP), and true negative (TN) True Positive (TP): This occurs when the model predicts that a sample belongsto a certain class and the sample belongs to that class. For Jatropha leaf disease classification, a true positive would be when the model predicts that a leaf is diseased and the leaf is diseased.

True Negative (TN): This occurs when the model predicts that a sample does not belong to a certain class and the sample does not belong to that class. For Jatropha leaf disease classification, a true negative would be when the model predicts that a leaf is healthy and the leaf is healthy.

False Positive (FP): This occurs when the model predicts that a sample belongs to a certain class, but the sample does not belong to that class. For Jatropha leaf disease classification, a false positive would be when the model predicts that a leaf is diseased, but the leaf is healthy.

False Negative (FN): This occurs when the model predicts that a sample does not belong to a certain class, but the sample belongs to that class. For Jatropha leaf disease classification, a false negative would be when the model predicts that a leaf is healthy, but the leaf is diseased.

**TABLE 5.1 PERFORMANCE OF THE TRANSFER LEARNING MODELS**

| Transfer Learning Models | Accuracy | Precision | Recall | F1 Score |
|---|---|---|---|---|
| DenseNet-121 | 87.2 | 86 | 85.2 | 86.8 |
| Xception | 94 | 93 | 93 | 95.6 |
| Resnet50 | 83 | 84 | 82 | 83 |
| MixNet-X | 98.2 | 98 | 97.8 | 98 |

## 5.9 CONCLUSIONS

In conclusion, this study compared various transfer learning algorithms for their ability to accurately identify leaf diseases. The proposed MixNet -M model demonstrated excellent performance, outperforming the other transfer learning algorithms and achieving a maximum accuracy rate of 98.2% in classifying leaf disease. Furthermore, the MixNet -M model required significantly less computation time when compared to the Deep Convolutional Neural Network (DCNN). The findings suggest that MixNet -M can be a highly effective and efficient model for identifying leaf diseases, with potential applications in the field of agriculture and crop management.

# CHAPTER 6

# PESTICIDES RECOMMENDATIONS SYSTEM WITH WEB APPLICATION

## 6.1 INTRODUCTION

A pesticides recommendations system with a web application is a solution that can provide farmers with personalized recommendations on pesticide use based on their specific crop type, location, and pest/disease issues. This system uses machine learning algorithms and data analytics to analyze historical crop data, weather patterns, soil type, and other relevant factors to provide customized recommendations on which pesticide to use, when to apply it, and how much to apply. By minimizing the negative impact of pesticides on human health and the environment, this system has the potential to benefit farmers, their customers, and the sustainability of the agricultural industry.

## 6.2 CASE STUDY

### 6.2.1 Background

Pesticides are an essential tool for farmers to protect their crops from pests and diseases. However, their improper use can lead to adverse effects on human health and the environment. Therefore, it is crucial to provide farmers with accurate and personalized recommendations on pesticide use that take into account their specific crop type, location, and pest/disease issues.

### 6.2.2 Challenge

The challenge in developing a pesticides recommendations system with a web application was to leverage machine learning algorithms and data analytics to analyze historical crop data, weather patterns, soil types, and other relevant factors to generate customized recommendations for each farmer. The system also needed to be easily accessible to farmers, providing them with information on which pesticide to use, when to apply it, and how much to apply.

### 6.2.3 Solution

To address this challenge, a team of developers built a pesticides recommendations system with a web application using Python programming language and Flask web framework. The system consisted of a database of historical crop data, weather patterns, soil types, and other relevant factors that were used to generate personalized recommendations for farmers. The deep learning algorithms used in the system included CNN, Transfer Learning, and Hybrid Mode. These algorithms were trained on the historical data to generate accurate recommendations for each farmer's specific crop type, location, and pest/disease issues. The web application component of the system allowed farmers to access the recommendations from anywhere and at any time. Farmers could input their crop type, location, and pest/disease issues, and the system would generate a customized recommendation based on the historical data and machine learningalgorithms.

### 6.2.4 Results

The pesticides recommendations system with a web application has been successfully deployed and has provided farmers with accurate and personalized recommendations on pesticide use. The system has helped to minimize the negative impact of pesticides on human health and the environment by reducing the amount of pesticides used in agriculture. It has also helped farmers to save time and money by using pesticides more efficiently.

### 6.2.5 Conclusion

The development of a pesticides recommendations system with a web application is a significant step forward in sustainable agriculture. By providing farmers with accurate and personalized recommendations on pesticide use, the system has the potential to significantly reduce the amount of pesticides used in agriculture while also minimizing their negative impact on human health and the environment. This would not only benefit farmers and their customers but also contribute to the sustainability of the agricultural industry.

## 6.3 RECOMMENDATIONS SYSTEM

A fertilizer recommendation [22] system is a tool that provides guidance on the appropriate type and amount of fertilizer to apply to crops for optimal growthand yield. These systems typically take into account various factors such as soil type, crop type, climate, and nutrient levels to generate personalized fertilizer recommendations for farmers. The goal of these systems is to help farmers optimize their fertilizer usage, reduce costs, and improve crop yields while minimizing environmental impact. Fertilizer recommendation systems may use a variety of techniques, including machine learning algorithms, to analyse and process data and generate accurate recommendations.

### 6.3.1 Tea Plant Natural Fertilizer

Tea cultivation demands a judicious use of fertilizers to ensure optimal plant nutrition and yield [23]. While chemical fertilizers have traditionally been a common option, the use of natural fertilizers has gained traction owing to their ecological benefits and slow-release properties. Natural fertilizers, derived from organic materials such as compost, manure, fish emulsion, seaweed, and bone meal, offer a sustainable and effective approach to enhance soil fertility and plant growth. In this discourse, we will delve into the various natural fertilizers that can be used for tea plant cultivation, highlighting their key nutrient compositions and optimal application methods to maximize plant health and yield.

- Compost: Compost is a mixture of organic materials that have decomposed and broken down into a nutrient-rich soil amendment. Composting tea leaves, grass clippings, vegetable scraps, and other organic matter can create a compost that's high in nitrogen, phosphorus, and potassium - three essential nutrients that tea plants need to grow and thrive. To use compost as a fertilizer for tea plants, simply spread it around the base of the plant and work it into the soil with a garden fork orhoe. It's important to avoid piling the compost up against the trunk of the plant, as this can lead to stem rot.

- Manure: Well-aged animal manure, such as cow or horse manure, can be used as a natural fertilizer for tea plants. Manure is rich in nitrogen, phosphorus, and potassium, and can help improve soil fertility and increase the growth and yield of tea plants. However, it's important to use only well-composted manure, as fresh manure can be too hot and can burn the plant roots. To use manure as a fertilizer for tea plants, spread it around the base of the plant and work it into the soil.
- Fish emulsion: Fish emulsion is a liquid fertilizer made from fish waste that's been broken down into a nutrient-rich solution. It's high in nitrogen, phosphorus, and potassium, and can be applied directly to the soil around tea plants. Fish emulsion is also a good source of trace minerals that tea plants need, such as calcium and magnesium. To use fish emulsion as a fertilizer for tea plants, dilute it according to the manufacturer's instructions and apply it to the soil around the base of the plant.
- Seaweed: Seaweed can be used as a natural fertilizer because it's rich in potassium, nitrogen, and other minerals that can benefit tea plants. Seaweed can be added to the soil around tea plants or used as a foliar spray. When used as a foliar spray, seaweed can help improve the plant's ability to absorb nutrients from the soil and can also help the plant resist pests and disease. To use seaweed as a fertilizer for tea plants, either mixit into the soil or dilute it and apply it as a foliar spray.
- Bone meal: Bone meal is made from ground-up animal bones and is a good source of phosphorus, which is important for root development in tea plants. It can be added to the soil around tea plants to help improve soil fertility and increase plant growth. Bone meal is also a good source of

calcium, which tea plants need to build strong cell walls. To use bone meal as a fertilizer for tea plants, simply sprinkle it around the base of theplant and work it into the soil with a garden fork or hoe.

### 6.3.2 Tea Plant Natural Pesticides

Tea plants are a valuable crop [24] that requires careful attention to prevent pestinfestations, which can cause damage and reduce yields. While chemical pesticides have been traditionally used to control pests, their long-term effects on the environment and human health have raised concerns. Fortunately, natural pesticides offer a safer and more sustainable alternative to control pests in tea plantations. Natural pesticides are derived from organic materials and are less toxic than chemical pesticides. They work by targeting specific pests and do not harm beneficial insects, such as bees and butterflies, which play a critical role in pollination. In this article, we will discuss some effective natural pesticides for tea plants, their modes of action, and how to use them safely and effectively to control pests and promote a healthy tea plantation

- Neem oil: Neem oil is a potent natural insecticide that disrupts the hormonal balance of insects and inhibits their ability to feed and reproduce. It is effective against a wide range of pests that attack tea plants, including aphids, mites, and caterpillars. To use neem oil, mix it with water as per the manufacturer's instructions and apply it as a spray tothe affected parts of the tea plant. It is important to note that neem oil mayalso kill beneficial insects like bees, so it should be used with caution.
- Garlic spray: Garlic is a natural insecticide that contains sulfur compounds that repel pests. It is effective against aphids, whiteflies, and spider mites that commonly attack tea plants. To make garlic spray, crusha few garlic cloves and steep them in hot water for several hours. Strain the mixture and dilute it with water before applying it as a spray to the teaplant.
- Soap spray: Soap spray is a natural pesticide that can control soft-bodied

insects like aphids and spider mites. It works by suffocating the pests and inhibiting their breathing. To make soap spray, mix a few drops of liquid soap with water and apply it as a spray to the affected parts of the tea plant. It is essential to use mild soap, as harsh detergents can damage the plant.

- Oil spray: Oil spray is a natural pesticide that suffocates and kills pests like mites and scale insects. To make oil spray, mix vegetable oil with liquid soap and water and apply it as a spray to the affected parts of the tea plant.
- Diatomaceous earth: Diatomaceous earth is a natural insecticide that dehydrates insects and causes them to die. It is made from the fossilized remains of diatoms, a type of algae. To use diatomaceous earth, sprinkle a thin layer of it around the base of the tea plant and on the leaves and stems. It is important to note that diatomaceous earth may also kill beneficial insects like bees, so it should be used with caution.

### 6.4 CLOUD – AWS

Amazon Web Services (AWS) is a cloud computing platform[25] that provides a wide range of services, including computing power, storage, and databases, among others, to individuals, organizations, and governments. It allows users to easily access and use scalable, reliable, and cost-effective infrastructure resources without having to invest in physical hardware. With its global network of data centers and high-speed connectivity, AWS enables users todeploy and manage applications and services quickly and efficiently, with the added benefits of security, flexibility, and innovation. Whether you're a startup or a large enterprise, AWS offers a comprehensive suite of tools and services to help you meet your business needs and stay ahead of the competition Implementing an image classification model on AWS can be achieved by leveraging AWS services such as Amazon SageMaker, Amazon Recognition, or AWS Deep Learning AMIs. These services provide a range of pre-built algorithms, development tools, and data sets that can be used to train and deploy an image classification model quickly and efficiently. To get started, you'll typically need to prepare your data, train your model using machine learning algorithms, and then deploy the model to an AWS instance. Once deployed, the model can be used to classify new images in real-time, helping to automate processes and improve decision-making. The server response time for an image classification model on AWS can vary depending on a range of factors, such as the complexity of the model, the size of the dataset, and the compute resources allocated to the instance. Generally, AWS provides high-performance computing and storage resources that are optimized for machine learning workloads, enabling rapid processing of large volumes of data. With services such as Amazon SageMaker, developers can easily build and deploy machine learning models on AWS instances that offer low-latency, high-throughput performance. Additionally, AWS offers auto-scaling capabilities that can automatically adjust the compute and storage resources available to the model based on workload demands, ensuring consistent and reliable performance. Overall, the server response time for an image classification model on AWS can be very fast, providing real-time insights that can help businesses automate processes and make better decisions.

The advantages of implementing an image classification model on AWS are numerous. Firstly, AWS provides a cost-effective solution to building and deploying machine learning models, allowing businesses to save on infrastructure costs. Additionally, AWS offers scalable computing and storage resources that can easily accommodate the needs of growing businesses. AWS also provides a high level of security and reliability, ensuring that your data is protected and your applications are always available. With AWS, you can leverage the power of cloud computing to build and deploy a robust image classification solution that meets your business needs.

## 6.5 USER INTERFACE:

To provide a seamless user experience, it's important to create two types of applications - mobile-based and web-based - with a minimalist user interface that can be easily connected to the cloud using APIs. By doing so, you can ensure that your users can access your application from any device, whether it's a mobilephone or a computer, and have a consistent and intuitive experience.

To connect your user interface with the cloud, you can use services such as Amazon API Gateway, AWS Lambda, or AWS Amplify. These services allow you to build and deploy APIs and serverless functions that can retrieve information from the cloud and deliver it to your user interface in real-time.

By using a minimalist user interface, you can ensure that your application is easy to use and navigate, providing a frictionless experience for your users. Additionally, leveraging the power of the cloud can offer a range of benefits, including scalability, reliability, and cost-effectiveness. By storing your application data in the cloud, you can easily scale up or down as needed to meet changing demands, while also ensuring that your data is always secure and available. Overall, by creating mobile and web-based applications with a minimalist user interface that are connected to the cloud through APIs, you can build a professional and seamless application that delivers real value to your users. A minimalist user interface provides an enhanced user experience by reducing clutter, simplifying navigation, and making it easier for users to focus on the core features of an application. By minimizing distractions and unnecessary elements, users can more easily understand and interact with the interface, which can lead to increased engagement and satisfaction. Additionally,a minimalist design can make an application more accessible and intuitive, reducing the learning curve for new users and allowing them to quickly accomplish their tasks. Overall, a minimalist user interface can help create a moreenjoyable and effective user experience, leading to increased usage and loyalty.

## 6.6 CONCLUSIONS

A pesticides recommendation system with a web application can help farmers make informed decisions about the type and amount of pesticides to use on their crops. By taking into account various factors such as crop type, pest type, and weather conditions, the system can generate personalized recommendations that help farmers optimize pesticide usage and reduce costs while minimizing environmental impact. The web application provides an easy-to-use interface for farmers to access and implement the recommendations, making it a convenient and practical tool for modern farming. Overall, a pesticides recommendation system with a web application is a valuable resource that can enhance the effectiveness and sustainability of pesticide usage in agriculture.

# CHAPTER 7

# IMPLEMENTATION AND CODE

## 7.1 PYTHON:

The AI Model software is implemented using python. Python is a high-level interpreted, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. The python version used for this software is 3.9.2.

Python modules used in the software:

- Numpy
- Matplotlib
- Pandas
- Torch
- Torchvision
- Twilio
- .Smtplib
- Openpyxl
- Opencv

## 7.2 REACT JS:

ReactJS is a popular JavaScript library for building user interfaces (UIs) for webapplications. Developed by Facebook, it allows developers to create reusable UIcomponents that can be used to build complex and dynamic applications.
ReactJS uses a declarative approach to programming, where developers specify

how the UI should look and behave, and React handles the updates and rendering of the components. This makes it easier to manage large and complex codebases,and enables faster development and debugging.

ReactJS also features a virtual DOM (Document Object Model), which is an in-memory representation of the actual DOM. This enables React to efficiently update the UI by only rendering the parts that have changed, rather than re-rendering the entire page. React also supports server-side rendering, which improves the performance and search engine optimization (SEO) of web applications. Other features of React include its ability to work with other libraries and frameworks, its strong developer community and ecosystem, and its focus on performance and scalability.

### 7.3 HARDWARE REQUIREMENTS:

Operating System: Ubuntu – 18.0.3 LTS

RAM: 4 GB

GPU ( CUDA 10.2 Supported): 2GB (optional)

Internet Required

# CHAPTER 8

## 8.1 CONCLUSION

This study demonstrates the potential of AI-based frameworks in agriculture for crop disease identification and fertilizer recommendation. The proposed MixNet-M model achieved a maximum accuracy rate of 98.2% and required less computation time than other algorithms. The React.js and React Native framework provides a scalable solution for farmers of all sizes. The study highlights the importance of collecting and pre-processing datasets for developing reliable and robust AI-based systems to optimize crop yield and quality. The use of natural pesticides can make agriculture more sustainable and eco-friendlier. Further research is needed to improve existing systems and develop AI-based frameworks for other crops, but overall, AI-based frameworks can revolutionize agriculture by providing fast and accurate identification of crop diseases and recommendations for treatment.

## 8.2 FUTURE SCOPUS:

In Future Scopus we are planning to improve the Fertilizer Recommendation with the Detail Explanation and process of the Fertilizer Manufacturing in Home itself and we are required to collect and improve the Real Time leaf image process by deploying the maximum accuracy in disease classification.

# APPENDIX 1

## CODING

Python App.py

```
from flask import Flask, request, jsonify
from werkzeug.utils import secure_filename
import os
from flask_cors import CORS
from flask_pymongo import PyMongo
from flask_bcrypt import Bcrypt
from bson.objectid import ObjectId
import jwt
import datetime
import io
import cv2
import string
import numpy as np
import torch
import torch.nn as nn

from flask import Flask, jsonify, render_template, request, redirect
from torch.autograd import Variable
from torchvision import models, transforms
```

```
from PIL import Image


app = Flask(__name__)

CORS(app,resources={r'/': {'origins': ''}})


@app.after_request

def add_cors_headers(response):

    response.headers['Access-Control-Allow-Origin'] = '*'

    response.headers['Access-Control-Allow-Headers'] = 'Content-Type'
    return response

app.config['MONGO_URI'] =
'mongodb+srv://rajeshkumar:9787234640@cluster0.inzkc.mongodb.net/Image
App'

mongo = PyMongo(app)

bcrypt = Bcrypt(app)

app.config['SECRET_KEY'] = 'thisisthesecretkey'


def create_token(user_id):

    payload = {'user_id': str(user_id), 'exp': datetime.datetime.utcnow() +
datetime.timedelta(minutes=30)}

    token = jwt.encode(payload, app.config['SECRET_KEY'])

    return token
```

```
@app.route('/login', methods=['POST'])

def login():

    print(request)

    user = mongo.db.users.find_one({'email': request.json['email']})

    if user and bcrypt.check_password_hash(user['password'],
request.json['password']):

        token = create_token(user['_id'])

        return jsonify({'token': token}), 200
    else:

        return jsonify({'message': 'Invalid credentials'}), 401


@app.route('/register', methods=['POST'])

def register():

    user = mongo.db.users.find_one({'email': request.json['email']})

    if user:

        return jsonify({'message': 'User already exists'}), 400

    else:

        password =
bcrypt.generate_password_hash(request.json['password']).decode('utf-8')

        user_id = mongo.db.users.insert_one({'email': request.json['email'],
'password': password})

        token = create_token(user_id)

        return jsonify({'token': token}), 201
```

```
MODEL_PATH = "./model/dense_net.pt"

device = torch.device("cpu")

class_names = [

    "Bean_Angular_Leaf_Spot",

    "Bean_Heathly",

    "Bean_Rust",

    "Cotton_Disease",

    "Cotton_Healthy",

    "Cucumber_Disease",

    "Cucumber_Healthy",

    "Guava_Disease",

    "Guava_Healthy",

    "Lemon_Diseased",

    "Lemon_Healthy",

    "Mango_Disease",

    "Mango_Healthy",

    "Pomegranate_Diseased",

    "Pomegranate_Healthy",

    "Potato_Early_Blight",

    "Potato_Healthy",

    "Potato_Late_Blight",

    "Tea_Disease",
```

```
    "Tea_Healthy",
    "Tomato___Bacterial_spot",
    "Tomato___Early_blight",
    "Tomato___healthy",
    "Tomato___Late_blight",
    "Tomato___Leaf_Mold",
    "Tomato___Septoria_leaf_spot",
    "Tomato___Spider_mites Two-spotted_spider_mite",
    "Tomato___Target_Spot",
    "Tomato___Tomato_mosaic_virus",
    "Tomato___Tomato_Yellow_Leaf_Curl_Virus",
]


# Defining Model Architecture
def CNN_model(pretrained):
    inf_model = models.densenet121(pretrained=pretrained)
    inf_model.classifier.in_features = len(class_names)
    inf_model.to(torch.device("cpu"))
    return inf_model


inf_model = CNN_model(pretrained=False)


# Loading the Model Trained Weights
```

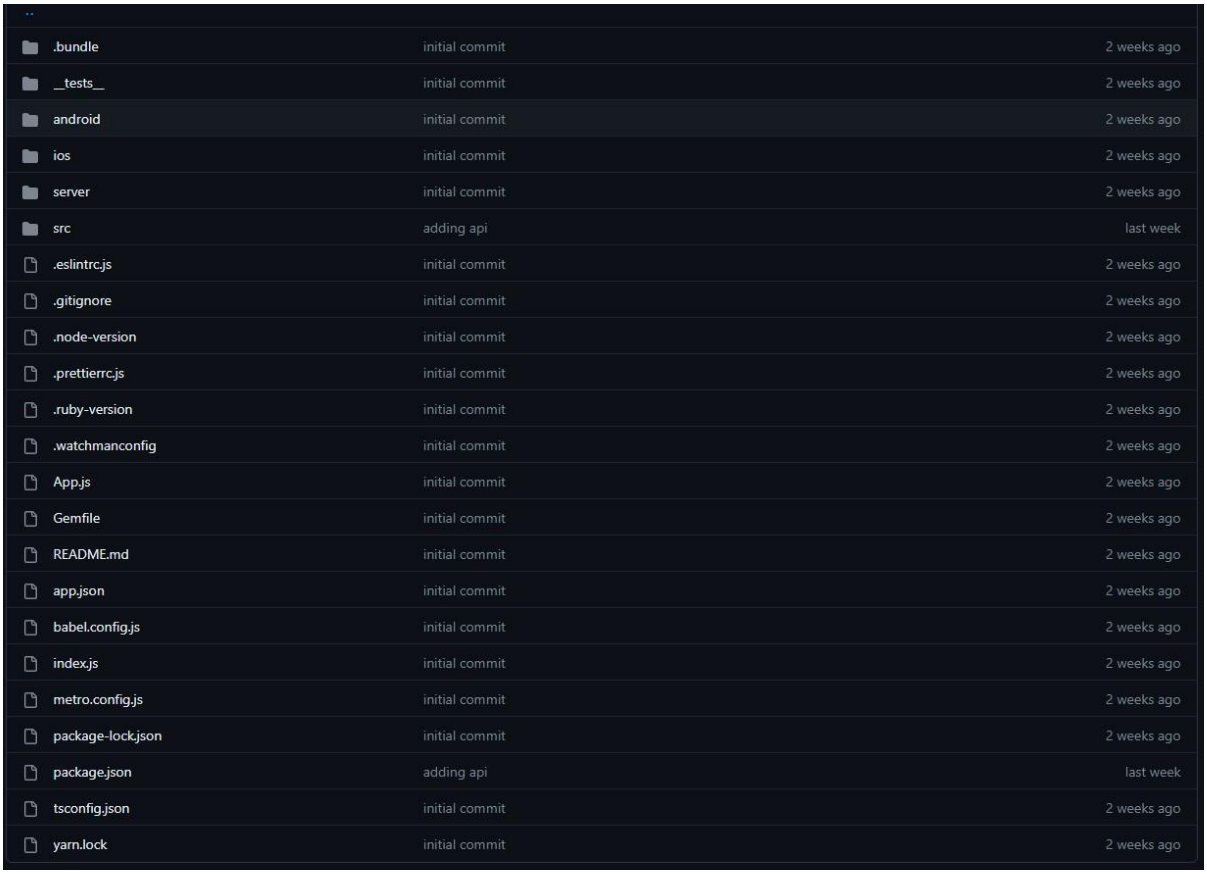
```
inf_model.to(torch.device("cpu"))
inf_model.load_state_dict(torch.load(MODEL_PATH, map_location="cpu"))
inf_model.eval()
print("Inference model Loaded on CPU")

# Image Transform
def transform_image(image_bytes):
    test_transforms = transforms.Compose(
        [
            transforms.ToPILImage(),
            transforms.Resize((224, 224)),
            transforms.ToTensor(),
            transforms.Normalize((0.485, 0.456, 0.406), (0.229, 0.224, 0.225)),
        ]
    )
    image = Image.open(io.BytesIO(image_bytes))
    image = np.array(image)
    if image.shape[-1] == 4:
        image_cv = cv2.cvtColor(image, cv2.COLOR_RGBA2RGB)
    if image.shape[-1] == 3:
        image_cv = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
    if len(image.shape) == 2:
        image_cv = cv2.cvtColor(image, cv2.COLOR_GRAY2RGB)
```

```
    return test_transforms(image_cv).unsqueeze(0)

def get_prediction(image_bytes):
    tensor = transform_image(image_bytes=image_bytes)
    outputs = inf_model.forward(tensor)
    _, prediction = torch.max(outputs, 1)
    print(prediction)
    return class_names[prediction]


@app.route("/api/upload-image", methods=["POST"])
def upload_image():
    image = request.files.get('image')
    if not image:
        return
    img_bytes = image.read()
    prediction_name =  get_prediction(img_bytes)
    return jsonify({'message':prediction_name.lower()})
    # filename = secure_filename(image.filename)
    # image.save(os.path.join("./uploads", filename))
@app.route('/')
def index():
    return jsonify({'message': 'Success'})
```

```
if __name__ == '__main__':
    app.run(debug=True,host='0.0.0.0')
```

7.3.2 Requirements.txt

```
torchvision==0.6.1+cpu

torch==1.5.1+cpu

numpy==1.17.2

opencv_contrib_python==4.2.0.34

Pillow==8.1.1

Click==7.0

Flask==1.1.1

gunicorn==19.9.0

itsdangerous==1.1.0

Jinja2==2.11.3

MarkupSafe==1.1.1

Werkzeug==0.15.6
```



| | | |
|---|---|---|
| .bundle | initial commit | 2 weeks ago |
| __tests__ | initial commit | 2 weeks ago |
| android | initial commit | 2 weeks ago |
| ios | initial commit | 2 weeks ago |
| server | initial commit | 2 weeks ago |
| src | adding api | last week |
| .eslintrc.js | initial commit | 2 weeks ago |
| .gitignore | initial commit | 2 weeks ago |
| .node-version | initial commit | 2 weeks ago |
| .prettierrc.js | initial commit | 2 weeks ago |
| .ruby-version | initial commit | 2 weeks ago |
| .watchmanconfig | initial commit | 2 weeks ago |
| App.js | initial commit | 2 weeks ago |
| Gemfile | initial commit | 2 weeks ago |
| README.md | initial commit | 2 weeks ago |
| app.json | initial commit | 2 weeks ago |
| babel.config.js | initial commit | 2 weeks ago |
| index.js | initial commit | 2 weeks ago |
| metro.config.js | initial commit | 2 weeks ago |
| package-lock.json | initial commit | 2 weeks ago |
| package.json | adding api | last week |
| tsconfig.json | initial commit | 2 weeks ago |
| yarn.lock | initial commit | 2 weeks ago |

Fig 8.1 Structure of the Folder

## APP RESULT:

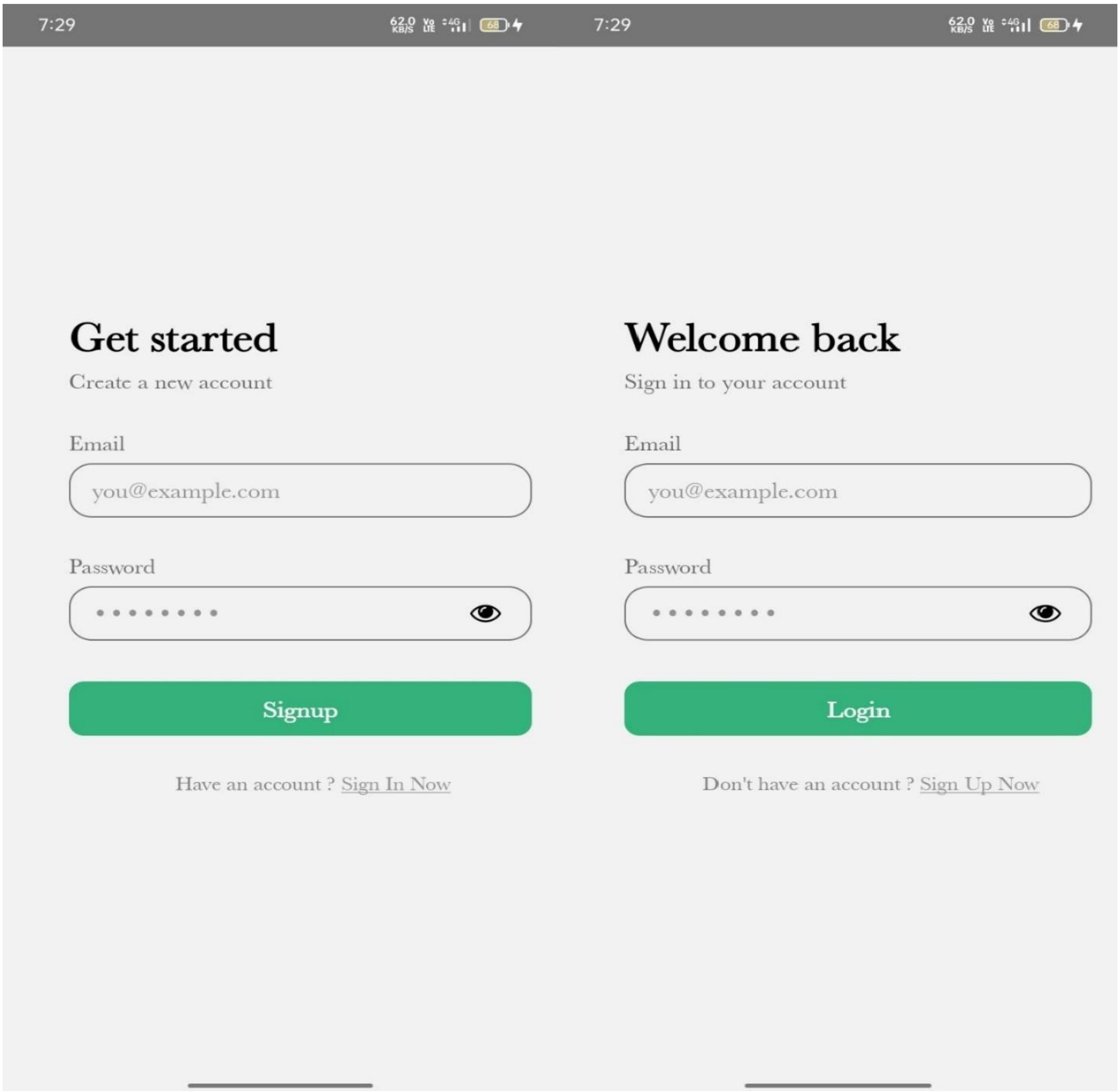



Fig 8.2 Sign Up and Login Page

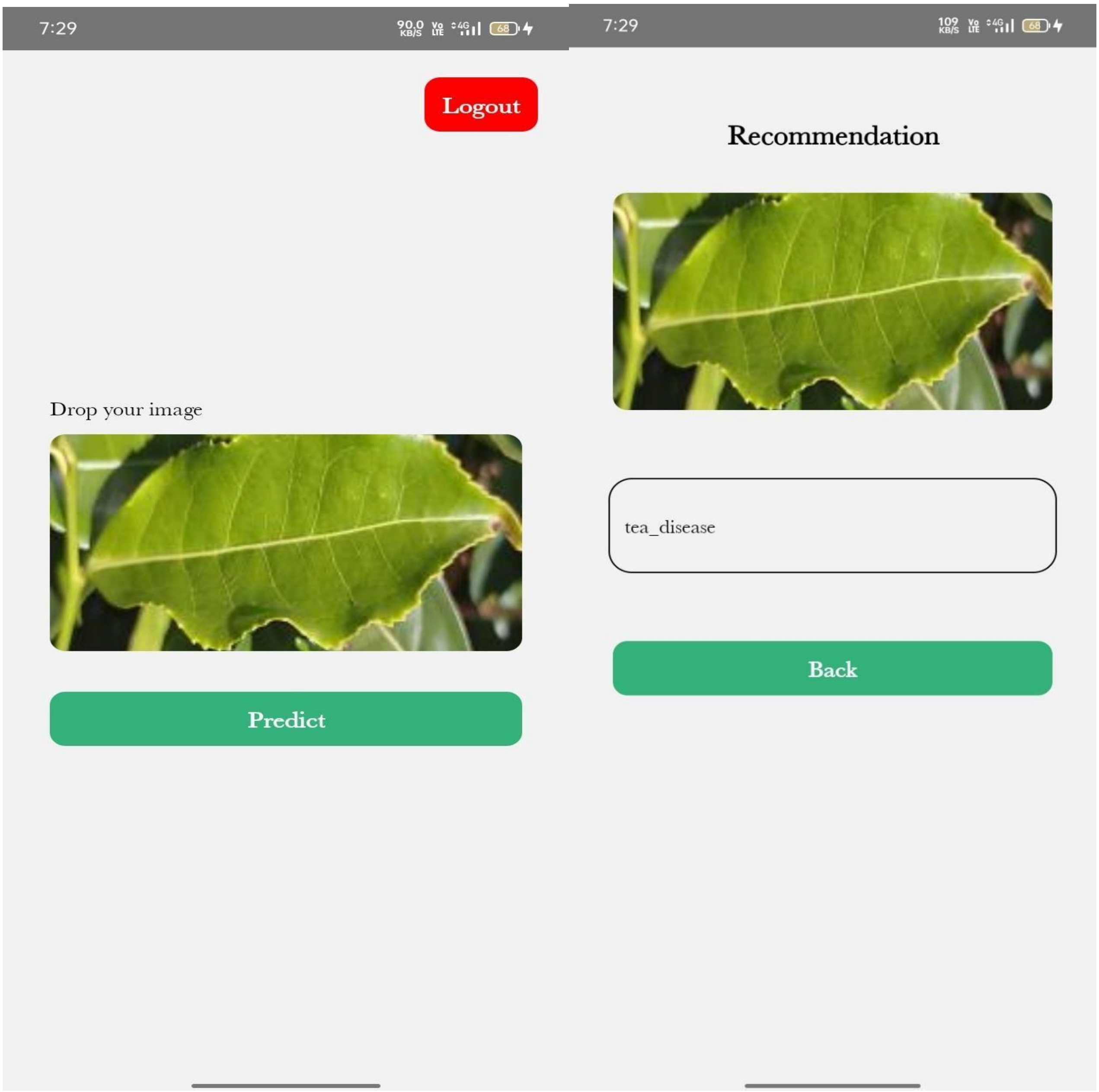



Fig 8.3 Predict Page and Disease Page


Fig 8.4 Recommendation Page

## REFERENCE:

[1] X. -W. Chen and X. Lin, "Big Data Deep Learning: Challenges and Perspectives," in IEEE Access, vol. 2, pp. 514-525, 2014, doi: 10.1109/ACCESS.2014.2325029.

[2] P. J. Werbos, "An overview of neural networks for control," in IEEE …. Control Systems Magazine, vol. 11, no. 1, pp. 40-41, Jan. 1991, doi: 10.1109/37.103352.

[3] Z. Li, F. Liu, W. Yang, S. Peng and J. Zhou, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," in IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 12, pp. 6999-7019, Dec. 2022, doi: 10.1109/TNNLS.2021.3084827.

[4] F. Zhuang et al., "A Comprehensive Survey on Transfer Learning," in Proceedings of the IEEE, vol. 109, no. 1, pp. 43-76, Jan. 2021, doi: 10.1109/JPROC.2020.3004555.

[5] M. Ravanelli, T. Parcollet and Y. Bengio, "The Pytorch-kaldi Speech Recognition Toolkit," ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 2019, pp. 6465-6469, doi: 10.1109/ICASSP.2019.8683713.

[6] Saleem, M.H., Potgieter, J. & Arif, K.M. Automation in Agriculture by Machine and Deep Learning Techniques: A Review of Recent Developments. Precision Agric 22, 2053–2091 (2021). https://doi.org/10.1007/s11119-021-09806-x

[7] Poornima Singh Thakur, Pritee Khanna, Tanuja Sheorey, Aparajita Ojha, Trends in vision-based machine learning techniques for plant disease identification: A systematic review, Expert Systems with Applications,Volume 208,2022,118117,ISSN 0957-4174,https://doi.org/10.1016/j.eswa.2022.118117.

[8] Tawseef Ayoub Shaikh, Tabasum Rasool, Faisal Rasheed Lone,Towards leveraging the role of machine learning and artificial intelligence in precision agriculture and smart farming, Computers and Electronics iAgriculture,Volume198,2022,107119,ISSN 0168

[9] Vol. 36 No. 11: IAAI-22, EAAI-22, AAAI-22 Special Programs and Special Track, Student Papers and Demonstrations

[10] Prasad, R. P., Gill, R., Gupta, V., Bordoloi, P., Ahmed, M., & Rao, R. K. (2022). Recent Advances in Agricultural Science and Technology for Sustainable India.

[11] Danish Vasan, Mamoun Alazab, Sobia Wassan, Babak Safaei, Qin Zheng, Image-Based malware classification using ensemble of CNN architectures (IMCEC),Computers & Security,Volume 92,2020,101748,ISSN 0167-4048,https://doi.org/10.1016/j.cose.2020.101748.

[12] Kumar, V., Sinha, D., Das, A.K. et al. An integrated rule based intrusion detection system: analysis on UNSW-NB15 data set and the real time online dataset. Cluster Comput 23, 1397–1418 (2020). https://doi.org/10.1007/s10586- 019-03008-x

[13] Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. Journal of big data, 6(1), 1-48.

[14] K. M. Kahloot and P. Ekler, "Algorithmic Splitting: A Method for Dataset Preparation," in IEEE Access, vol. 9, pp. 125229-125237, 2021, doi: 10.1109/ACCESS.2021.3110745.

[15] K. Sampigethaya and R. Poovendran, "A Survey on Mix Networks and Their

Secure Applications," in Proceedings of the IEEE, vol. 94, no. 12, pp. 2142-2181, Dec. 2006, doi: 10.1109/JPROC.2006.889687.

[16] P. Cao, S. Zhang and J. Tang, "Preprocessing-Free Gear Fault Diagnosis Using Small Datasets With Deep Convolutional Neural Network-Based Transfer Learning," in IEEE Access, vol. 6, pp. 26241-26253, 2018, doi: 10.1109/ACCESS.2018.2837621.

[17] W. Tong, W. Chen, W. Han, X. Li and L. Wang, "Channel-Attention- Based DenseNet Network for Remote Sensing Image Scene Classification," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 4121-4132, 2020, doi: 10.1109/JSTARS.2020.3009352.

[18] B. Chen, X. Liu, Y. Zheng, G. Zhao and Y. -Q. Shi, "A Robust GAN- Generated Face Detection Method Based on Dual-Color Spaces and an Improved Xception," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 32, no. 6, pp. 3527-3538, June 2022, doi: 10.1109/TCSVT.2021.3116679.

[19] I. Z. Mukti and D. Biswas, "Transfer Learning Based Plant Diseases Detection Using ResNet50," 2019 4th International Conference on Electrical Information and Communication Technology (EICT), Khulna, Bangladesh, 2019, pp. 1-6, doi: 10.1109/EICT48899.2019.9068805.

[20] Yanxin Wang, Jing Yan, Zhou Yang, Yiming Zhao, Tingliang Liu,Optimizing GIS partial discharge pattern recognition in the ubiquitous power internet of things context: A MixNet deep learning model,International Journal of Electrical Power & Energy Systems,Volume 125,2021,106484,ISSN 0142-0615,

[21] S. Loyka and G. Tsoulos, "Estimating MIMO system performance using the correlation matrix approach," in IEEE Communications Letters, vol. 6, no. 1, pp. 19-21, Jan. 2002, doi: 10.1109/4234.975485.

[22] Goh, K. J., & Po, S. B. (2005, March). Fertilizer recommendation systems for oil palm: estimating the fertilizer rates. In Proceedings of MOSTA Best

practices workshops-agronomy and crop management. Malaysian Oil Scientists' and Technologists' Association (pp. 1-37).

[23] Chong, K. P., Ho, T. Y., & Jalloh, M. B. (2008). Soil nitrogen phosphorus and tea leaf growth in organic and conventional farming of selected fields at Sabah Tea plantation slope. Journal of Sustainable Development, 1(3), 117-122.

[24] Lin, W., Lin, M., Zhou, H., Wu, H., Li, Z., & Lin, W. (2019). The effects of chemical and organic fertilizer usage on rhizosphere soil in tea orchards. PloSone, 14(5), e0217018.

[25] K. R. Jackson et al., "Performance Analysis of High Performance Computing Applications on the Amazon Web Services Cloud," 2010 IEEE Second International Conference on Cloud Computing Technology and Science, Indianapolis, IN, USA, 2010, pp. 159-168, doi: 10.1109/CloudCom.2010.69

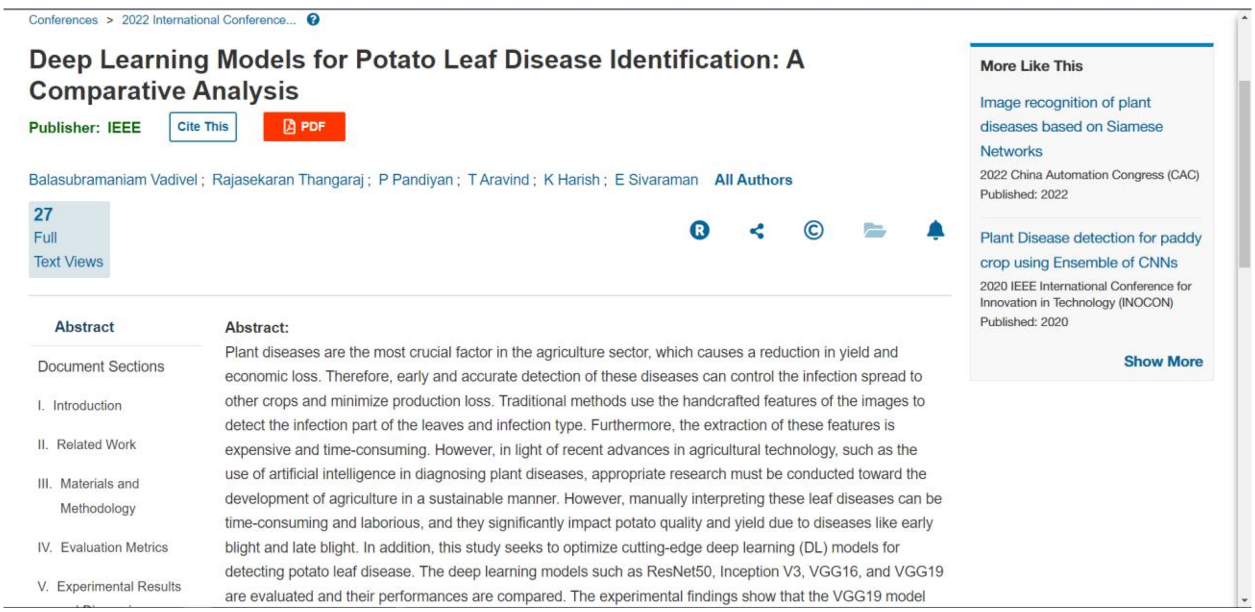
## PAPER -PUBLICATION

- Paper 1 - Deep Learning Models for Potato Leaf Disease Identification: A Comparative Analysis

  Status - Published (IEEE Index)

- Paper 2 - Tea Leaf Disease Identification using Improved Convolution Neural Network

  Status - Accpected on IEEE Index



Conferences > 2022 International Conference...

### Deep Learning Models for Potato Leaf Disease Identification: A Comparative Analysis

Publisher: IEEE | Cite This | PDF

Balasubramaniam Vadivel ; Rajasekaran Thangaraj ; P Pandiyan ; T Aravind ; K Harish ; E Sivaraman All Authors

27 Full Text Views

Abstract

Document Sections

I. Introduction

II. Related Work

III. Materials and Methodology

IV. Evaluation Metrics

V. Experimental Results

**Abstract:**
Plant diseases are the most crucial factor in the agriculture sector, which causes a reduction in yield and economic loss. Therefore, early and accurate detection of these diseases can control the infection spread to other crops and minimize production loss. Traditional methods use the handcrafted features of the images to detect the infection part of the leaves and infection type. Furthermore, the extraction of these features is expensive and time-consuming. However, in light of recent advances in agricultural technology, such as the use of artificial intelligence in diagnosing plant diseases, appropriate research must be conducted toward the development of agriculture in a sustainable manner. However, manually interpreting these leaf diseases can be time-consuming and laborious, and they significantly impact potato quality and yield due to diseases like early blight and late blight. In addition, this study seeks to optimize cutting-edge deep learning (DL) models for detecting potato leaf disease. The deep learning models such as ResNet50, Inception V3, VGG16, and VGG19 are evaluated and their performances are compared. The experimental findings show that the VGG19 model

More Like This

Image recognition of plant diseases based on Siamese Networks
2022 China Automation Congress (CAC)
Published: 2022

Plant Disease detection for paddy crop using Ensemble of CNNs
2020 IEEE International Conference for Innovation in Technology (INOCON)
Published: 2020

Show More

# Deep Learning Models for Potato Leaf Disease Identification: A Comparative Analysis

Balasubramaniam Vadivel
Department of Computer Science and Engineering
KPR Institute of Engineering and Technology
Coimbatore, India
balasubramaniamvadivel@gmail.com

Rajasekaran Thangaraj *
Department of Computer Science and Engineering
KPR Institute of Engineering and Technology
Coimbatore, India
rajasekaran30@gmail.com

Pandiyan P
Department of Electrical and Electronics Engineering
KPR Institute of Engineering and Technology
Coimbatore, India
pandyyan@gmail.com

Aravind T
Department of Computer Science and Engineering
KPR Institute of Engineering and Technology
Coimbatore, India
aravindthangaraj7@gmail.com

Harish K
Department of Computer Science and Engineering
KPR Institute of Engineering and Technology
Coimbatore, India
harishkvel73@gmail.com

Sivaraman E
Senior Lecturer,Computing(Cyber Security),Department of Electrical and Computer Engineering
Curtin University,Malaysia
sivaraman.eswaran@gmail.com

*Abstract*— **Plant diseases are the most important factor in the agriculture sector, which causes a reduction in yield and economic loss. Therefore, early and accurate detection of these diseases can control the infection spread to other crops and ensure the minimization of production loss. Traditional methods use the handcrafted features of the images to detect the infection part of the leaves and infection type. Furthermore, the extraction of these features is expensive and time-consuming. In light of recent advances in agricultural technology, such as the use of artificial intelligence in the diagnosis of plant diseases, it is vital that appropriate research be conducted toward the development of agriculture in a sustainable manner. However, manually interpreting these leaf diseases can be time-consuming and laborious, and they have a significant impact on potato quality and yield due to diseases like early blight and late blight. This study seeks to optimize cutting-edge deep learning (DL) models for detecting potato leaf disease. The deep learning models such as ResNet50, Inception V3, VGG16, and VGG19 are evaluated and their performances are compared. The experimental findings show that the VGG19 model outperforms the other models with an accuracy of 99%.**

***Keywords— Plant Disease Identification, Deep Learning Model, Potato Leaf Disease, Transfer learning, Fine-tuning.***

## I. INTRODUCTION

India's economy is primarily based on agriculture. Agriculture is the primary contributor to India's economic growth, and the country is one of the world's emerging nations. Infectious plant diseases represent a significant threat to human health and cause heavy damage to small farmers. Plants are influenced by various types of diseases, which reduce crop yield and cause significant economic losses to farmers [1].Potato is India's most important agricultural crop, accounting for around 28.9% of the country's total harvest. The potato ranks behind only the cereals maize, wheat, and rice as the world's fourth most important agricultural food crop. India makes 48.5 million tonnes of potatoes every year, making it the world's second-largest country that grows potatoes [2]. Potato output exceeds 0.3 billion metric tonnes worldwide, and it contains a variety of substances such as minerals, vitamins, carbohydrates, and phytochemicals, all of which are essential for assisting and maintaining human health [3]. In general, potato plants are susceptible to two different types of disease: late blight (Phytophthorainfestans) and early blight (Alternariasolani) [4].

Early disease identification is essential for enhancing the quantity and quality of the crop. Nowadays, detection of the disease in its early stages is a challenging task. Furthermore, continuous monitoring of crops in the agricultural field for farmers is also difficult. Plant diseases have been identified by a specialist with the naked eye, but this technique produces infeasible results due to the lack of expertise and too much processing time [5]. So, the automatic detection of diseases using images of leaves has made a lot of effort to find diseases in their early stages so that the right preventive steps can be taken to avoid a huge loss. In recent years, DL has become a subset of machine learning (ML) that does better than other methods in many fields, such as natural language processing, speech recognition, computer vision, image pattern identification, detection, and analysis [6–8]. Furthermore, DL's agricultural accomplishments are noteworthy in a variety of fields, such as automatic plant species classification, disease identification and classification, weed detection, and yield prediction. The main goal of DL is to automatically extract the high-level features from raw data, which used to be done by hand using machine learning methods. In the agricultural domain, it is difficult to obtain large datasets, which allow researchers to adapt pre-trained networks and utilize the transfer learning concept.

Transfer Learning (TL) is one of the deep learning techniques that applies the knowledge learned through solving one problem to other related problems, and it provides a guaranteed result with a small dataset [9]. The TL uses two approaches, such as feature extraction and fine-tuning. The first approach is feature extraction, in which the features obtained from the pre-trained model are utilized to train the images of the new datasets. Another one is to fine-tune the weights of the network by retraining the network with the new dataset. Subsequently, the number of classes at the output layer is modified to match the number of classes in the target dataset [10, 11]. This paper focus on the detection of potato leaf diseases by introducing the DL models namely Inception V3,ResNet50, VGG16, and VGG19. The work mainly contributes to improve the accuracy of leaf disease recognition by applying the TL method.

The rest of this work is organized as mentioned below: Section 2 discusses the related work conducted for plant disease identification. The image datasets used for training and testing, and the methodology applied for potato leaf disease classification are presented in Section 3. Section 4 indicates the metrics for performance evaluation of the model. In Section 5, the results of the experiments are given, and in Section 6, a conclusion is drawn.

## II. Related Work

Recently, research on Convolution Neural Network (CNN) based DL models has been widely employed to resolve issues pertaining to agricultural areas like weed detection, plant disease identification, pest recognition in crops and fruit classification [12]. The plant disease identification is done through the CNN model by training the network by means of images taken from the healthy and disease-affected plants. This section deals with the work that was done to use leaf images and the CNN model to figure out which plants were affected by diseases.

Subetha et al. [13] compared two different deep learning algorithms, namely VGG19 and ResNet50. Their model was allowed to identify four different types of apple leaf disease with a classification accuracy of 87.7%. Karlekar and Seal [14] used two modules to correctly diagnose diseases in soybean leaves. The first module is utilized to segregate soybean leaves from a complicated background, which ultimately results in a reduction in the background's potential to interfere with the analysis. The second module is used to figure out which disease is on a segmented image of a soybean leaf. Finally, the model's identification accuracy was 98.14%.

Hu et al.[15] introduced the CNN method for maize leaf disease recognition. The authors used data augmentation to improve the quality of the training set and TL to make the CNN model more accurate. Using a portion of the PlantVillage open dataset containing 4 different types of maize leaves such as grey leaf spot, northern leaf blight, common rust, and healthy leaves, the improved CNN produced an average accuracy of 97.6%, demonstrating its superior performance.

Agarwal et al. [16] created a CNN composed of 3-convolution layers, 2-fully connected layers, and 3-max-pooling layers. The authors employed the PlantVillage dataset that had maize leaves with 3- diseases: northern leaf blight, grey leaf spot, and common rust. Also in the dataset were images of healthy leaves, which showed that the architecture was 94% accurate. Waheed et al.[17] developed a densely-optimized convolutional neural network for the classification of 4-maize leaf classes available from the PlantVillage open dataset. The model was made up of 5-dense blocks and a SoftMax layer for classification. After being trained, the CNN was able to correctly sort the four different classes achieving the 98.06% of accuracy.

Aravind et al. [18] proposed pre-trained DL models like VGG16 and AlexNet based on the transfer learning concept. The features extracted are used to classify the healthy and disease-affected leaves in tomato plants with an accuracy of 97.49% and 97.23%, respectively. In the same way, Keke Zhang et al. [19] used leaf images to create CNN models based on transfer learning that can find and classify diseases in tomato crops. The pre-trained models like AlexNet, ResNet, and GoogLeNet proposed in this study utilize "Stochastic Gradient Descent" (SDG) and "Adaptive Moment Estimation (Adam)" optimizers. From the experimental results, ResNet with the SGD optimizer achieves an accuracy of 97.28%. Furthermore, the identification of disease in banana plants is done using the CNN-based LeNet model. The LeNet model produces better accuracy even if the images have different sizes, resolutions, and complex backgrounds [20].

Mohanty et al. [21] did the research to detect 26 varieties of plant diseases spread from 14 plants using the AlexNet and GoogLeNet models. The output is obtained from the model in two different categories, namely constructing the model from the beginning and applying the TL approach. The performance is better with the transfer learning approach, and it gives a classification accuracy of 99.35% and 99.27% for the GoogLeNet and AlexNet models, respectively.

Also, HalilDurmu et al. [22] suggested that AlexNet and SqueezeNet be used to find the disease in tomato plants. The PlantVillage database used in this research work consists of 10 varieties of tomato leaf images, including 9 diseased and 1 healthy class. The results reveals that AlexNet gives the highest recognition accuracy of 95.65%.

Likewise, Mohammed et al. [23] presented work related to recognizing tomato diseases using leaf images by means of AlexNet and GoogLeNet models. From the results, the GoogLeNet gives an accuracy of 99.18% in comparison with the Alexnet model, which has an accuracy value of 98.60%.

Liu et al. [24] developed a novel CNN model based on the AlexNet architecture. The proposed CNN model’s performance and the AlexNet model's performance are evaluated in the identification of four varieties of disease affected apple plants by means of leaf images. The performance evaluation results show that the novel CNN model gives better accuracy of 97.62% than AlexNet, which is at 91.19%.

According to the findings of the preceding study, early detection of plant disease can easily control the disease to some extent, which leads to an increase in production. The existing method stated in the above study utilized a variety of deep learning models to improve the accuracy of prediction of plant leaf diseases. Furthermore, a transfer learning method is integrated to improve the classification accuracy.

## III. Materials and Methodology

### A. Image Dataset

The potato dataset extracted from the PlantDoc [25] database is used in this work. The potato leaf diseases such as early blight and late blight consist of 1000 images each, which are utilised to assess the model performance. Fig. 1 shows the sample images of each class from the potato dataset. The complete dataset is split into two parts: training and testing, with 80% of the images used for training and 20% used for testing. Table.1 presents the details of training, validation, and testing images. Besides, to satisfy the dimensional requirements of the model, the images are resized to 224 x 224 pixels for VGGNet and ResNet. Similarly, the images are resized to 299 x 299 pixels for the Inception V3 model. The parameters, including learning rate, which was set to 0.01, and batch size, which was set at 32, and the number of epochs set to 25, are used in model training. Furthermore, Adaptive Moment Estimation (Adam) is used as the optimizer.

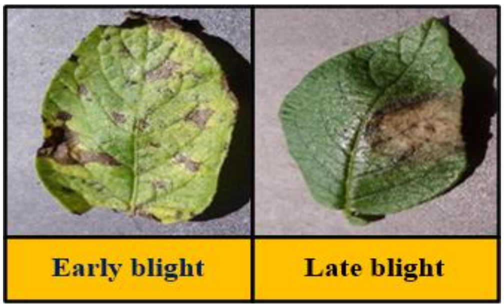



Fig. 1. Sample images of potato leaf diseases

TABLE I. DETAILS OF TRAINING, VALIDATION AND TESTING IMAGES USED

| Diseases | Training | Validation | Testing |
|---|---|---|---|
| Early Blight | 720 | 80 | 200 |
| Late Blight | 720 | 80 | 200 |

### B. *Deep Learning Models*

#### *1) ResNet*

The Residual Network (ResNet) model was introduced by He et al. in [26], which is a deep CNN that was trained effectively through the use of the Residual Unit. This set of residual units constitutes the network's structural components. The convolution and pooling layer components make up the residual units. The structure of ResNet makes it possible to speed up the process of training ultra-deep neural networks to an incredibly high degree, and the accuracy of the model has also been significantly enhanced. The ResNet50 network has 50 hidden layers and it is loaded with pre-trained weights of the ImageNet. In this work the dense layer is used as the output layer and it trained with the potato leaf disease dataset. The sigmoid function is also used as the activation function to find early blight and late blight on potato leaves.

#### *2) Inception V3*

Inception V3 is the refined version of the GoogLeNet model introduced by [27]. It consists of 42 layers with fewer parameters, and factorizing convolution reduces the number of parameters without affecting the model's efficiency. The inception module of this model has different-sized convolutions, such as 1x1, 3x3, and 5x5.The factorization is a significant feature of InceptionV3, which factorizes the 5x5 convolution layer into two 3x3 layers. The parameter reduction evades model overfitting and increases accuracy. Inception V3 is fine-tuned by truncating the existing fully connected layer and softmax layer. A new flattening layer is introduced, which converts the multidimensional input into a single-dimensional input. Finally, a new dense layer with sigmoid activation is used for the binary classification of early blight and late blight. In the fine-tuning process, the Inception V3 model utilizes the weights of ImageNet, and the weights of the new dense layer are updated through training with the potato dataset.

#### *3) VGGNet*

The Visual Geometry Group at Oxford University came up with the idea for the VGG network. This group took part in the 2014 ImageNet Large Scale Visual Recognition Challenge (ILSVRC), where they won first and second place in the tasks of classification and localization. VGG networks use the superposition of numerous 3x3 convolution filters to replace a large convolution filter, which increases network depth while decreasing total parameter count [28]. Therefore, the concept of 3x3 convolution filters has been the foundation for numerous classification models that followed. VGG16 and VGG19 are two popular VGG network topologies that differ in the number of convolutional layers they contain. In order to extract characteristics from the ImageNet image, convolutional layers are employed. VGG16 has a total of 13 convolution layers, as well as 3 fully connected layers, 5 pooling layers, and a softmax layer. Each convolution layer has a multiple filter that is 3 x 3. VGG19, on the other hand, has 16 convolution layers, 3 fully linked layers, 5 pooling layers, and 1 softmax layer. During the fine-tuning process of the VGG models, the original softmax layer is truncated and a new dense layer with a sigmoid activation function is added for the classification of potato leaf diseases. The new layer is trained with potato leaf images and also a number of class matches with the potato dataset.

## IV. EVALUATION METRICS

The performance of the models in the recognition of potato leaf diseases is measured using metrics such as accuracy, precision, recall, and F1 score. The following is a list of the formulas for the various measuring indicators:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

$$\Pr ecision = \frac{TP}{TP + FP} \quad (2)$$

$$\mathrm{Re}\, call = \frac{TP}{TP + FN} \quad (3)$$

$$\mathrm{Re}\, call = 2x \frac{\Pr ccision \mathrm{Re}\, call}{\Pr ecision + \mathrm{Re}\, call} \quad (4)$$

where, true positive (TP) indicates the number of positive values for which the label was correctly identified as positive; false positive (FP) represents the number of negative values for which the label was incorrectly identified as positive; true negative (TN) denotes the number of negative values for which the label was correctly identified as negative; and false negative (FN) denotes the number of positive values for which the label is wrongly identified as negative.

## V. EXPERIMENTAL RESULTS

In this work, the performance of Inception V3, ResNet50, VGG16, and VGG19 models that have already been trained to classify leaf diseases like early blight and late blight is compared. Initially, the experiment runs for a maximum of 25 epochs. The training and validation accuracy obtained for each model through training is depicted in Fig.2 (a). Similarly, the loss variations for all the models obtained in training and validation are also presented in Fig.2 (b). The results show that training and validation loss are initially high for the model and are gradually reduced after increasing the number of epochs. From Fig.2, it can be seen that VGG19 achieved the highest classification accuracy of 99%, whereas ResNet50 obtained the lowest performance with an accuracy of 90.50% Furthermore, each model's prediction accuracy is determined using metrics such as "precision, recall, and F1 score". The values of the different evaluation metrics are listed in the Table.2.The results of Table.1 illustrate that VGG19 provides the best performance compared to the other models.

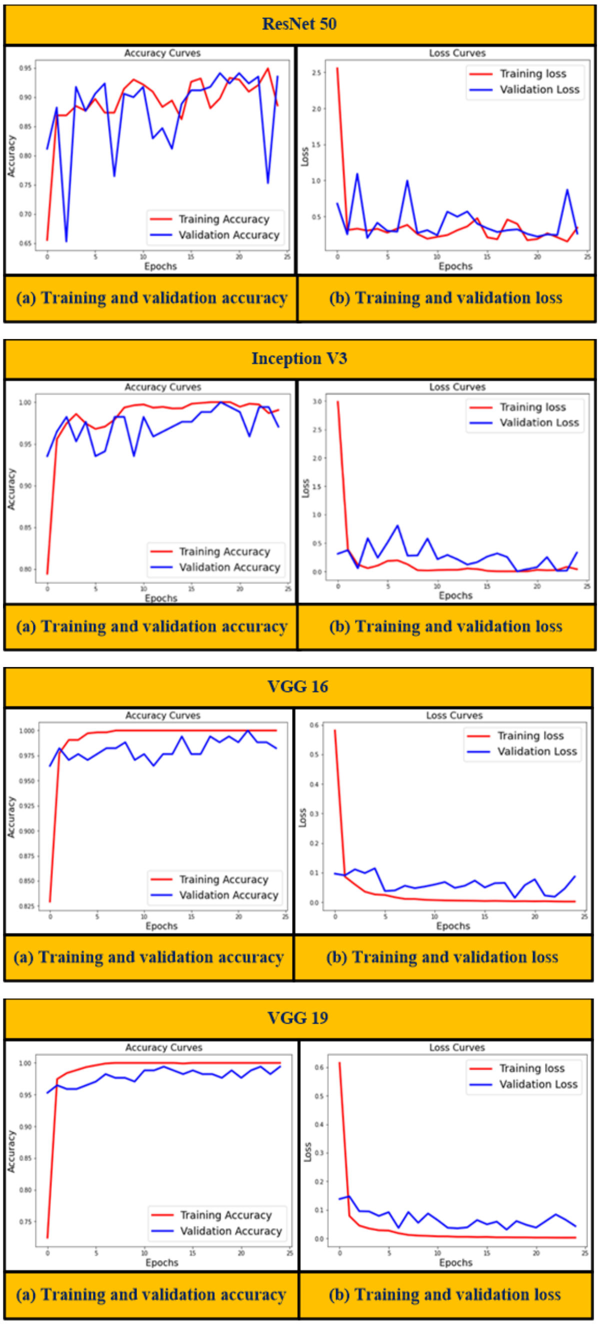



Fig. 2. The proposed DL models performance

TABLE II. PERFORMANCE COMPARISON VARIOUS CNN MODELS

| Models | Accuracy (%) | Precision | Recall | F1 score |
|---|---|---|---|---|
| Inception V3 | 98 | 0.98 | 0.98 | 0.98 |
| ResNet50 | 90.50 | 0.91 | 0.91 | 0.90 |
| VGG16 | 98.50 | 0.99 | 0.98 | 0.98 |
| VGG19 | 99 | 0.99 | 0.99 | 0.99 |

## VI. CONCLUSION

In recent years, the agricultural industry has experienced many changes. Furthermore, the identification of plant diseases is essential to increase the crop yield. In this work, the most recent DL models, including Inception V3, ResNet50, VGG16, and VGG19, are used to identify the early blight and late blight leaf diseases of potato. Moreover, the effectiveness of transfer learning for the identification of diseases with a limited amount of data is proposed in this work. The results of the experiment conducted shows that the VGG19 model with the transfer learning approach achieved the highest recognition accuracy of 99% compared with the other models. The performance comparison of all the models reveals that VGG19 applying fine-tuning shows the best performance. The performance of the model becomes evident that transfer learning based deep learning models are highly appropriate for automatic identification of leaf diseases.

## REFERENCES


[1] H. Park, J. S. Eun and S. H. Kim, Image-based disease diagnosing and predicting of the crops through the deep learning mechanism, In Information and Communication Technology Convergence (ICTC), IEEE 2017 International Conference on, pp. 129-131, 2017.

[2] [2] List of countries by potato production https://en.wikipedia.org/wiki/List_of_countries_by_potato_production.

[3] M. Islam ,AnhDinh , K. Wahid , P. Bhowmik, Detection of potato diseases using image segmentation and multiclass support vector machine, IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE),p.no:1-4,2017.

[4] UNEP (2013) Smallholders, food security, and the environment.

[5] Thangaraj, R., Anandamurugan, S., & Kaliappan, V. K. (2021). Automated tomato leaf disease classification using transfer learning-based deep convolution neural network. *Journal of Plant Diseases and Protection*, *128*(1), 73-86.

[6] Y.LeCun,Y.Bengio and G. Hinton, Deep learning, Nature, 521(7553) (2015) 436.

[7] K.Noda,Y.Yamaguchi, K.Nakadai, H.G.Okuno andT.Ogata,Audio-visual speech recognition using deep learning,ApplIntell. 42(4) (2015) 722-737.

[8] G.Wu,W.Lu, G.Gao,C. Zhao andJ.Liu,Regional deep learning model for visual tracking, Neurocomputing, 175 (2016) 310-323.

[9] S.J.Pan and Q.Yang, A survey on transfer learning. IEEE Trans. Knowl. Data Eng.22 (10) (2010) 1345-1359.

[10] T. Akram, B. Laurent, S.R. Naqvi, M. MbomAlex and N. Muhammad, A deep heterogeneous feature fusion approach for automatic land-use classification, Inf. Sci. (Ny) 467 (2018) 199–218.

[11] A.I. Shahin,Y.Guo, K.M. Amin and A.A. Sharawi, White Blood Cells IdentificationSystem Based on Convolutional Deep Neural Learning Networks, Comput. Methods Programs Biomed.168 (2017) 69-80.

[12] A.Kamilaris and F.X.Prenafeta-Boldu,2018. Deep learning in agriculture: a survey.Comput. Electron. Agric. 147 (2018) 70–90.

[13] Subetha, T., Khilar, R., Subaja Christo, M., 2021. A comparative analysis on plant pathology classification using deep learning architecture - Resnet and VGG19. In: Materials Today: Proceedings. https://doi.org/10.1016/j.matpr.2020.11.993.

[14] Karlekar, A., Seal, A., 2020. SoyNet: Soybean leaf diseases classification. Comput. Electron. Agric. 172 https://doi.org/10.1016/j.compag.2020.105342.

[15] R. Hu, S. Zhang, P. Wang, G. Xu, D. Wang, and Y. Qian, ''The identification of corn leaf diseases based on transfer learning and data augmentation,'' in *Proc. 3rd Int. Conf. Comput. Sci. Softw. Eng.*, May 2020,pp. 58–65.

[16] M. Agarwal, V. K. Bohat, M. D. Ansari, A. Sinha, S. K. Gupta, and D. Garg, ''A convolution neural network based approach to detect the disease in corn crop,'' in *Proc. IEEE 9th Int. Conf. Adv. Comput. (IACC)*, Dec. 2019,pp. 176–181.

[17] A. Waheed, M. Goyal, D. Gupta, A. Khanna, A. E. Hassanien, and H. M. Pandey, ''An optimized dense convolutional neural network model for disease recognition and classification in corn leaf,'' *Comput. Electron.Agricult.*, vol. 175, Aug. 2020, Art. no. 105456.

[18] AravindKrishnaswamyRangarajan, Raja Purushothaman, Anirudh Ramesh, Tomato crop disease classification using pre-trained deep learning algorithm, International Conference on Robotics and Smart Manufacturing (RoSMa2018), Procedia Computer Science 133 (2018) 1040–1047.

[19] K.Zhang,Q.Wu,A. Liu, and X.Meng,Can Deep Learning Identify Tomato Leaf Disease?, Adv.Multimedia. 2018.

[20] J.Amara, B.Bouaziz, A.Algergawy, A deep learning-based approach for banana leaf diseases classification, in: Lecture notes in informatics (LNI) (2017) pp.79e88

[21] S.P.Mohanty, D.P.Hughes, M.Salathé, Using deep learning for image-based plant disease detection, Front. Plant Sci. 7 (2016) 1419.

[22] H.Durmuş,E.O.Güneş, and M.Kırcı, Disease detection on the leaves of the tomato plants by using deep learning. In 2017 6th International Conference on Agro-Geoinformatics (2017) (pp. 1-5). IEEE.

[23] M.Brahimi, K.Boukhalfa, A.Moussaoui, Deep learning for tomato diseases: classification and symptoms visualization, ApplArtifIntell. 31(4) (2017) 299-315.

[24] A.Ramcharan,K.Baranowski,P.McCloskey,B.Ahmed,J. Legg and D.P. Hughes, Deep learning for image-based cassava disease detection. Fron.PlantSci. 8(2017) 1852.

[25] Singh, D., Jain, N., Jain, P., Kayal, P., Kumawat, S., & Batra, N. (2020). PlantDoc: a dataset for visual plant disease detection. In Proceedings of the 7th ACM IKDD CoDS and 25th COMAD (pp. 249-253).

[26] He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 2016-Decem, 770–778. Doi: 10.1109/CVPR.2016.90.

[27] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2818-2826).

[28] Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv 2014, arXiv:1409.1556.

Gmail

**Balasubramaniam V <balasubramaniamvadivel@gmail.com>**

# [ICDATE 2023] #1570906699 has been uploaded

1 message

**ICDATE 2023** <icdate2023-chairs@edas.info> Sat, Apr 22, 2023 at 8:42 PM
To: Rajasekaran Thangaraj <t.rajasekaran@kpriet.ac.in>, Balasubramaniam Vadivel <balasubramaniamvadivel@gmail.com>, Chandhru L <chandhru0808@gmail.com>, Harish K <harishkvel73@gmail.com>, Aravind T <aravindthangaraj7@gmail.com>, Sivaraman Eswaran <sivaraman.eswaran@gmail.com>

Dear Dr. Sivaraman Eswaran:

Thank you for uploading your paper 1570906699 (*Tea Leaf Disease Identification Using Improved Convolution Neural Network*) to **ICDATE 2023 (International Conference on Digital Applications, Transformation & Economy 2023)**. The paper is of type application/pdf and has a length of 591555 bytes.

You can modify your paper at 1570906699 and see all your submissions at https://edas.info/index.php?c=30358 using the EDAS identifier sivaraman.eswaran@gmail.com

Regards,
Saaveethya Sivakumar, General Chair

# Tea Leaf Disease Identification using Improved Convolution Neural Network


Rajasekaran Thangaraj
Department of Computer Science and Engineering
KPR Institute of Engineering and Technology
Coimbatore, India
rajasekaran30@gmail.com

Balasubramaniam Vadivel
Department of Computer Science and Engineering
KPR Institute of Engineering and Technology
Coimbatore, India
balasubramaniamvadivel@gmail.com

S.Raguvaran
Department of Computer Science and Engineering
KPR Institute of Engineering and Technology
Coimbatore, India
prof.sraguvaran@gmail.com

Chandhru L
Department of Computer Science and Engineering
KPR Institute of Engineering and Technology
Coimbatore, India
chandhru0808@gmail.com

Harish K
Department of Computer Science and Engineering
KPR Institute of Engineering and Technology
Coimbatore, India
harishkvel73@gmail.com

Aravind T
Department of Computer Science and Engineering
KPR Institute of Engineering and Technology
Coimbatore, India
aravindthangaraj7@gmail.com



***Abstract*—Plant diseases are a significant concern for the agricultural industry, as they can reduce crop yields and cause economic losses. Tea is a popular and widely consumed beverage in India, and the tea crop can be affected by different leaf diseases. Early detection of the diseases is essential to prevent the spread of other crops and minimize production losses. Traditional methods of detecting leaf diseases involve manually interpreting the images, which can be time-consuming and laborious. To address this, artificial intelligence techniques, specifically deep learning models are used for more accurate and efficient detection of tea leaf disease. This study compares the performance of several deep learning models including ResNet50, Resnet50-RS, ResNet101-V2, and Modified-ResNet50V2 (M-RN50$_{V2}$), and finds that the M-RN50$_{V2}$ model has the highest accuracy at 90.84%. Overall, this work aims to promote sustainable agriculture by leveraging cutting-edge technology for disease classification**.



***Keywords— Classification of Tea Leaf Disease, Deep Learning (DL) Model, Classification of Tea Leaves, Transfer Learning (TL), Modified-ResNet50-V2 (M-RN50$_{V2}$)***


## I. INTRODUCTION

The cultivation of tea has seen a rapid increase in many countries, owing to its health benefits, and it has become a crucial cash crop. However, when tea plants become diseased, the leaves drop prematurely, and the fresh shoots wither, leading to reduced plant vigor and lower yields[1]. Severe diseases can even cause young tea trees to die, resulting in significant losses for growers. To ensure high-quality tea production and profitability, it is crucial to accurately identify diseased tea leaves. Currently, identifying the disease requires plant protection specialists to visit the affected areas and use manual methods, which are time-consuming and expensive, particularly in rocky and mountainous terrain[2]. As computer technology has advanced, machine learning and image processing techniques have become more widely employed for plant disease detection and identification. Traditional agricultural disease identification methods are subjective and have long cycles[3]. Moreover, manual feature extraction of diseased leaves is essential for traditional machine-learning approaches, and the accuracy of manual features can have a significant impact on identification and detection performance[4-8]. This study aims to accurately detect and assess the severity of tea leaf disease using deep learning techniques, which can help in selecting and applying pesticides efficiently, managing disease, and reducing soil and tea pollution.

The remaining part of the article is structured as follows. Part II reviews the related research on identifying tea leaf diseases. Section III provides information on the tea leaf dataset. Sections IV and V cover the various DL models used to identify diseases and the metrics employed to evaluate the effectiveness of Deep Learning, respectively. The experiment's results are discussed in Section VI, and the study is concluded in Section VII.

## II. RELATED WORKS

Jing Chen et al [9]. Developed LeafNet, a deep convolutional neural network (CNN), with the objective of identifying diseases in tea plants from leaf images. The model constructs a bag of visual words model by employing DSIFT features and feature extractor filters of various sizes. The effectiveness of LeafNet in disease classification was compared to classifiers such as Support Vector Machines (SVM) and Multi-Layer Perceptron (MLP). The findings revealed that LeafNet performed better than both SVM (60.62%) and MLP (70.77%) methods, with an average classification accuracy of 90.16%. These results suggest that by accurately identifying tea leaf diseases, LeafNet can enhance disease monitoring in tea plants.

Hu Gensheng et al [10]. proposed a method for diagnosing tea leaf illnesses using an enhanced deep CNN with a multiresolution feature extraction module and depthwise separable convolution. Their method outperformed both common machine learning methods and deep learning models, demonstrating an average identification accuracy of 92.5%. In comparison to the VGG16 and AlexNet deep learning network models, the enhanced model also featured fewer parameters and shorter convergence iteration times. According to the experts, this technique can quickly and accurately identify illnesses that harm tea leaves.

S. Gayathri et al[11]. stated that advances in deep learning are responsible for the increased interest in computerized image classification and plant disease diagnostics using image recognition technology. India, which is the world's largest user and second-largest manufacturer of tea, may face growth challenges in its tea plants due to diseases that affect the plant. In this project, they developed LeNet, a deep CNN, with the goal of utilizing leaf image sets to detect illnesses in tea plants. In the future, LeNet may enhance the accuracy of tea leaves and other crop leaves analysis.

Bao et al. [12] aimed to improve the accuracy of diagnosing tea leaf infections in order to prevent and manage disease in tea plantations. The authors recognized that tea leaf photos are challenging to analyze accurately due to their complex backgrounds, dense leaves, and significant environmental fluctuations. Convolutional neural networks (CNNs) are commonly used for detecting diseases in plants, but they often struggle with identifying diseases in tea leaves. To address this issue, Bao et al. proposed updated versions of RetinaNet and AX-RetinaNet networks that contained an improved multiscale feature fusion module, X-module, and channel attention module. These enhancements allowed users to choose more useful parameters while minimizing interference from redundant ones, resulting in a mean average precision (mAP) value of 93.83% and an F1-score level of 0.954. Testing findings showed that AX-RetinaNet surpassed existing target detection and identification networks for tea leaf illnesses in natural scene photos, with an improvement in identification accuracy of about 1.5% compared to the initial network.

Dikdik Krisnandi et al. [13] proposed the utilization of a concatenated Convolutional Neural Network(CNN) for automatic tea leaf disease detection. The system includes GoogleNet, Xception, and Inception-ResNet-v2 CNNs and achieves a high accuracy of 89.64%. The training dataset consists of 4727 tea leaf images, including a healthy class and three classes that represent common diseases in Indonesia. Since expert visual inspection for disease identification can be costly, the proposed deep learning approach not only reduces identification costs but also offers accurate disease detection

## III DATASET DESCRIPTION FOR REAL TIME DATA

The dataset of tea crop used in this study is collected from a private farm in the kotagiri district.The dataset consist of binary classes such as healthy and diseased leaves.Totally, 250 images are presented in each class. The data augmentation techniques, including cropping, saturation image correction, exposure adjustment, clockwise and anticlockwise rotation, and horizontal and vertical flipping, to increase the dataset size. In a 70:20:10 ratio, the authors divided the dataset into training, validation, and testing sets, using 70% of the images for training, 20% for validation, and the remaining 10% for testing. During the training process, a learning rate of 0.01, batch size of 32, and 50 epochs, and the Adaptive Moment Estimation (Adam) optimizer is used. The sample real-time tea leaf images and their augmented versions are shown in Fig. 1 and Fig. 2, respectively, while the number of images utilized for each set is presented in Table 1

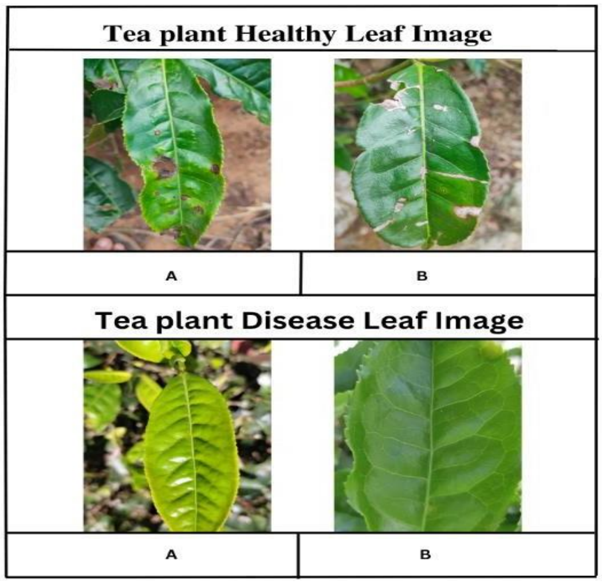



Fig.1 Sample images of tea leaves collected from kotagiri tea farm

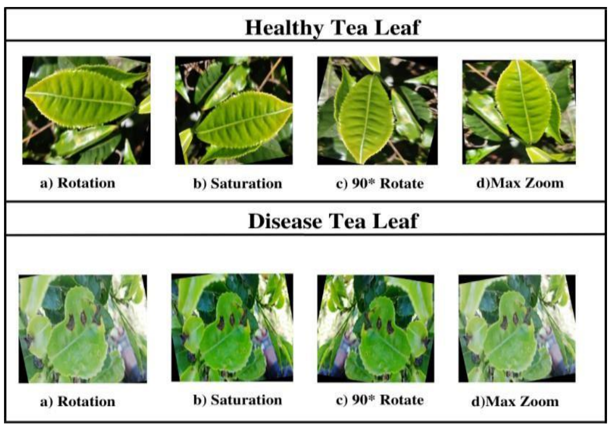



Fig.2 Augmented Images of Tea Leaf

Table.I Dataset Summary

| Classes | Training | Validation | Testing |
|---|---|---|---|
| **Healthy** | 1,680 | 61 | 32 |
| **Disease** | 1,256 | 47 | 21 |

IV TRANSFER LEARNING TECHNIQUE

Transfer Learning (TL) is a method for tackling similar problems, especially those with little available data, by using the information gained from addressing one problem. The two main approaches in TL are fine-tuning and feature extraction. While fine-tuning entails retraining the model using a fresh dataset and modifying its weights, feature extraction entails leveraging features derived from a pre-trained model to train new picture datasets. The number of classes in the hidden layers is modified during fine-tuning to match the target dataset. The purpose of this research is to apply TL to the ResNet50, ResNet101-V2, ResNet50-RS, and M-RN50$_{V2}$ DL models to improve the accuracy of tea leaf disease identification. Figure 2 from the research illustrates how the TL system can improve leaf disease recognition precision.

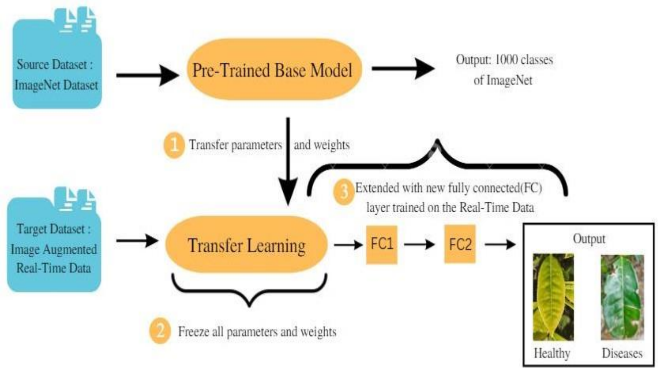



Fig.3 Proposed approach for Tea Leaf Disease Identification

*A. ResNet50*

The ResNet50 model[14] is a deep Convolutional Neural Network (CNN) architecture developed by He et al. in 2015 that employs the Residual Unit for training. Residual Units are utilized as structural components in the network, including convolution and pooling layer components. The ResNet architecture enables ultra-deep neural networks to be trained quickly while also improving model accuracy. The ResNet50 model, which is pre-trained with weights from ImageNet, contains 50 hidden layers. In this study, the dense layer is trained using the Tea leaf disease dataset and used as the output layer. Additionally, the activation function utilized for identifying healthy and diseased tea leaves is the sigmoid function.

*B. Resnet50-RS*

The ResNet50 architecture has a module called ResNet50-RS[15]. Its goal is to lessen the overfitting issue with deep neural networks and increase the model's adaptability to fresh data. Residual units, which are composed of convolution and pooling layers, make up the ResNet50-RS network architecture. When compared to conventional CNNs, deep neural networks can be trained more quickly thanks to these residual units. The ImageNet dataset served as the basis for pre-training the ResNet50RS network, and in this work, it is fine-tuned using a new dataset for a particular application. The output layer is a dense layer, and the activation function is a sigmoid function. This fine-tuning is done to obtain the desired performance on the objective task.

*C. ResNet101-V2*

The ResNetV2 architecture has been changed to become ResNetV2-101. With 101 hidden layers and a deeper network design, it is better able to learn intricate representations of the input data. ResNetV2-101, which consists of convolution and pooling layers, uses residual units as its structural building blocks, just like ResNetV2. In comparison to conventional CNNs, the accuracy of ultra-deep neural networks can be increased by faster training thanks to ResNetV2-residual 101's units. In this work, the ResNetV2-101 network is fine-tuned on a new dataset for a particular purpose after being pre-trained on the ImageNet dataset. The output layer is a dense layer, and the activation function is a sigmoid function.

*D. Modified-ResNet50-V2(M-RN50$_{V2}$)*

ResNetV2 is a modified version of the Residual Network (ResNet) Model that utilizes residual units as its structural components. These units consist of convolution and pooling layers, and allow for faster training of ultra-deep neural networks, leading to improved accuracy compared to traditional Convolutional Neural Networks (CNNs). The ResNetV2 network has a hidden layer architecture that has been further improved from the ResNet model and is pre-trained on the ImageNet dataset, like the ResNet50. To enhance the model's performance, additional layers, including dropout layer, batch normalization, activation function Relu, and dense layer, are added to the ResNet50-V2 model architecture. These modifications help to train the fully connected layer of the model, which is used to classify healthy and diseased tea leaf images.

V PERFORMANCE METRICS

Performance metrics are used to evaluate the effectiveness and efficiency of a system or process. They provide quantitative measurements that can be used to track progress, identify areas for improvement, and make data-driven decisions.

**Accuracy:** The proportion of correctly classified tea leaf image among the datasets. This metric is often used to assess the overall performance of a classifier.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

**Precision:** The proportion of tea leaves identified as diseased that are actually diseased, out of all tea leaves identified as diseased. This metric is a measure of the classifier's ability to avoid false positives.

$$\Pr ecision = \frac{TP}{TP + FP}$$

**Recall:** The proportion of actual diseased tea leaves that are correctly identified as diseased, out of all actual diseased tea leaves in the dataset. This metric is a measure of the classifier's ability to avoid False Negatives.

$$\Pr ecision = \frac{TP}{TP + FP}$$

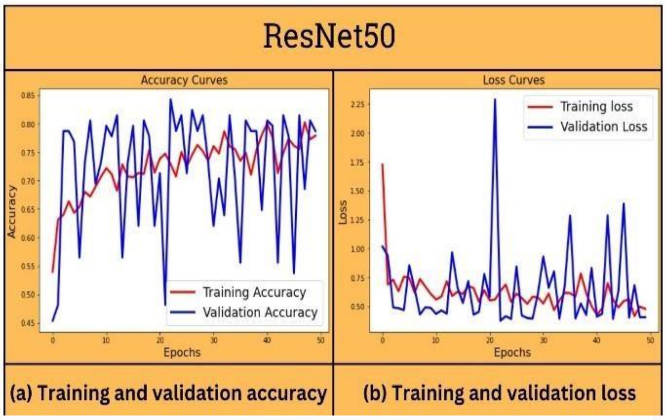
**F1-score:** A weighted average of precision and recall, which provides a single measure of the classifier's overall performance. This metric is useful when both precision and recall are important.

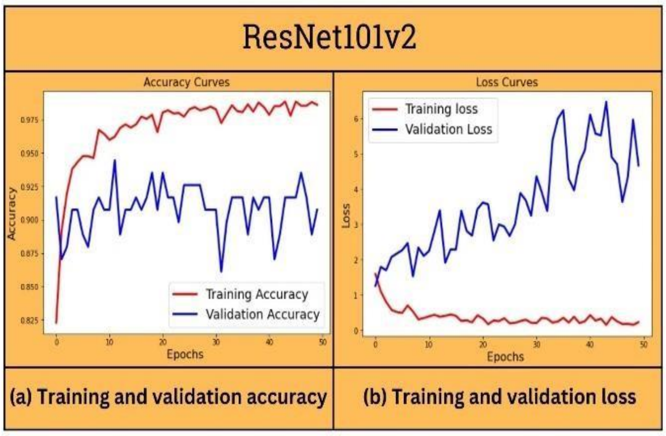
F1 score = 2 * (True Positives) / (2 * True Positives + False Positives + False Negatives)

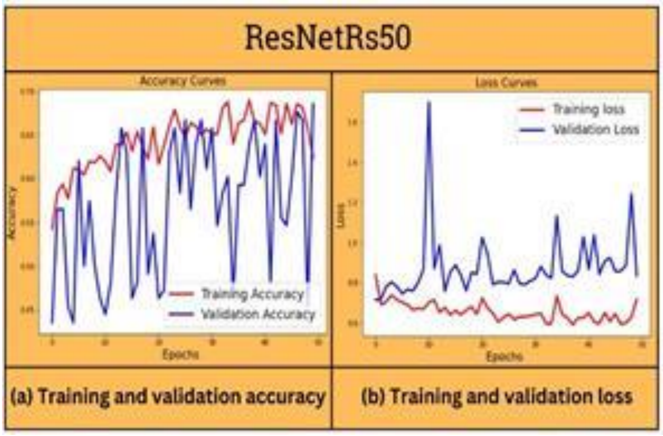
By comparing the predicted labels of the model to the actual labels of the test data, it is possible to determine how many true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) were generated by the model. True positives (TP) refer to the number of instances where the model predicted a positive class correctly, given that the true class was positive. In other words, the model correctly identified a positive instance as positive. False negatives (FN) refer to the number of instances where the model predicted a negative class incorrectly, given that the true class was positive. In other words, the model incorrectly identified a positive instance as negative.

VI EXPERIMENTAL RESULT

In this study, the performance of models such as ResNet50, ResNet50-RS, ResNet101-V2, and M-RN50V2 trained to differentiate between healthy and unhealthy tea leaves is compared. The study initially runs for a maximum range of fifty epochs. Figure (2-5) displays each model's post-training training and validation accuracy (a). Similar, Figure (2-4) displays the changes in loss for all models obtained during validation and training (b). The results show that the model's initial training and validation loss starts off high and slowly reduces as the number of epochs rises. While M-RN50V2 had the best classification accuracy (90.84%), ResNet50 had the lowest performance (75.47% accuracy). . The prediction accuracy of each model is also examined using metrics including Recall,Precisionand F1 score. Values for the several evaluation indicators are compiled in Table.2. The results in Table.2 demonstrate that the M-RN50V2 performs more effectively in comparison to the other models.



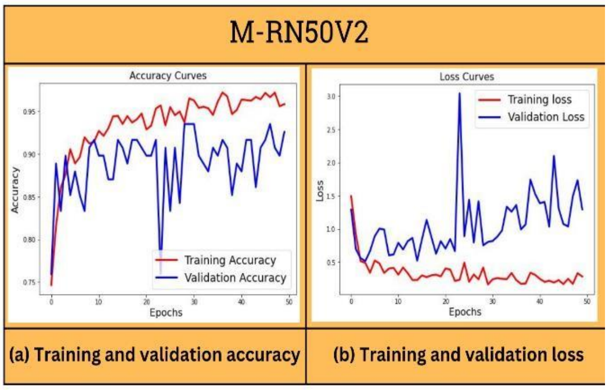



Fig.4 Training and Validation loss of different DL models a) ResNet50 b)ResNet50-RS c)ResNet101-V2 d) M-RN50$_{V2}$

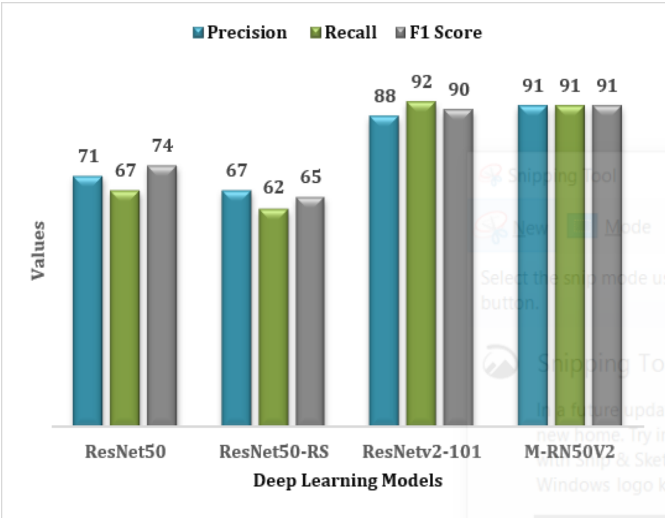



Fig.5 Comparison of Evaluation Metrics for different DL models

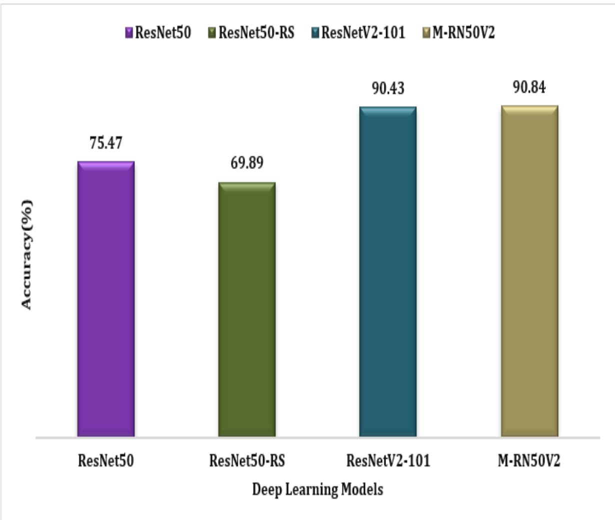



Fig.6 Accuracy comparison of DL models

## VII. CONCLUSION

The identification of plant diseases is crucial for the agricultural industry to improve crop productivity. This study aims to detect healthy and diseased tea leaves using various deep learning models, such as ResNet50, ResNet101-V2, ResNet50-RS, and M-RN50$_{V2}$. The study recommends the use of transfer learning in identifying diseases, especially in small or insufficient datasets. The results of the experiment suggest that, while compared to other models, With a classification results of 90.84%, the M-RN50V2 model with Proposed Transfer Learning Model outperformed the competition. The performance comparison indicates that fine-tuning is the best method for achieving the highest accuracy. Based on these findings, deep learning models that incorporate transfer learning are recommended for automatic leaf disease detection.

## REFERENCE

[1] Fang J, Sureda A, Silva AS, Khan F, Xu S, Nabavi SM. Trends of tea in cardiovascular health and disease: A critical review. Trend Food Sci Tech 2019;88:385–96.

[2] Qi LC, Zhang YD. Major tea plant diseases and their control. J Tea 2016;42(1):10–2.

[3] Sun YY, Jiang ZH, Zhang LP, Dong W, Rao Y. SLIC_SVM based leaf diseases saliency map extraction of tea plant. Comput Electron Agricult 2019;157:102–9.

[4] Pantazi XE, Moshou D, Tamouridou AA. Automated leaf disease detection in different crop species through image features analysis and One Class Classifiers. Comput Electron Agricult 2019;156:96–104.

[5] Hossain MS, Mou RM, Hasan MM, Chakraborty S, Razzak MA. Recognition and detection of tea leaf's diseases using support vector machine. In: 2018 IEEE 14th International Colloquium on Signal Processing & Its Applications; 2018. p. 150–4.

[6] Zhang SW, Wang HX, Huang WZ, You ZH. Plant diseased leaf segmentation and recognition by fusion of superpixel, K-means and PHOG. Optik 2018;157: 866–72.

[7] Dhingra G, Kumar V, Joshi HD. Basil leaves disease classification and identification by incorporating survival of fittest approach. Chemometr Intel Lab Syst 2019;186:1–11.

[8] Thangaraj, R., Anandamurugan, S., & Kaliappan, V. K. (2021). Automated tomato leaf disease classification using transfer learning-based deep convolution neural network. *Journal of Plant Diseases and Protection*, *128*(1), 73-86.

[9] Chen, Jing, Qi Liu, and Lingwang Gao. "Visual tea leaf disease recognition using a convolutional neural network model." *Symmetry* 11, no. 3 (2019): 343.

[10] Hu, Gensheng, Xiaowei Yang, Yan Zhang, and Mingzhu Wan. "Identification of tea leaf diseases by using an improved

deep convolutional neural network." *Sustainable Computing: Informatics and Systems* 24 (2019): 100353.
[11] Gayathri, S., DC Joy Winnie Wise, P. Baby Shamini, and N. Muthukumaran. "Image analysis and detection of tea leaf disease using deep learning." In *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*, pp. 398-403. IEEE, 2020.
[12] Bao, Wenxia, Tao Fan, Gensheng Hu, Dong Liang, and Haidong Li. "Detection and identification of tea leaf diseases based on AX-RetinaNet." *Scientific Reports* 12, no. 1 (2022): 2183.
[13] Krisnandi, Dikdik, Hilman F. Pardede, R. Sandra Yuwana, Vicky Zilvan, Ana Heryana, Fani Fauziah, and Vitria Puspitasari Rahadi. "Diseases classification for tea plant using concatenated convolution neural network." *CommIT (Communication and Information Technology) Journal* 13, no. 2 (2019): 67-77.
[14] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778. 2016.
[15] ResNet-RS Explained | Papers With Code