PLANT DISEASE DETECTION USING DEEP LEARNING-BASED APPROACH

Thesis

Submitted in partial fulfilment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

by

Sunil C K

Register No. 187129IT005

DEPARTMENT OF INFORMATION TECHNOLOGY

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA
SURATHKAL, MANGALORE - 575025

August, 2023

Declaration

I hereby *declare* that the Research Thesis entitled "Plant Disease Detection Using Deep Learning-based Approach" which is being submitted to the National Institute of Technology Karnataka, Surathkal, in partial fulfillment of the requirements for the award of the Degree of Doctor of Philosophy in Information Technology is a *bonafide report of the research work carried out by me*. The material contained in this thesis has not been submitted to any University or Institution for the award of any degree.

NITK Surathkal
Date

Sunil C K
Register No: 187129IT005
Department of Information Technology

Certificate

This is to *certify* that the Research Thesis entitled "Plant Disease Detection Using Deep Learning-based Approach" submitted by Sunil C K (Register Number: 187129IT005) as the record of the research work carried out by him, is *accepted as the Research Thesis submission* in partial fulfillment of the requirements for the award of the degree of Doctor of Philosophy.

28/08/2023

Dr. Jaidhar CD
Research Guide,
Associate Professor
Department of Information Technology
NITK Surathkal - 575025

25/08/23

Dr. Nagamma Patil
Research Co-Guide
Assistant Professor
Department of Information Technology
NITK Surathkal - 575025

28/08/2023

Chairman - DRPC
(Signature with Date and Seal)

CHAIRMAN - DRPC
Department of Information Technology
NITK Surathkal, Srinivasnagar PO.
Mangaluru 575 025, INDIA

This thesis is dedicated to
My Grand Mother and My Parents

Acknowledgements

First and foremost, I would like to express my sincere and profound gratitude to my supervisor **Dr. Jaidhar C D**, co-supervisor **Dr. Nagamma Patil**, Head of the Department, Information Technology department, and Research Progress Assessment Committee members **Dr. Bhawana Rudra**, and **Dr. Prabu K** for their continuous support and encouragement.

I thank all the teaching and non-teaching staff of the Department of Information Technology and my fellow doctoral students for their cooperation.

It is always impossible without the family support to invest huge time for research. I owe this thesis to my grand mother **Late Gowramma** and father-in-law **Mr. Sidde Gowda C M**. I am debted to my parents, **Mrs. Ponnamma C K** and **Mr. Komari Gowda C S**, for the whole life. I also thank my sisters **Mrs. Vinutha C K** and **Mrs. Rajeshwari C K**. I also thank to my in-laws. Life would be incomplete without my daughter **Urvika S** for spending lovely time. Finally, I should mention my source of inspiration during my research work, my wife, **Mrs. Sowjanya C R**, who consistently gave mental support all through the tough journey. Infinite thanks to her for keeping my life green and lovable.

(Sunil C K)

Abstract

Food security is threatening due to the exponentially growing global population. There are many reasons for food scarcity, such as exponential population, environmental disasters, climate change, the impact of COVID-19, and wars. Agriculture's productivity has decreased in the last decade due to climate change and inappropriate usage of water, fertilizer, and pesticides, which stimulate plant diseases. Plant diseases and pests are also the cause of reducing the production of food all over the globe. Plant diseases cause around 20% to 40% loss in the production of agricultural products. Plant diseases extensively impact agrarian production growth. It results in a price hike on food grains and vegetables. Early detection of plant disease is essential to reduce economic loss and predict yield loss. Early perception of pathogens and insinuating proper medications are crucial to enhance crop yield and quality. Current plant disease detection involves the physical presence of domain experts to ascertain the disease. As a result, timely plant disease recognition entails sustained crop supervision from the start. Some research works have contemporarily been proposed as curative control measures. However, such an approach requires expensive equipment that is out of reach for small or middle-scale yeoman.

Deep learning-based approaches vary in network architecture, and learning of the features by each model varies from one another in some aspects. To take this as an advantage, this study proposed an ensemble-based deep learning approach using AlexNet, ResNet, and VGGNet. Seven different plant disease dataset is used with the binary and multiclass dataset. This ensemble-based approach enhances the classification result by minimizing the miss-classification effect. It constructively perceives plant diseases by analyzing plant leaf images. A broad set of experiments were conducted using different plant leaf image datasets such as Cardamom, Cherry, Grape, Maize, Pepper, Potato, and Strawberry to assess the agility of the proposed approach. Experiential results show that the proposed method attained a maximal detection accuracy of 100% for binary and 99.53% for multiclass datasets.

Deep learning-based plant disease detection is proposed in this work by addressing some of the challenges. Precise plant disease detection is essential, where more than one disease has similar symptoms and nature, and also to achieve excellent performance in spite of the imbalanced data. This study proposed a Multilevel Feature Fusion Network (MFFN), which combines the features learned at different levels of the network and also uses the adaptive attention technique by employing channel and pixel attention mechanism, which fabricates the network more robust by considering the deeper network features which are shown in different channels and also with the pixel level features, with this the network is able to classify the diseases precisely on tomato plant dataset. The proposed deep learning-based approach is trained and tested on a tomato plant leaves dataset and achieved 99.36% training accuracy, 99.88% validation accuracy, and 99.5% external testing accuracy. It outperformed the existing approaches relevant to the tomato plant dataset. Further, this work also proposes a pesticide prescription module that provides pesticide information based on the type of tomato leaf disease.

Plant disease detection using a complex background and images captured in different conditions is one of the challenges; this study proposed a cardamom plant disease detection approach by collecting the images in a complex background using different electronic gadgets. This study proposed a hybrid deep learning-based approach consisting of two stages: the background removal stage and the classification stage. U^2-Net is used for the background removal task, and EfficientNetV2 is used for the classification task. This makes the network more robust to handle the plant leaf images captured in complex nature.A large number of experiments were conducted to evaluate the proposed approach's performance and compare it to other models such as EfficientNet and Convolutional Neural Network (CNN). According to the experimental results, the proposed approach achieved a detection accuracy of 98.26%.

The approaches proposed in this study are producing prominent results. This study also suggested a pesticide prescription module for tomato plant leaf diseases. The proposed solutions in this study contribute to the field of plant disease detection, which can be adopted for real-time plant disease application. The overall aim of this study is to provide an efficient and robust plant disease detection approach.

Keywords: Attention Mechanism; Climate Change; Deep Learning; Plant Disease; Pesticide Prescription.

Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Meaning
ABC	Artifcial Bee Colony
ACSPAM	Adaptive Channel Spatial and Pixel Attention Mechanism
AFM	Adaptive Feature Maps
CNN	Convolutional Neural Network
DCNN	Deep Convolutional Neural Networks
DLM	Deep Learning Model
DSC	Depthwise Separable Convolution
FC	Fully Connected
FM	Feature Maps
IoT	Internet of Things
MCNN	Multilayer CNN
MCLCNN	Multi Convolutional Layer-based Convolutional Neural Network
MFF	Multilevel Feature Fusion
MFFN	Multilevel Feature Fusion Network
RoI	Region of Interest
UAV	Unmanned Aerial Vehicle
VGG	Visual Geometry Group
YOLO	You Only Look Once

Chapter 1

Introduction

Farming production and quality are threatened by plant diseases, pests, and weeds, affecting farmers' production and economic loss. It signifies that around 15-25% food production will decline in India (Mahlein et al., 2018) and India is the largest contributor of undernourished people in the world, with around 194.4 Million people, or 14.37% of its population not receiving enough nutrition (Aditya, 2022) and around 720-811 million people were malnourished around the globe; it accounts for 9.9% of the world population. Agriculture products and their security are one of the significant anxieties of the world population. Food security is expected to be more than 9.7 billion population by 2050; expected 60% more food to be produced to feed such a huge population (Nations (2015)). Food squandering happening majorly in developing countries due to poor infrastructure and fewer investments in the production, harvest, storage, post-harvest, and processing phases. Various causes are degrading the quality and quantity of crops, such as implementing modern techniques, globalization, climate change, modern cultivation techniques with huge chemical fertilizers, and many more. However, the agriculture and food industries generally demand quality products. Cultivating a quality agricultural crop is challenging. Climate change is one of the sectors that affect plant diseases due to unusual behavior in the past few years, and it also affects the agriculture activities such as harvest and post-harvest activities; it majorly affects middle and small-scale farmers.

The emergence of plant diseases disquiets agricultural cultivation and production. If vegetation disorders are not diagnosed in time, food scarcity will aggravate (Faithpraise et al., 2013). Plant diseases and pest detection models need to be automated in agriculture. Since disease and pest detection is essential, the automated structure monitors the environmental conditions with various techniques and methodologies and minimizes the effect of disease and pests when early detection happens. Detection and prevention of diseases in an early stage is the most fundamental requirement to improve the production and caliber of the nature of the crop. The fundamental practice of plant disease is ascertained by domain expert involvement; this requires considerable human effort. Further, it does not produce accurate results always. Consequently, imprecise detection of disease guides to a humongous loss in the quality and quantity of the crop.

1.1 Plant Disease

When some causal agent interferes with a plant's normal structure, growth, function, or other activities repeatedly, an abnormal physiological process frequently results. This disruption of one or more essential physiological or biochemical processes in a plant causes recognisable illnesses or symptoms. Generally speaking, the main cause of plant diseases can be divided into infectious and noninfectious agents. Infectious plant diseases are brought on by pathogenic organisms, such as fungi, bacteria, mycoplasma, viruses, viroids, nematodes, or parasitic flowering plants. A contagious agent has the ability to grow both inside and outside of its host and spread to other vulnerable hosts.

Non-infectious plant ailments are caused by adverse growing conditions, such as temperature extremes, unfavourable oxygen-to-moisture ratios, toxic substances in the soil or atmosphere, and an abundance or deficiency of a key mineral. Since they are not living organisms that can reproduce inside of a host, non-infectious causal agents cannot be transmitted.

Plant diseases are getting affected by various environmental factors. Temperature is one of the factors; each disease has a certain temperature range for growth. Relative-Humidity is another factor; most of the leaf and fruit fungi diseases depend on relative humidity. Soil moisture affects root rot diseases. Excess watering with low oxygen and high CO2 is more prone to root rot diseases. Soil pH also affects plant diseases; certain pH scores need to be maintained depending on the type of crop. Soil fertility also matters; the lower the certain nutrients, the higher the chances of contagious diseases.

1.2 Kinds of Diseases

The diseases are consequential and harmful to plants since they avoid the photosynthesizes process of the plant life cycle; due to this, plants can not absorb the nutrients; this affects the development of plants and leads to lower quality and yield, resulting in economic loss and time (Lincy, 2021). The plants suffer from bacterial and fungal diseases.

Fungi Diseases:

A variety of dangerous plant diseases are brought on by fungi, which make up the majority of plant pathogens. Fungi are censured for the majority of vegetable plant diseases. By damaging plant cells or stressing plants, they harm plants. Infected seeds, soil, agricultural debris, neighboring crops, and weeds are sources of fungal diseases. Through the movement of contaminated soil, animals, people, equipment, tools, seedlings, and other plant material, as well as by wind and water splash, fungi are spread. They penetrate plants through stomata, which are naturally occurring openings, as well as wounds brought on by pruning, harvesting, hail, insects, various illnesses, and mechanical harm(Rosa-Márquez et al., 2003)(Punja et al., 2004).

Bacterial Diseases: Numerous major vegetable illnesses are brought on by pathogenic microorganisms. They must enter through wounds or organic plant holes since they cannot penetrate plant tissue directly. Insects, other diseases, and tool damage can cause wounds during tasks like trimming and plucking. Only when conditions are favorable for their growth do bacteria become active and provide a threat. They have the capacity to grow rapidly. High humidity, crowding, poor air circulation, plant stress brought on by excessive, insufficient, or irregular watering, poor soil health, and an imbalance of nutrients are a few reasons that can lead to infection (Rosa-Márquez et al., 2003)(M et al., 2021).

Different bacterial disease strains, or pathovars, can harm different vegetable crop kinds or cause several diseases in the same crop. For instance, pseudomonas syringae pv. syringae and P. syringae pv. phaseolicola cause various illnesses in beans, while Xanthomonas campestris pv. vitians and X. campestris pv. cucurbitae cause different infections in lettuce and cucurbits, respectively (Rosa-Márquez et al., 2003).

Infected plants often show apparent signs or sores on plant leaves, trunks, stems, roots, flowerets, or fruits. Figure 1.1 to Figure 1.6 shows some of the diseased plant leaf images. Figure 1.1 shows the cardamom plant leaf images, Figure 1.2 shows the cherry plant leaf images, Figure 1.3 shows the grape plant leaf images, Figure 1.4 shows the maize plant leaf images, Figure 1.5 shows the pepper plant leaf images, Figure 1.6 shows the potato plant leaf images, and Figure 1.7 shows the tomato plant leaf images. Diseases appear in the different parts of the plants depending on the nature of the plant and climate. Some diseases are more prone to rainy and winter seasons, and some are sunny. The majority of diseases appear in the leaf of the plant. Generally speaking, each illness or insect environment creates a single visual archetype that can be utilised to understand the anomalies. In general, plant leaves are a primary source of plant disease, and the illness may first start to manifest itself on the leaves of the plant (Ebrahimi et al., 2017).

Figure 1.1: Diseased Cardamom Plant Leaf Images

Figure 1.2: Cherry Plant Leaf Images

Figure 1.3: Grape Plant Leaf Images

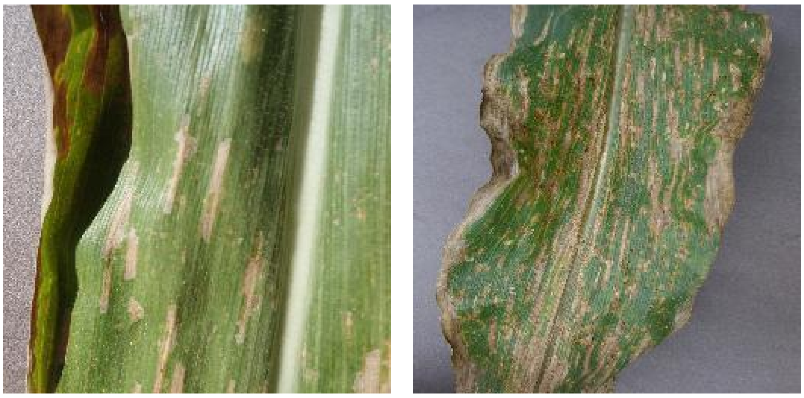


Figure 1.4: Maize Plant Leaf Images

Figure 1.5: Pepper Plant Leaf Images

Figure 1.6: Potato Plant Leaf Images

Figure 1.7: Tomato Plant Leaf Images

1.3 Deep Learning

Deep learning is a subset of Artificial Intelligence. The kind of data it uses and its learning strategies set deep learning apart from traditional machine learning. Machine learning algorithms utilise structured, labelled data to make predictions, therefore the input data of the model is used to identify certain features that are then organised in tables. This doesn't mean that it doesn't use unstructured data; rather, it means that if it does, it usually goes through some pre-processing to organise it. Deep learning eliminates some of the data pre-processing that is often required for machine learning. By automating feature extraction and handling unstructured text and visual data, these algorithms eliminate the need for human experts (Sarker, 2021).

To achieve an acceptable level of accuracy, deep learning systems require access to vast volumes of training data as well as computing capacity. Programmers did not have easy access to any of these resources prior to the era of big data and cloud computing. Because deep learning programming can create intricate statistical models directly from its own repetitive output, it can develop precise predictive models from vast volumes of unlabeled, unstructured data. As the internet of things (IoT) grows, it will become increasingly important because the vast majority of data produced by people and devices is unstructured and unlabeled (Janiesch et al., 2021). Figure 1.8 describes the basic deep learning architecture, consisting of input followed by a few layers of convolution and sub-sample layers, then two fully connected layers, and finally, the output layer. The complexity of the network varies from application to application with the corresponding dataset (Sarker, 2021) (Janiesch et al., 2021).

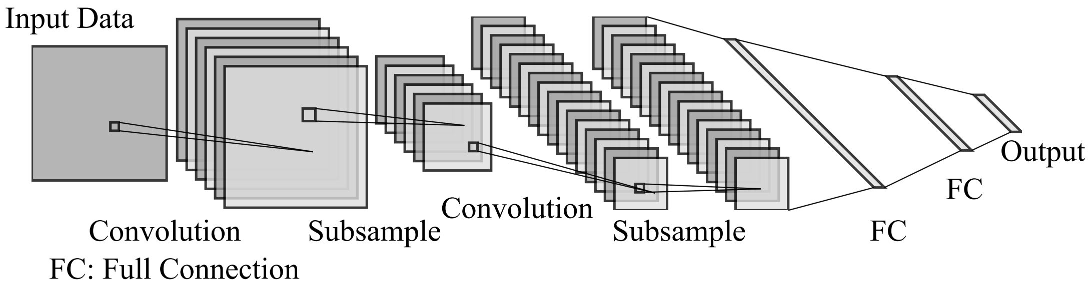


Figure 1.8: Basic Deep Learning Architecture (LeCun (2015))

1.3.1 Working of deep learning

Deep neural networks consist of multiple layers of interconnected nodes, each of which enhances the prediction or classification produced by the layer behind it. Calculations flow through the network using a process known as forward propagation (FP). The input and output layers of a deep neural network are the layers that are visible. The final prediction or classification is carried out by the deep learning model in the output layer after the data has been processed in the input layer (Sarker, 2021).

A different approach is called back propagation (BP), which calculates prediction errors using methods like gradient descent before iteratively travelling back through the layers to alter the weights and biases of the function in an effort to train the model. The predictions depend on the FP and BP of the neural network, and FP and BP minimize the errors accordingly. As the number of epochs increases, the neural network gradually improves the accuracy for a certain number of epochs. This describes the fundamental deep learning network in an unembellished way. The deep learning network varies in terms of complexity based on the type of task, such as classification or clustering with the precise application (Sarker, 2021)(Shrestha & Mahmood, 2019).

1.3.2 Role of hyperparameters in Deep Learning

A hyperparameter is a variable defined before the training of the Deep Learning Models (DLM); it represents the structure of the DLM based on the number of hidden layers. There are various hyperparameters, such as the number of neurons, different activation functions, learning rate, batch size, and the number of epochs. All these hyperparameters are tunable. The performance of the DLMs depends on these hyperparameters. To enhance the performance of the DLM, tuning the hyperparameters is essential to build a robust model. Selecting the correct hyperparameters is one of the challenges.

The performance of the DLMs varies with different datasets with different sets of hyperparameters. There is no standard way or algorithm to fix the number of layers and neurons or which optimizer is eminent for a particular dataset. Tuning the hyperparameters is discovering the prominent set of hyperparameters to build deep learning for a particular dataset.

The number of layers is set to be small for simple problems, and more layers are required for complex issues. Similarly, the number of neurons varies from layer to layer or can be constant for all the layers; the number of neurons is generally more for complex problems since, to solve the problem, the best features need to be extracted. Less number of layers leads to underfitting, and more layers lead to overfitting (Srivastava et al., 2014); choosing the correct number of layers is one of the limitations. To avoid overfitting, insert the regularization layers in the network. Batch Normalization (BN) and dropout are the regularizations; Batch Normalization (BN), normalizes the values passed to it for each batch, and dropout drops the neurons as set by the dropout rate (Tsai et al., 2020).

Various activation functions are available, and the input changes from one layer to another based on the activation functions applied in each layer. Activation functions determine the output for each input value; the corresponding output values are fed to the next layer. This process continues till the last layer. The different activations are Softmax, Tanh, ReLU, Sigmoid / Logistic, Softsign, ELU, Leaky ReLU, and Linear function.

1.4 Plant Disease Detection using Artificial Intelligence

The onsite visual investigation is the primary method domain experts follow to detect plant diseases, which requires a significant amount of human resources, time, and expensive devices. It may not produce fruitful results. Agriculturists with less knowledge may misjudge and usage of pesticides or insecticides indiscriminately during the screening process, resulting in dispensable economic losses.To solve these problems, image processing with automatic leaf disease diagnosis is crucial. Agrarian products must be supervised and decisions made about them based on timely perception, which is the foundation for efficient coercive measures and supervision of plant leaf diseases.

More accurate plant disease detection is essential to avoid loss in farm production and improve crop quality. Various modern technologies, such as sensor-based and image processing-based techniques, were employed to detect plant diseases (Bravo et al., 2004)(Moshou et al., 2005). These techniques can recognize plant diseases relatively fast, though they need expensive sensors (Lu et al., 2017).

Deep Learning is the technology at the frontiers for the classification of the images (Brahimi et al., 2017). The Deep Learning-based technique's significant advantage is that it spontaneously extracts the attributes. Deep Learning-based approaches have been employed in several fields such as audio, video, signal processing, precision agriculture, text-based classification, and plant image classification (Ji et al., 2020)(Liang et al., 2019)(Kamal et al., 2019).

Deep learning is an emerging technique that fuses data analysis with images and produces promising results. In recent studies, deep learning has been employed in numerous applications such as signal (Deng & Yu, 2014), image (Krizhevsky et al., 2017), video processing, object detection, medical image segmentation, biomedical image segmentation, and remote sensing (Wang et al., 2018). Various agriculture applications also employ deep learning, such as fruit classification, to categorize agricultural products and recognize plant diseases (Dyrmann et al., 2016).

In a recent study, methods based on computer vision and machine learning were created for the identification of plant leaf disease. The complexity of the background and the severity of the disease as a result of the photos being captured in real-time scenarios from the farm field are two major obstacles to real-time plant disease diagnosis.

Detection of plant diseases precisely with appropriate precaution measures is essential, and complex background images encourage a more sophisticated approach that helps the farmers detect the diseases by clicking the images of the plant. The result of this enhances crop production, reduces the cost of production, and enhances the quality of the crop. Which helps the farmers to become economically strong.

1.5 Challenges in Plant Disease Detection

Plant disease classification approaches using Deep Learning rely on plant leaf images which have many challenges. The background of the image is one of the challenges; segmentation of the Region of Interest (RoI) is another challenge that is very hard to extract. Image capturing conditions is another challenge that is hard to control due to environmental conditions. If the images are not captured in the same condition, they may present different characteristics, which may significantly impact on identifying the diseases. The image's orientation is another challenge; a few lesion parts may be out of bounds due to the capturing angle.

Segmentation of the disease symptoms is also a challenge. The most lesion does not have well-defined boundaries. So, the segmentation of disease symptoms is very hard. Some parts of the lesion gradually disappear, which affects the performance of the disease detection model.

Diseases have different stages, i.e., the initial stage to the final stage, and the lesion size, color, and shape differ from stage to stage. Based on the expertise, an expert easily identifies disease in any one of the stages accurately; however, it is difficult for a novice to identify symptoms of diseases for all the stages. A symptom may have multiple disorders; identifying such disorders is also a challenge. Different disorders may have similar kinds of symptoms which is also another challenge. If two or more disorders have similar symptoms, then it is tough to segment and separate disease symptoms.

Semi-supervised learning has a few challenges, such as limited data and imprecise labeled data leading to further erroneous results; with this, developing a model becomes computationally costly.

Unsupervised learning also suffers from limited data. Further, it will not extract the relevancy and spatial features, and interpreting between classes is a complex task.

The performance of the DLMs relies on various hyper-parameters and combinations of those. Selecting the best and most promising hyper-parameters is one of the challenges. Fine-tune the hyperparameters for DLMs precisely, and it is a very difficult process to find the optimal hyperparameters for classification purposes.

The development of a real-time, cost-effective Deep Learning-based plant disease detection model is an additional challenge. Since features of the lesion vary with the geographic position, train, and test, the classification with a geography-specific dataset may not be appropriate.

1.6 Research Outline

The fundamental architecture of the plant disease detection method is depicted in Figure 1.9. This method includes a stage referred to as the "dataset preparation stage," which is responsible for collecting both primary and secondary datasets. The primary datasets are produced by collecting photographs from farms, and the secondary dataset is collected from the benchmark dataset. Finally, the most relevant dataset for the study is chosen from among all of the datasets collected.

The second stage of processing is called preprocessing, and it involves applying a variety of methods to the data that was obtained. At this point (stage three), choose

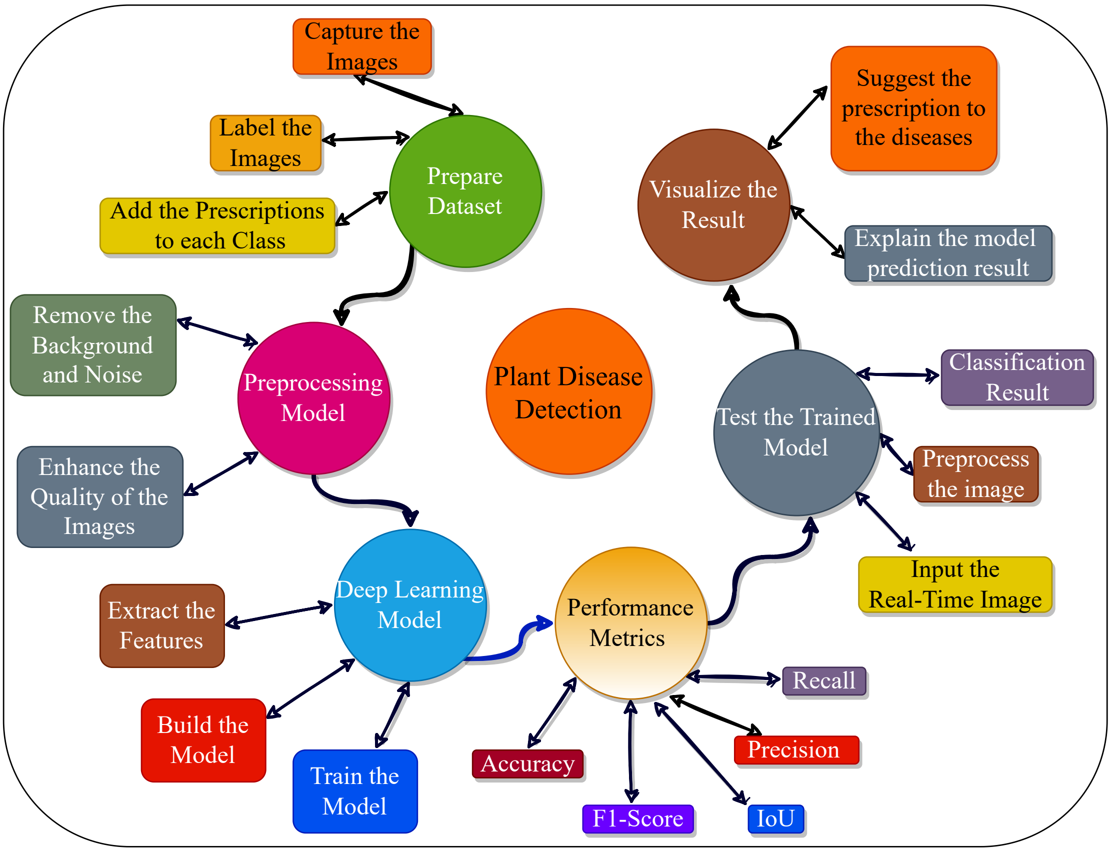


Figure 1.9: General Plant diseases detection approach

the deep learning model that will be used for categorization. In the fourth step, multiple types of performance measurements should be used to evaluate the deep learning model. In the subsequent step, the deep learning model will be put through its paces using testing photos; finally, the outcome will be visualized, and recommendations will be made regarding possible preventative measures.

1.6.1 Motivation

The climate crisis is threatening global food security; according to Food and Agriculture Organization report 2020, 189.2 million people were undernourished in India. By this measure, 14% of the population is undernourished in India (Nations, 2015). Due to climate change and modern cultivation with huge chemical fertilizers used in agriculture, there is an increase in plant diseases. Plant diseases are another factor, along with climate change, for declining food production globally. Plant diseases damage / degrade the crop's quality, affecting the farmers economically. Damages in the crops / low-quality crops affect the country's agricultural exports, which also economically hits farmers. A huge amount of farmers' economy is spent on plant disease protection and their management. Detection of plant diseases precisely with appropriate precaution measures is essential, and complex background images encourage a more sophisticated approach that helps the farmers detect the diseases by clicking the images of the plant. The result of this enhances crop production, reduces the cost of production, and enhances the quality of the crop. Which helps the farmers to become economically strong.

1.6.2 Research Contributions

This study presents an overview of several different approaches to the identification of plant diseases that make use of deep learning. On the other hand, deep learning-based systems automatically extract the features and produce results that are superior to those obtained by more conventional methods. Transfer learning enables individuals to complete specific tasks with a more manageable amount of data. This study considered 160 different research works in plant disease detection or classification. It also has different plant disease datasets and various deep-learning models for plant disease detection or classification. Based on the review of different plant disease detection approaches framed some of the research gaps and challenges. Which lays the foundation for further study.

As a preliminary work, proposed a multi-convolutional layer based CNN. This study conducted an empirical study on plant disease detection three different binary datasets to analyze the effect of epoch sizes of CNN model in plant disease detection and understand the working of CNN model, and hyper-parameters.

A detection method for plant leaf diseases using ensembles of deep learning models is proposed. In the suggested method, the base models employed were AlexNet, ResNet50, and VGG16. This is due to the fact that each base model has a distinct manner in which it classifies the images, and each model extracts independent characteristics. The proposed method's primary purpose is to reduce the number of incorrect classifications as much as possible; this is accomplished by utilizing the aforementioned three distinct Deep Learning-based models; the ultimate classification result is determined by taking into account the outcomes of the majority of the classifiers. This research used a dataset containing information about cardamom plants to meet the difficulty of image-capturing settings and compare the results of experiments conducted with those of existing methods. When applied to the cardamom plant dataset, the new method achieved a higher level of precision than the existing EfficientNetV2 model, which had been utilized in previous research. This study also takes into account the classification time, which is an essential component for demonstrating how quickly the suggested method classifies the provided input data in the context of real-time operations. Tomato and cardamom plant datasets also are used in this study to analyze the working of the ensemble model.

This study collected a cardamom plant dataset with three classes in the farm field, where the images are acquired using various mobile phones against a complex background. This was done to stimulate disease detection on noisy or low-resolution photographs with complex environments. In this study, a cardamom plant dataset with three classes was obtained. In order to identify plant diseases in farm fields, it is vital to have a plant leaf disease detection method that is efficient. In this regard, the cardamom plant leaf disease detection approach is provided, in which the cardamom plant leaf dataset was obtained from a field with a complicated background. Because the images are related to other characteristics such as the background of the image, environmental factors such as illumination, and the angle of the capturing conditions, segmenting and identifying diseases in real-time photos is a tough process. This is because the images themselves are associated with other aspects.

The proposed method uses the U^2-Net architecture to get rid of the complicated background. This method generates results without degrading the quality of the original image; hence it is an improvement over the existing methods. In this work, CNN, EfficientNet, and EfficientNetV2 models were trained for classification rather than utilizing the pre-trained weights for EfficientNet and EfficientNetV2. Experiments covering a wide range of scenarios were carried out on the CNN, EfficientNet, and EfficientNetV2 models. In terms of performance, the EfficientNetV2-S and EfficientNetV2-L models were superior to the others. The grape dataset was also employed in this work to assess the workings of the proposed models and the performance of those models.

A classification strategy for the leaves of the tomato plant is proposed in this paper. It does this by employing a Multilevel Feature Fusion Network, which searches for and extracts the essential elements needed to categorize photos of tomato plant leaves. An Adaptive Attention Mechanism (AAM) with channel, spatial, and pixel attention is used, emphasizing the information inside each channel to reduce the amount of perceptual loss. By utilizing the inter-spatial interaction of features, spatial attention is utilized in order to concentrate one's attention on the specific position of the relevant features. Utilizing pixel attention allows for the extraction of the important elements that contribute to increased potential learning.

The proposed method additionally includes a pesticide prescription module that outlines the various pesticides that can be used to treat the tomato plant's identified ailment. A number of experiments are carried out in order to determine whether or not the proposed method is capable of classification. In comparison to other methods, it achieved the maximum possible level of categorization accuracy. To analyze the working of the proposed model on different datasets, grape and cardamom plant datasets also used in this work.

1.7 Outline of the thesis

The rest of the thesis is organized as follows: Chapter 2 presents a Literature review on various plant disease detection using different approaches. Chapter 3 describes Ensemble deep learning-based plant disease detection and analyzes the effect of epochs sizes on CNN. Chapter 4 addresses complex background leaf images in plant disease detection. Chapter 5 describes tomato plant leaf detection with appropriate precautionary measures. The conclusion and future work are discussed in Chapter 6.

Chapter 2

Literature Survey

The most common traditional methods that are followed to detect plant leaf disease are visual inspection and biotic techniques[1] [2]. Since traditional methods require manpower and a huge number of sensor equipment, modern technologies such as computer vision works like a human; so, the various tasks that can be performed by computer vision are a) object recognition or classification, b) classification and localization, c) object detection, and d) image segmentation.

Various Deep Learning-based approaches are described in Figure 2.1, which are used in various plant disease classification and they are classified as supervised and unsupervised models. Supervised learning is a task where the machine learns from a training dataset with supervision (i.e., pre-existing labels). In contrast, unsupervised learning is a task where the machine learns the pattern without supervision (i.e., without pre-existing labels).

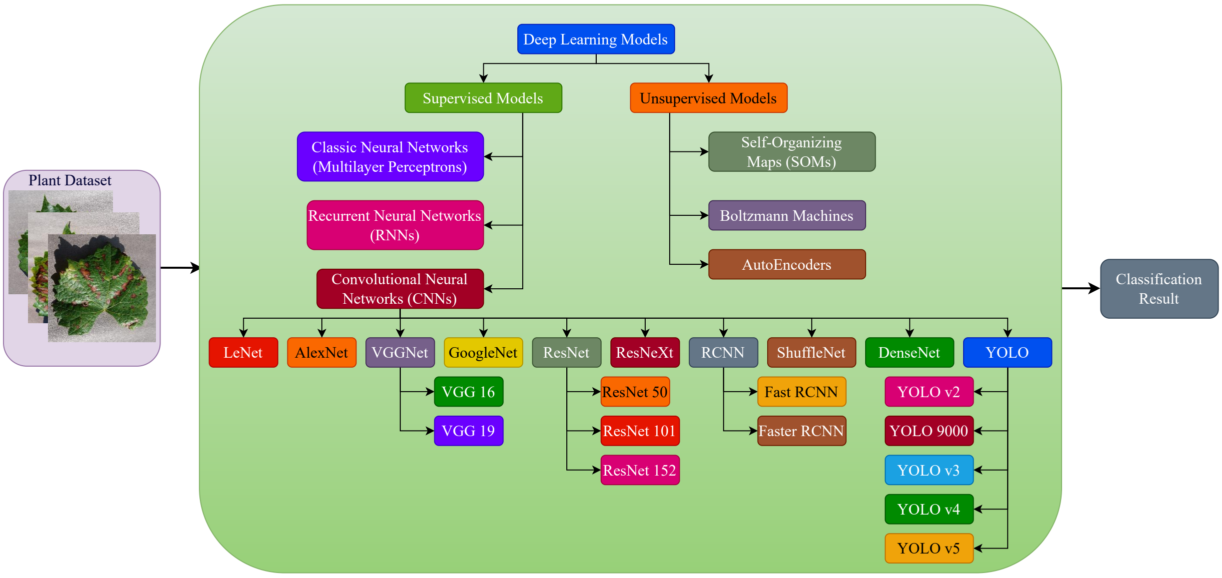


Figure 2.1: General Classification of Deep Learning Models

Classic Neural Network is a multilayer perceptron that consists of at least two layers, which can be used for data visualization, compression, and data encryption. **Recurrent Neural Network (RNN)** is a neural network that utilizes sequential information to recognize the patterns such as speech recognition, text, and the like. The most

[1]Biological Indicators: Certain indicator plants are susceptible to specific diseases.

[2]Microbial Detection:Utilizing specific microbes or microbial products to detect the presence of pathogens or disease-causing agents in the plant.

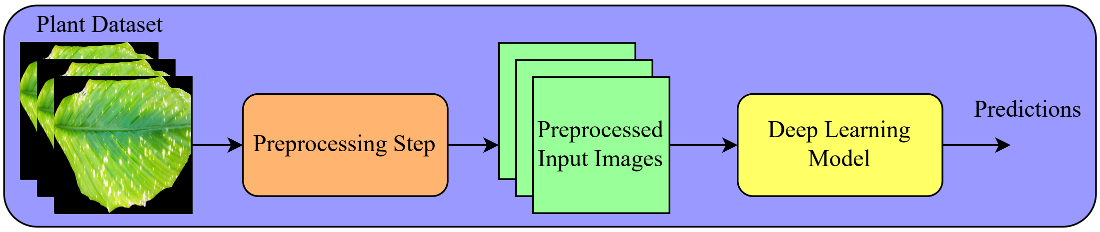


Figure 2.2: Disease Detection System

common RNN is Long Short Term Memory. **Convolutional Neural Network** is a neural network that consists of convolution and pooling operations with different activation functions. **Self-Organizing Maps** is an unsupervised model which mainly works for dimensionality reduction and for knowledge gain regarding the dataset where output was not known. **Boltzmann Machines** is an unsupervised model; it does not move in a defined direction. Unlike other DLMs in this, all the nodes are connected in a circular fashion. Since it learns to regulate, it can be used to monitor specific systems. **Auto-Encoders (AE)** automatically encode the input data so as to reduce the dimensionality of the input data into lower dimensions and reconstruct the output from the compressed input. Various types of AE are available such as sparse AE, denoising AE, contractive AE, and stacked AE.

Plant disease classification using Deep Learning methods overcomes the limitations of traditional methods. Various CNN architectures are available such as LeNet, AlexNet, VGGNet, DenseNet, GoogleNet, ResNet, etc.

Figure 2.2 describes the plant disease classification system based on Deep Learning techniques. Deep Learning-based plant leaves detection approaches that are categorized based on internal architecture are shown in Figure 2.3. It broadly classifies the proposed approaches into Single Network-based and Hybrid models.

2.1 Single Network-based approaches

This section provides a systematic study of various deep learning-based plant leaf disease detection approaches that use color images. Most of the work focused on single plant leaf disease detection, and few studies concentrated on multi-plant leaves with various diseases.

Apple Leaf Disease Detection: Jiang et al. (2019) proposed a recognition of apple plant leaf disease. Rainbow Concatenation (RC) Single-Shot Multi-box Detector (SSMD) is used to detect the type of object and coordinates of the corresponding bounding boxes. RC is used to enhance the recognition of smaller objects which are not

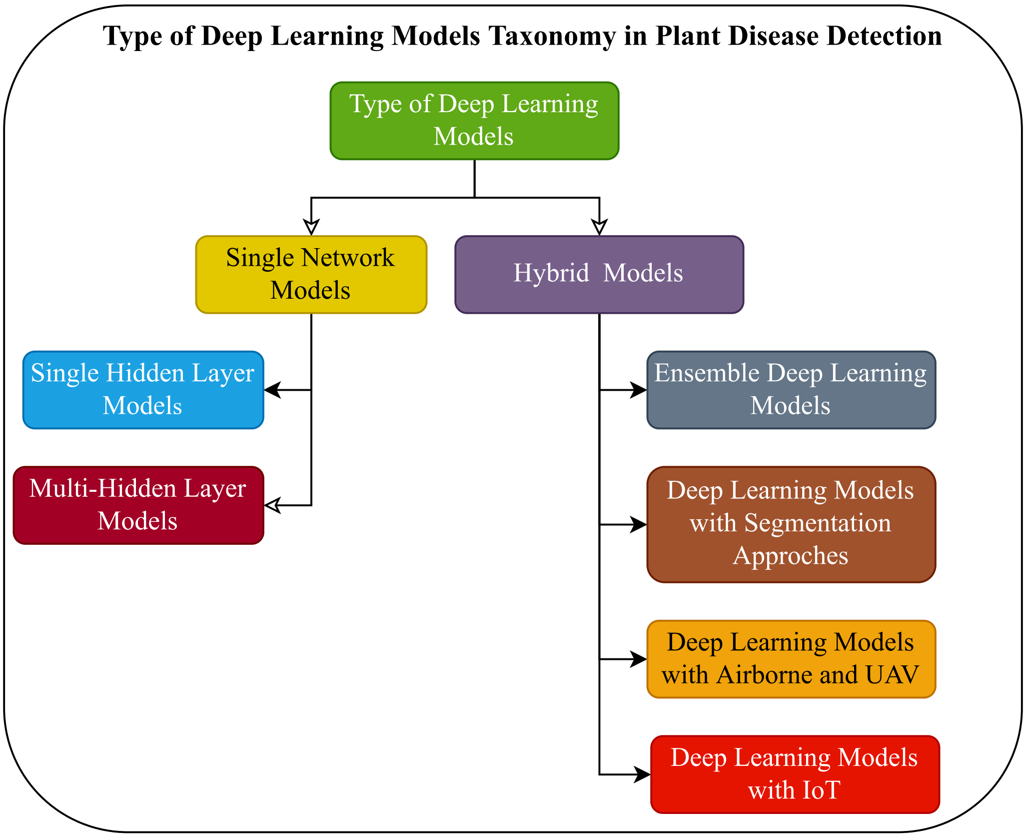


Figure 2.3: Taxonomy of Deep Learning Models

recognized by SSMD alone. Deep CNN (DCNN) was employed, which extracted the eminent features to classify five different diseases of the apple plant. The limitation of this work is the two-stage process; a system developed is not end-to-end. After completion of the first stage, the second stage trains the Deep CNN. The number of images used in this work is very less, which has only 2029 images for five classes; the authors used a data augmentation Technique (AT) to enhance the dataset.

One of the studies (Zhong & Zhao (2020)) proposed a method for apple plant leaf disease detection by employing Deep Learning. The authors used DenseNet-121 with three combinations; DensNet-121 with regression attained 93.51% detection accuracy, DensNet-121 with multilabel classification attained a detection accuracy of 93.31%, and DensNet with focus loss function achieved 93.71% detection accuracy. As mentioned earlier, the authors infer that the methods overcome oversampling and undersampling. Based on RC SSMD, the authors have developed an Inception and Rainbow (INAR) SSMD, where it has three parts, pre-network, attribute extraction, and fusion structure. The authors have developed a VGG-INCEP model by using a combination of VGGNet and Inception, where Inception is used to enhance the feature extraction. The limitation of this work is the images used in this work were taken in the laboratory. The authors have used pre-trained models.

Elfatimi et al. (2022a) proposed beans disease detection using MobileNetV2. Evaluated the performance of the hyperparameters such as optimizers, learning rate, and batch sizes on the beans dataset. The optimizers used in the study are Adam, Nesterov Momentum Adam, Stochastic Gradient Descent (SGD), RMSprop, and Adagrad. The learning rate used in the study is 0.001, 0.0001, and 0.00001. The different batch sizes used in the study are 32, 64, and 128. This empirical work compares the different hyperparameters such as batch size, learning rate, and optimizers using a pre-trained model. Abed et al. (2021) worked on beans disease detection, segment the leaf images with the background using UNet, and then conducted a empirical study on five different deep learning models. DenseNet121 outperformed the other models.

Banana Leaf Disease Detection: Seetharaman & Mahendran (2022) proposed banana leaf disease detection by employing Region-Based CNN (R-CNN). Preprocess the collected images with a histogram-based pixel localization approach which minimizes the training time. Then segmentation is performed using the region-based edge normalization technique, which minimizes the data complexity, further extracting the features using Gabor-based binary patterns with CNN. For classification, authors have used different approaches, such as R-CNN, CNN, DCNN, K-Nearest Neighbours (KNN), and Support Vector Machine (SVM). R-CNN attained an accuracy of 98% among other classifiers. The system has three independent stages, such as localization, feature extraction, and classification, with different approaches, which makes the system more complex. The dataset size is very small, which consists of 1875 images with four classes.

Citrus Leaf Disease Detection: Syed-Ab-Rahman et al. (2022) proposed citrus plant disease detection by employing a deep learning approach. Preprocess the dataset by employing a histogram-based equalization approach which manages all the images into a single intensity range and applies different data augmentation to enhance the dataset size. ResNet101 is used for feature extraction and to find the RoI, and region proposals are generated to train the network end-to-end anchor-based approach Region Proposal Network (RPN) was utilized. The dataset used in this study is small and does not use data AT to enhance the dataset. The two-stage process makes the system computationally costly.

Coffee Leaf Disease Detection: Esgario et al. (2020) have proposed a coffee plant leaf disease detection and severity estimation system using Deep Learning-based techniques. The multi-task approach was used with various Deep Learning models. In the multi-task model independent, Fully Connected (FC) layers were added in parallel to the classifier. The limitation is the shallow representative of the dataset, which considers the prime biotic stresses of the coffee leaves, and other diseases with a more enhanced dataset required.

Cucumber Leaf Disease Detection: Symptom-wise cucumber plant leaf disease classification approach was proposed by a few researchers (Ma et al. (2018a)), and experiments were conducted by using a dataset that consists of four different types of symptoms such as Leaf Spot (LS), Downy Mildew (DM), anthracnose, and Powdery Mildew (PM). This approach segments the symptom images using the comprehensive color feature (CCF) with region growing. The CCF comprises three components: excess red index, H from HSV (Hue, Saturation, Value), and b* from CIELAB color space, which discriminates the backgrounds of the disease spots. For symptom image segmentation, the region growing method was used on the color features.The limitation of this work, it is a two-stage; the first stage is for segmenting the diseased part of the leaf, then training and testing using CNN. This affects the computational cost and deteriorates the classification result.

Zhang et al. (2019) proposed a cucumber plant leaf disease detection approach using a Global Pooling Dilated CNN (GPDCNN) by incorporating the convolution with global pooling. Widened convolution was added instead of the kernel of an AlexNet; it intensifies the feature extraction ability and recovers the spatial resolution. To minimize the numerous training parameters and steer clear of overfitting global pooling layer was employed. Multiscale convolutional kernels were used to take out the multiscale attributes from the input image. This approach improved the detection accuracy and robustness of the model. The limitation is that this approach could enhance the performance by exploring the crucial role of probabilistic graphical models.

Zhang et al. (2020a) proposed an approach to classify the cucumber plant leaf diseases which has a complex background and which have high similarities, such as PM and DM, by employing EfficientNetB4 with ranger optimizer; the ranger optimizer is a combination of RAdam and Lookahead optimizer and attained 96% detection accuracy. The limitation is that the system was not tested in farmland.

Grape Leaf Disease Detection: Research (Cruz et al. (2019)) proposed Grapevine Yellows (GY) in grapevine (Vitis vinifera L) recognition using 6 CNN models such as

AlexNet, SqueezNet, GoogleNet, Inception v3, ResNet-50, and ResNet-101 with leaf clipping images of Vitis vinifera. The authors concluded that the ResNet-101 model outperformed as compared to ResNet-50, and SqueezNet showed the least performance among all the models. The limitations are that this is a comparative study with different pre-trained DLMs. This research could consider the different leaves occlusion and light illumination; further, this study considers the state-of-the-art approach for segmentation. The dataset used in this study was imbalanced.

Zinonos et al. (2021) proposed grape plant disease detection using deep learning with IoT. Employed Long Range (LoRa), which is a wireless modulation technique for low power and low data rate applications. Farmers send images from their farmland using LoRa nodes; six such nodes are deployed in the different farmlands; the farthest is 600 meters away. All the images collected from the LoRa are resized to 255×255. DLMs such as MobileNetV2 and ResNet50 were employed to detect the diseases. LoRa has a limitation that the node can be active for a maximum of 36 sec/hour; capturing the grape leaf image and sending it to the backend is challenging through LoRA due to its 1% duty cycle.

Shantkumari & Uma (2022) proposed an approach by employing CNN and improved KNN to detect the grape plant diseases. It has three stages; in the first stage, minimizing the noise by utilizing histogram-based gradient patterns is used to enhance the detection accuracy; these obtained features are depicted with the pixel encoding approach. In the second stage, segmenting the lesions in the leaf is done by the Adaptive Snake Algorithm (ASA) approach. In the third stage, to classify the images, CNN and improved KNN were employed. The limitation is that the features extracted using different feature extraction techniques, further ASA for lesion segmentation, and KNN classifier are independent systems; this approach is not suitable for in-farm plant disease detection.

Andrushia & Patricia (2020) proposed a grape plant diseases detection; two-stage preprocessing was done, leaf background subtraction was used, then applied histogram-based equalization approach was used to remove the noise. Artificial Bee Colony (ABC) is employed to extract prominent features such as color, texture, and shape. 110 features were selected to train and test the proposed SVM classification method. The limitation is that the authors have used just 350 images, in that 175 images with four classes for training the model.

Greengram Leaf Disease Detection: Kumar et al. (2020) proposed IoT based green gram plant disease detection approach by employing sensor data and Multi-Layer Perceptron (MLP). Sensor data such as humidity, moisture, and temperature; satellite data such as rain, pressure, wind speed, etc., then fuse the sensor and satellite data. Artificial Neural Network (ANN) with ReLU activation, binary cross-entropy as loss, and adam optimizer is the work proposed by the authors. The number of sensors used in this work is less, and spatial information's not considered to detect plant diseases.

Maize Leaf Disease Detection: LeNet-based DCNN-based maize plant leaf disease classification approach was proposed (Ahila et al. (2019)). The maize leaf is influenced by various factors caused by bacteria and fungi. Bacterial diseases are northern leaf blight and gray LS, and fungal disease is common rust. All the raw images were resized to 64×64; then, Principal Component Analysis (PCA) was employed on the resized images to make the features less correlated. The authors concluded that the proposed approach obtained a detection accuracy of 97.89% with a kernel size of 3×3. The images used for the study are from a conditioned environment.

Haider et al. (2021) proposed a wheat disease detection approach by employing a multimodel dataset. The dataset consists of an image with the bunt, sooty mold, and fusarium blight disease samples. The authors collected the images on the farmland and used crowdsourcing, where 200 farmers from different regions shared the wheat plant disease and healthy images. Conducted an empirical study by considering various DLMs such as AlexNet, VGGNet, ResNet50, and sequential CNN. This approach could be enhanced for early detection of the diseases of the crop and also could utilize the UAV to monitor the crop for early detection in in-farm to take timely action.

Yu et al. (2021) proposed an improved DLM by employing K-means, VGG19, and CNN to detect the three corn diseases. K-means is used as a preprocessing; each pixel is split into different clusters such as 4, 8, 16, 32, and 64, and each time the cluster center is updated until all the pixels passed; further proposed a CNN model by considering $0^{th}, 5^{th}, 10^{th}, 19^{th}$, and 28^{th} layers of VGG19 to extract the features and softmax as an activation function in the classification layer. The authors could consider the different kinds of diseases with different conditions. The performance could further improve with state-of-the-art optimization approaches such as Swarm Optimization (SO) with cross-fold validation.

Amin et al. (2022) proposed an approach to extract the most efficient features by employing EfficientNetB0 and DenseNet121 to detect corn plant diseases. In DLMs, the first few layers extract the global features, and the deeper layers extract the local features. Each DLM extracts different features to enrich the feature set and enhance the classification result, fusing the features extracted by EfficientNetB0 and DenseNet 121. This work could also use augmentation approaches with different DLMs and feature fusion techniques.

Singh et al. (2022) proposed maize plant disease detection approach by employing AlexNet. It has 11 layers with convolutional, batch normalization, max-pooling layers, ReLU activation function for intermediate layers, and softmax activation for the classification layer. Conducted an empirical study with 25, 50, and 100 epochs and attained an accuracy of 99.16% for 100 epochs, and the performance is better than Artificial Neural Network, SVM, and VGGNet. The limitation is that this study is a comparative study using a pre-trained AlexNet with two kinds of diseases. This could be explored with all the diseases of maize, and the authors could use state-of-the-art DLMs to classify the images.

Mango Leaf Disease Detection: Singh et al. (2019) proposed mango leaf Anthracnose disease detection using Multilayer CNN (MCNN). This work initially classifies the input image as a mango leaf or not, then recognizes the mango leaf as either healthy or diseased. For this study, the authors used two classes of mango leaf and two classes of other plant leaf datasets. The limitation is initially the images are checked with mango leaf or not, for this authors have used only one set of other plant leaf images, this could be extended with many other plants, then the authors have used to detect only Anthracnose disease of mango leaf, this also could be extended with all the diseases of the mango tree.

Rao et al. (2021) proposed mango and grape leaf disease detection by employing Transfer Learning (TL) using AlexNet with ReLU activation function, SGD as an optimizer, learning rate with 0.0001, and dropout layers to avoid the overfitting and attained 99.03% detection accuracy. The dataset used in this study is a self-acquired dataset. In this work, the authors have trained the model with just eight epochs, the early detection of the disease is not handled, and the images used in this experiment are captured in controlled conditions, which do not have external factors.

Mia et al. (2020) proposed mango leaf disease detection based on SVM and attained 80% detection accuracy. The authors have developed the model using a neural network with an ensemble approach using a self-acquired dataset. The model has not achieved good performance with respect to accuracy and has not used any image segmentation techniques since the images were captured by UAV. Saleem et al. (2021) proposed a mango leaf disease detection using CNN to extract the features; the authors have extracted the features based on the structure of the mango leaf vein, further classified the images using 10 different Machine Learning classifiers, and attained a 99.5% detection accuracy with SVM. The authors have collected 135 images with three classes. The dataset size is very tiny to train the deep learning model.

Peach Leaf Disease Detection: Zhang et al. (2019) proposed peach leaf disease classification by employing TL-based on AlexNet (TLAlexNet). This method was compared with other Machine Learning algorithms such as K-Nearest Neighbors, SVM, and Backpropagation. The t-Distributed Stochastic Neighbor Embedding (t-SNE) algorithm visualizes the attributes of the convolutional first and the last layer. The authors have used only one disease of the peach plant, and the dataset size is very small to use the Deep Learning approaches.

Pepper Leaf Disease Detection: Ahmad et al. (2021) proposed pepper plant disease detection by addressing class imbalance and overfitting issues. The authors have employed data synthesis and Generative Adversarial Networks (GAN) data argumentation to address the class imbalance since class imbalance deteriorates the model's performance. The authors have proposed a step-wise TL; instead of freezing out the layers, if the loss saturates, un-freeze the layers and train the model for a certain number of epochs. The authors observed the best result at 60% frozen layers. For the experiment, various DLMs were employed; MobileNetV3 performed better compared to other models. The number of images per class is not mentioned in the article. This work further improved with UAV images with a cloud approach and state-of-the-art Deep Learning approaches with an image segmentation approach to handle the complex images and detect the diseases in the farmland.

Mathew & Mahesh (2022) proposed a pepper disease detection approach by employing YOLOv5. Used pepper dataset from PlantVillage (PV) for training and the model has been tested by using the images collected from farmland. YOLOv5 is an object detection model; it builds a bounding box for the detected object. Augmentation Technique (AT) was employed, such as scale, color, and mosaic, where mosaic augmentation fuses the images with a specific ratio, which resolves the issue of small object detection. This work used only the disease of pepper; this could be enhanced with all the diseases of the pepper plant.

Rice Leaf Disease Detection: A DCNN-based paddy plant leaf disease- detection approach was proposed by another study (Lu et al. (2017)) to ascertain 10 common rice plant diseases; its motivation is to provide a simple system to detect the diseases in early-stage. The DCNN extracts the attributes automatically from the raw input images by using the sparse Auto Encoder (AE). The limitation is that this work further improved with state-of-the-art object detection algorithms such as Boltzmann Machine and also trained the model with unlabelled images.

Some studies proposed an optimized DNN for detecting and classifying paddy plant leaf disease (S & Vydeki (2020)). It uses K-Means Clustering (KMC) to segment the images into healthy parts and diseased parts. Extract the color and texture features by employing HSV and Gray-Level Co-Occurrence Matrix (GLCM), respectively. Then, classify the data using optimized DNN, where DNN updates the weights by setting a threshold. When an attained classification result is unsatisfactory, then the feedback is sent back to the segmentation stage. The weight updating task is tedious; this takes several rounds when the dataset is huge.

Joshi et al. (2022) proposed rice leaf diseases classification by employing CNN. The authors have collected 442 images from the farmland belonging to healthy, bacterial, and fungal classes. The authors proposed two superficial convolutional layers with batch normalization and max-pooling followed by three FC layers and a softmax activation function. The limitation is that the images used in this work were very few, with 442 images with three classes.

Patil & Kumar (2022) proposed rice leaf disease detection by employing CNN with IoT to extract the data such as moisture, nitrogen, potassium, and phosphorus details. Along with soil and environmental details, 3200 RGB images were collected using the camera. The authors have proposed the fusion network model by employing an MLP network for sensor input and CNN for RGB data, then fusing the feature extracted in two different models to detect the rice disease. This work could further improve by collecting hyperspectral images using sensors and UAV images which makes the system more robust. To the best of our knowledge, the details such as temperature and humidity, and soil moisture details not provide information to detect the diseases in the plant.

Soybean Leaf Disease Detection: Wu et al. (2019) proposed a Deep Learning-based soybean plant leaf disease recognition approach by using a data AT. The effect of batch size and the number of iterations on the classification capability of the proposed approach is also discussed. The observation is that this is a comparative study to measure the performance of the model with different hyperparameters variation, such as the number of iterations and different batch sizes.

Strawberry Leaf Disease Detection: Shin et al. (2021a) proposed strawberry plant leaf disease detection approach, state-of-the-art Deep Learning approaches were employed with the strawberry dataset, and the strawberry plant leaf images were captured from Baltimore Farm, Millen Farms, USA. The limitation of this study is that it considers the infected images, not specific to plant is affected by what kind of diseases; this work improved with different diseases of Strawberry plant with complex images.

Sugar-Beat Leaf Disease Detection: Updated Faster R-CNN (FR-CNN) approach was employed to classify the plant leaves spot disease in sugar-beet (M & Adem (2019)). The authors claim that FR-CNN could not segment the diseased part of the leaf as a disease and the healthy part as healthy. Consequently, some of the healthy parts were segmented as a disease due to shadow, and also, all the disease parts are not recognized correctly; so, there is a need to change the parameters of the CNN by updating the FR-CNN. The changes were made in CNN parameters in accordance with the input images.The limitation is that the number of images used in this study is very small, which consists of 155 images for four classes to train the DLM.

Nagasubramanian et al. (2021) proposed an approach by employing CNN and ensemble SVM with IoT to detect sugar beet diseases. The authors used various sensors to collect information such as temperature, humidity, soil moisture, etc. The authors have collected 600 hyperspectral images by using HySpexHD. To classify the diseases of sugar beat, ensemble SVM is used, which is non-linear with multilayer classifiers, and also CNN is used. The authors did not disclose the number of images collected and used for training and testing the model. This model further improved to find the nutritional deficiency of the plant using various sensors to collect the soil, climate, and environmental information.

Tomato Leaf Disease Detection: Deep Learning-based approach is proposed to classify and visualize the symptoms of tomato disease (Brahimi et al. (2017)) by considering the PV dataset (Hughes & Salathé (2015)) that consisted of 14,828 tomato leaf images with nine different diseases. Two CNN models, such as AlexNet and GoogleNet, were used with the intention of refining plant disease detection. Pre-processing steps are applied to avoid the background influence on detection accuracy; thus, the background of the image was replaced with black color, and all the original images were re-scale to 256×256. The feature extraction step was used to extract the features using wavelet moment, Gabor wavelet transforms, and color moment. The occlusion method is used to visualize the symptoms of the disease. The authors have used a pre-trained model for training the model, this can be improved using state-of-the-art algorithms to minimize the size of the trained model and computational cost, and the images used in this study are taken in a conditioned environment.

Abbas et al. (2021a) proposed tomato plant disease detection approach based on tuning the hyperparameters of DenseNet121 with softmax as an activation function with a learning rate of 0.0001; two new convolutional layers were added to the DenseNet121 just before the FC layer with ReLU activation. Employed conditional generative adversarial network to enhance the dataset. Limitations of this work are, used the images taken in a conditioned environment, and the different stages of the plant diseases are not considered.

Vadivel & Suguna (2022) proposed a tomato plant disease detection approach by using a PV dataset and fast-enhanced CNN with the mask, extracting the RoI with pixel-wise and further align the RoI to segment the boundaries of each RoI. To retain the spatial information, used 3×3 convolutional channel with one pixel. The proposed enhanced CNN has five layers with max-pooling in each layer and softmax as the activation function. The limitation is images used in this study are taken in a conditioned environment.

Zhou et al. (2021a) proposed a Restricted Residual Dense Network (RRDN) to detect ten tomato diseases. DenseNet cumulative all the previous and subsequent layers, enhancing the network's capability with fewer parameters. The residual network resolves the gradient vanishing issue by skip connections; the Residual Dense (ResDense) network takes the benefits of both the Residual and DenseNet; ResDense generously extracts deeper features. Further, Optimizing ResDense Network, then trained to detect the tomato diseases with 95% detection accuracy. This work could be further improved with IoT and sensors to collect other information and detect leaf diseases with nutritional deficiencies.

Chug et al. (2022) proposed a hybrid approach to detecting tomato plant diseases.

Preprocess the images by employing a bilateral filtering approach to remove the noise and background, then segment the plant leaf by employing K-means clustering. Extract the features using EfficientNetB0-B7 and then feed these extracted features in different combinations to classify using Machine Learning classifiers such as Linear Regression, KNN, Random Forest (RF), AdaBoost (ADB), and SGB. EfficientNetb3 with ADB and Stochastic Gradient Boosting (SGB) attained the best accuracy of 100%. The limitation is that the dataset is small, with 155 images for two classes.

Ashwinkumar et al. (2022) proposed a tomato plant disease detection approach by employing Optimal MobileNet-based CNN (OMCNN). Preprocess the tomato leaf images with bilateral filtering and a Kapur threshold-based approach to segment the images to get the RoI. MobileNet is used for feature extraction to enhance the model's performance, which optimizes the hyperparameters by employing emperor-penguin-optimizer, another extreme learning machine used as a classifier, and attained the best precision of 0.985. The dataset used in this study is imbalanced, and in class, it has just 60 images; this makes the DLM poorly trained.

Few works considered Unmanned Ariel Vehicles (UAV) images for the plant leaf disease classification. Research (Kerkech et al. (2018)) proposed a model for grapevine disease classification in vineyards using UAV images with CNN and color information. In the proposed method, the authors have used the LeNet-5 with different color spaces. Vegetation Indices (VI) with the different fusion of color spaces and different color spaces with VI were used to obtain the related fusion of feature spaces that steer to improve the accuracy. The resulting model was used for the detection and classification of the diseased area. This work used less labelled dataset; this work also could be enhanced by a UAV multispectral image dataset.

Tea Leaf Disease Detection: UAV images were used for tea leaf disease detection by using a low shot learning method proposed by a few researchers (Hu et al. (2019)), to generate the different types of disease spot images, Conditional Deep Convolutional Generative Adversarial Network (CDCGAN) with conditional labels were used, and VGG-16 to detect the disease. The limitation is, Tea is a plantation crop, where a broad area covers the plantation; a UAV-based plant disease detection approach is essential; this could be done with more images to train the model.

Wheat Leaf Disease Detection: A Deep Learning-based infield automatic wheat disease detection approach was proposed by a study (Lu et al. (2017)). Deep and multiple instances learning-based supervised learning framework for wheat disease detection was proposed, which can work in complex background infield wheat images.

Another study (Picon et al. (2019)) proposed Residual Neural Network-based DCNN for wheat disease detection by developing a mobile application. It is able to produce classification results within 5 seconds, and it also stores the captured image in the server for posterior statistical analysis.

Multi-plant Leaf Disease Detection: Most of the existing work considered datasets with multiple plants for their research. In the following, discuss the works which considered multiple plant leaf images. CNN-based plant leaves disease detection method is proposed by some researchers. (Ferentinos (2018)). Further, the author has used five different DLMs in their work, such as AlexNet, AlexNetOWTBn, GoolgeNet, OverFeat, and VGG. The AlexNetOWTBn and VGG models achieved the highest detection accuracy; additionally, the authors experimented with original images by training AlexNetOWTBn and VGG models with more epochs.

Jiang et al. (2021a) proposed a multi-task method by sharing the multiple related parameters for each task to detect the paddy leaf disease detection using VGG-16 and attained 97.99% and 97.56% detection accuracy for paddy and wheat, respectively. This approach used only 360 paddy leaf images with three classes and 240 wheat leaf images with two classes. The dataset used in the study was very small, not considering all the diseases of the plants.

A DCNN-based plant leaves detection approach was proposed by some studies (G & Arun (2019)), in which the image transformation is applied to reduce the overfitting. A total of 55,448 images were created for training, and 61,486 images were created using data augmentation. The DCNN was trained separately using two datasets; one with augmentation and another one without augmentation. The images used in this study are taken in a conditioned environment; complex images could be used.

Multi-functional-based plant leaves disease detection approach called Plant Disease Diagnosis, and Severity Estimation Network (PD2SE-Net) was proposed (Liang et al. (2019)). It uses the union of the residual network and Shuffle-NetV2. The ShuffleNet-V2 is used to minimize the computational convolution. It engages the shuffle-V2 channel of the residual block so as to split the channels; this, in turn, recognizes the plant species, classifies the diseases, and estimates the severity of the disease. This work could be further explored with an application on smart devices.

One study proposed (Kamal et al. (2019)) a Depth-wise Separable Convolutional (DSC) approach to classify various plant leaf diseases with Reduced-MobileNet, Modified MiobileNet, and MobileNet DLMs. The study used DSC, which employs each kernel filter channel with only one input channel, and the resulting output channels were mixed by a PointWise Convolution (PWC). This approach further explored the diseases that affected the other parts of the plants, such as the stem, roots, etc.

Delnevo et al. (2021) proposed an IoT-based Deep Learning approach to detect various plant diseases. The authors employed IoT devices and mobiles to gather information such as soil moisture, temperature, and solar rays and also employed crowd gathering to collect the images and also the labeling, where the user takes the leaf image and uploads it to the cloud-based system by using the app which is developed for disease detection. The authors collected six user feedback by setting a few questionnaires. For classification purposes, DenseNet121, MobilaNet, MobileNetV2, and NasNetMobile utilized and attained 94.58% detection accuracy with MobileNetV2. This work used limited dataset used for training; not gathering the large dataset by using IoT could be addressed. Further, this approach used the pre-trained models, and this could be analyzed with various different hyperparameters.

Chouhan et al. (2021a) proposed an IoT- based Deep Learning approach to detect plant diseases. Scale-Invariant Feature Transform (SIFT) was employed to extract the features from the plant leaf images and then employed Fuzzy-Based Function Network (FBFN) with an if-then rule to detect the diseases. This work could consider the hyperspectral images using sensors, and also this work could make it more realistic using IoT to collect the images using IoT devices.

Few works considered the multi-plant leaf such that those plants belong to the same family and cultivate in the same season. These crops are affected by similar kinds of diseases. Accordingly, a study proposed (Khamparia et al. (2020)) early stages of seasonal Kharif crops disease detection by using CNN. The authors used CNN models to extract the features and then detect the diseases based on CNN and Artificial Neural Network and also applied other Machine Learning classifiers such as KNN and SVM. The number of images used for this study was very less, with 600 images, which is very small to train a DLM. This work can be further enhanced by employing a deep stack approach and also with belief-network.

Aravind & Raja (2020) experimented with various DLMs such as AlexNet, VGG-

16, VGG-19, GoogLeNet, ResNet-101, and DenseNet-201 on diverse plant datasets such as eggplant, beans, ladies finger, and citrus and attained the best detection accuracy of 90% with VGG-16 and all other models performance was low in comparison to VGG-16. The authors observed that the cercospora of brinjal and canker of citrus were miss-classified since these two diseases share similar visible features. The limitation is that the model was trained with just 20 epochs with a limited dataset; the experimental result further improved with a larger dataset and more epochs.

Barburiceanu et al. (2021) proposed a CNN with a Machine Learning classifier to detect the various plant diseases on the Outex-TC-00013 and PV datasets. Extracted the texture features with the AlexNet, VGGNet, and ResNet by removing the FC layer; further employed SVM with Radial Basis Function (RBF) as a classifier for disease detection using the weights generated by the models. The authors have concluded that this approach performed better than TL with AlexNet, VGGNet, and ResNet. The PlantVillage dataset is used for image segmentation, which does not have a complex background; this work can extend with complex background images with more data.

Zhao et al. (2021) proposed plant disease detection by employing Double GAN. Wasserstein GAN was used to obtain the 64×64 images output of this fed to super-resolution GAN to obtain 256×256 images; both are connected serially. Double GAN ensures that the obtained images have more clarity and more information than images generated by Deep convolution GAN. For classification, various DLMs were used. DenseNet121 performed better than other DLMs by attaining average detection accuracy of 99.7%. The authors have trained the model with laboratory-conditioned images, and the model is not tested with the images taken in farm due to the unavailability of the real scenario images.

Hassan & Maji (2022) proposed a CNN model modifying the InceptionNet by employing DSC and PWC. DSC is for filtering, and PWC is to combines the output generated by DSC. Further, DSC independently maps the channel and spatial dimension to improve the efficacy and then de-couples channel and spatial correlation. The proposed approach outperformed the other DLMs. The limitation, this work could be explored with state-of-the-art unsupervised clustering techniques.

Liu et al. (2021) proposed a three-staged DLM to detect 271 different plant diseases with 220,592 images. The first stage is Lesion re-weighting based on clustering, which considers all the patches of the image and assigns the unique weight for each patch while performing clustering. During the second stage, assigns the weight to every loss. In the third stage, the features extracted from the second stage and corresponding weights from the first stage are combined to get the weighted features by employing LSTM. The model is slow due to the clustering process prior to training; this approach further extends to analyze the random patches present in the images and handles an unbalanced dataset.

Zhai et al. (2022) proposed an enhanced DLM; it has two stages. In the first stage, train the network with the original noisy labels; in the second stage, integrate the meta-learning with improved components to enhance the lenity of noise with various simulated small subsets with spurious labels for training. Further, apply the high penance for higher loss and less for less loss to avoid overfitting. This approach provides a robust model for the large dataset with noisy labels. The limitation is that this work used the PV dataset with various noise; this could be done with noisy images to make it more realistic.

Paymode & Malode (2022) proposed a multi-crop plant disease detection approach by employing VGG16 on grape and tomato plant datasets. To avoid overfitting, utilized data augmentation techniques such as GAN, and Neural Style transfer. Color, texture, and shape information is extracted to detect diseases. VGG16 outperforms the other models by attaining 98.40% for the grape and 95.71% for the Tomato plant dataset. This work could be further extends with complex images and an extended dataset which minimizes the use of AT.

Gajjar et al. (2021) proposed a hybrid model, the first model classifies the leaf, and the second is to detect the disease in the leaf. PV was employed to train leaf classification and disease detection models. Data AT, such as flip, brightness, and cropping, were used to enhance the size of the dataset. For classification single shot detector with MobileNet was employed, and for disease detection CNN model was used. This work could be extended with UAV images, and adapting IoT in the work results in a more robust system. This work also could use image segmentation approaches to remove the background noise.

Sai Reddy & Neeraja (2022) proposed a plant disease detection by employing the DenseNet model; it has three stages; in the first stage, train the model with apple, grape, potato, and strawberry datasets and attain 100% training accuracy. In the second stage, detect the disease on the testing dataset. The third stage suggests an appropriate precautionary measure for the detected diseases. Experimented with different ratios of train and test split such as 30:70, 40:60, 50:50, 60:40, and 70:30 for training and testing, respectively, and attained 100% training accuracy for 50:50, 60:40, and 70:30 for train and test ratio. The limitation is that the images used in this study are taken in controlled conditions, and the complex images could be used for training with state-of-the-art Deep Learning approaches.

Alguliyev et al. (2021) proposed a hybrid model; in the first stage, extracted the features using the CNN model with RMSprop as an optimizer, batch size 100, learning rate 0.001, and 10 epochs. The second stage is for classification by employing GRU, which has two layers, first with ten neurons second with ten neurons, followed by a fully-connected layer with softmax as an activation function. The limitation is to test the model for only two classes; it could consider all the diseases of the cucumber plant.

Hua et al. (2022) proposed a hybrid model with a multi-feature fusion approach to detect plant diseases. The authors used the Gabor filter and color detection independently to extract the features and fuse the obtained features for decision-making and further employed R-CNN for the detection. To enhance the result, a membership classification retrieval system was used.

Cristin et al. (2020) proposed a hybrid model to detect various plant diseases. It has three stages. In the first stage, preprocess the images using fuzzy C-means to segment the lesions of the plant leaf, then in the second stage, extract the features using Information gain, entropy, and histogram-based gradient. The third stage is the training of the classification model; in this, Rider-Chokoo Search Algorithm (R-CSA) based Deep Belief Network (DBN) was trained and tested on different plant datasets. This work can be further extended to in-farm plant leaf images and an in-filed disease detection approach.

Kharif crops used in the experiments were soya, rice, and corn. In the experiments, image dimensions were reduced, minimizing the training time. The proposed model achieved better disease detection accuracy, faster generalization, and convergence potentiality than the other Machine Learning models.

Some studies proposed (Khamparia et al. (2019)) Deep Convolutional Encoder Network (DCEN) to detect the diseases of seasonal crops. In this work, researchers used the encoder to extract the useful features of 3 different plants and five kinds of diseases; this technique achieved 100% detection accuracy with 3×3 filter size. This model outperformed as compared to conventional techniques. Seasonal crops which were used in this work were potato, tomato, and maize.

Autoencoder convolution with regularizing the deep clusters was proposed by Gokulnath & Gandhi (2021); in this method, features extracted from a maize leaf image belong to a different category by employing the autoencoder with local preservation and regularization approach. It has two phases; in the first phase, it extracted the hierarchical features using the stacked convolutional layers and autoencoder, then applied the FC layers. In the second phase, regularize the network by employing the Kullback–Leibler divergence, which assists in forming the cluster by avoiding the outliers to form an independent cluster. This work uses the learning rate as 0.01, which lacks the tiny features, and the dataset used does not have complex factors.

Qi et al. (2022a) proposed enhanced YOLOv5 for classifying tomato plant diseases. To make the work more realistic, the authors collected the images with a complex background and applied Zero-mean Gaussian noise and AT, such as rotation and mirroring, to avoid overfitting. To enhance the detection accuracy, detect the objects in the images by applying a bounding box and extracting features by applying spatial and channel attention. The spatial attention mechanism is used to extract the local features, and channel attention is used to gain knowledge of features associated with each feature channel and attain the weights associated with them. The attention mechanism approach can further be optimized and segmented the images using state-of-the-art methods to minimize the missclassification.

Ji & Wu (2022) proposed pixel-level severity identification of black grape measles by employing DeepLabV3 for segmentation purposes and ResNet50 as a base model to extract the most prominent features. Encoder and decoder used in DeepLabV3, an encoder is used to extract the semantic information and minimize the feature maps and conceptual knowledge; the decoder is used to retrieve the spatial expertise with the help of keen edges of the object.This work could be further improved with complex and complex images and also can be further extended with UAVs.

Kendler et al. (2022) proposed a multi-patch approach to classify the barley, potato, and wheat diseases by employing various Deep Learning techniques. The authors collected images in farmland by considering illumination, intensity, different pixel, and geometric conditions. The multi-patch approach splits the image into numerous patches; this improves the dataset size by retaining the spatial information and makes the dataset more realistic to get a robust model with the generalized result. The multi-patch approach improves the classification result of the unseen images taken from other areas of the same crop. This work can be further extended with UAV and airborne images to make the system more realistic and can adopt for UAVs.

Transfer learning is a procedure that makes use of the knowledge acquired by the system for solving one problem with varied yet related problems. This enhances the results in prediction rate and shortens the training time. TL was also used in various plant disease detection systems, such as peach (Zhang et al. (2019)).

Fine-tuning the DLM such as Inception v4, VGG-16, ResNet, and DenseNet121 for plant disease detection was compared by one of the studies (Too et al. (2019)); in this study, the PV dataset was used with 80% and 20% train, test split, respectively. This is a comparative study of the performance of the pre-trained models. SGD optimizer was used with a learning rate set to 0.001. This study is a comparative study, performance of the models were not tested with external testing data. TL based on GoogleNet was used to detect different plant diseases (Barbedo (2019)). To overcome the lack of suitable datasets, the authors used only the independent lesions and spots for detection purposes instead of using the whole leaf image since every lesion has its feature. The mutability of the data was enhanced without the additional images using this technique. The authors used the patches to enhance the dataset and detect the diseases; this can be further extended with a larger dataset generated in farmland with whole leaf images.

TL with VGG-16 was used for feature extraction in Millet crop images, which was pre-trained from the ImageNet (Coulibaly et al. (2019)). It had three main steps, image acquisition, feature extraction using a pre-trained model, and disease classification. The employed method steers clear of overfitting by early stopping technique with less validation loss. This work could be further improved with the state of the Deep Learning approaches and algorithms for image segmentation and classification.

2.2 Single Network-based approaches with hyperspectral images

A study proposed (Park et al. (2018)) a DNN-based Marssonina blotch or blot disease detection approach for Apple Leaf hyper-Spectral (ALHS) images. Minimum redundancy and maximum relevance attribute selection approach were employed to extricate the pivotal bands from hyperspectral images of apple leaf. It demonstrated better detection accuracy than RGB images using DNN for disease classification. This work could analyze the other Dimensionality Reduction Techniques (DRT) available.

3D Deep Learning soybean plant disease classification by employing hyper-spectral stem images was proposed (Koushik et al. (2019)). 3D CNN was the potential for spatial-temporal features, and it considers spatial and spectral correlations concurrently. They also discovered the most tactful wavelengths that were used for categorizing in the near-infrared region. The authors broaden the conviction of saliency maps to visualize the pivotal spectral bands for classification. 3D CNN achieved 95.73% detection accuracy. This work could use state-of-the-art DRT to select the pivotal spectral bands.

Pérez Roncal et al. (2022) proposed an approach to classify the esca disease of grapes using hyperspectral images, and the authors collected 72 images infected by Esca with a 900-1700nm spectral range. Preprocess the data using mean centering, smoothing, multiplicative scatter correction, standard normal variate, first and second derivatives, and PCA for dimensionality reduction. PWC with Partial Least Squares Discriminant Analysis (PLS-DA) approach was employed; further, variable importance in projection and selectivity ratio was employed to enhance the selection of best wavelengths for classification of Esca. This work could be further extended to analyze the spectral data to handle the images with similar symptoms; some plant diseases might have similar symptoms further, and this can be extended to other parts of the plant.

Bagheri et al. (2018) proposed pear leaf disease classification using hyperspectral images with Soft Independent Modeling of Class Analogy (SIMCA). The authors have collected 106 hyperspectral images infected by fire blight with different bandwidth ranges and applied PCA for dimensionality reduction. Further, VI and Near-infrared imaging (NIR) were utilized for detection, which helps to observe the color changes in the leaf periodically. NIR is more reliable for recognizing early disease detection. The limitation is that the dataset used was very small, with 106 images, and could use state-of-the-art DRT to select the suitable bands for classification.

Furlanetto et al. (2021) proposed soybean disease detection by using hyperspectral images. The authors have collected various stages of the soybean rust disease images with a spectral range of 350 to 2500nm. To monitor the variance of the reflections, PCA was used. To avoid preprocessing the images, a plant probe device was used; the plant-probe device avoids illumination interference and collects the data with reflectance spectral without noises, scattering, or attenuation with the environment and device. The stepwise and DISCRIM model was utilized to select the useful bands; ultimately, 87 bands were selected. This work considers only one disease of the soybean; this could be extended to all the diseases of the soybean plant.

2.3 Hybrid Deep Learning approaches

Combining different network models for the classification of the plant diseases, such as combining the features learned by different models and sharing the weights of the different models, is called a dual network model.

An ensemble model by employing Kuan Filtered Hough Transformation (KFHT) with Re-weighted Linear Program Boost Classifier(RLPBC) for plant disease detection was proposed by Deepa & Nagarajan (2021). KFHT is a noise remover; noise associated with each pixel that diverges from the mean is removed and further enhances the caliber of the images. RLPBC act as a classifier on the PV dataset. RLPBC finds the weak learners, ensembles the results, and weights further re-weighted those weak learners for the corresponding training error.

A lightweight hybrid model with fewer parameters was proposed by modifying the Inception model; the authors achieved fewer parameters by replacing the sequential convolutional layers of the Inception model with DSC and PWC. BN has pertained to all the layers except the hybrid part of the model, which enhances the model's performance and concatenates the modified Inception model (Tuncer (2021)). This work could be further enhanced the with state of the art DLMs and compared them with other state-of-the-art models.

Nandhini et al. (2022) proposed Gated Recurrent CNN (GR-CNN), a combination of RNN and CNN, to detect banana diseases. Banana images are taken for the research and preprocessed the images by employing a gated recurrent unit. GRU is faster than LSTM since it has fewer tensors and three gates each for an update, reset, and memory. Update the gate to add or remove the information, and reset the gate to forget past knowledge. Batch re-normalization is employed in the first layer of the network instead of BN to perform shifting and scaling, and it uses the samples individually to normalize the data. This work can be further enhanced to tune the best hyperparameters and optimization algorithms to make the system more robust and lightweight.

Li et al. (2022a) proposed Fast-Wide and Deep Feature Extraction Block (FWD-Block) to enhance the dataset size with better quality, extracted depth, and global features by employing ResNet and Inception-V1, respectively, with DCGAN. To minimize the computational cost of the training model, DSC with discriminator and Selu activation was applied. DSC has two layers, first with a single channel filter and second with point by point convolutional layer; further, to improve the efficacy, DSC independently maps the channel and spatial dimension and then de-couples channel and spatial correlation; further, for classification, Both channel Residual Attention Network (B-RAN) was employed. The GAN approaches need some attention to find the appropriate degrees and styles to enhance the dataset such that the dataset minimizes redundant data.

Turkoglu et al. (2021) proposed an ensemble Deep Learning approach by employing six different DLMs to detect 15 classes of different plant diseases of Turkey-PlantDataset, which consists of 4447 images. Shape, texture, and color-based features are extracted and ensemble the AlexNet, GoogleNet, ResNet18, 50, 101, and DenseNet201, further applying the majority voting with SVM as a classifier to detect the diseases. This work can be further extended with a combination of other State of the art Deep Learning approaches, further analyzing and selecting the feature reduction approaches to choose the most prominent features.

.

Multi-network model for apple plant disease detection was proposed by Turkoglu et al. (2019), employed SVM, CNN, and LSTM model with Ensemble majority voting, and achieved an accuracy of 99.2%. This model outperforms the pre-trained models, such as AlexNet with RMSProp, GoogleNet with Adam, and DenseNet with RMSProp. The authors have used apple leaf images captured by considering various categories such as light illumination, size, area of the disease, and complex background.

Astani et al. (2022) employed an ensemble approach; the authors designed an approach that consists of four different approaches to preprocess the images. The first model employs K-Means clustering for segmentation, and the second model enhances the contrasts and extracts the features by utilizing the wrapper. The third model is for removing the shadow in the image and extracting the feature by utilizing SIFT. The fourth model is for the background removal task and extracting the features using GLCM and GWT techniques. This work is computationally costly, and the testing time is more as it takes 44 minutes. It can be examined with different classifiers by eliminating the identical process and also examining with an ensemble combination.

Sathiya et al. (2022) proposed a three-staged hybrid model for herb-plant disease detection. In the first stage, the multi-swarm coyote optimization technique was employed for segmentation purposes. Enhanced Chan-Vese snake optimization with a Gaussian kernel was employed in the second stage to extract the most dominant features and to reduce the dimensions of such features. Fitness-distance balance DNN was employed with normalized fitness, and distance value was utilized to classify the leaf and diseases. The size of the dataset used in the work is very small, with 70 images per class to train the DLM. The authors have used 700 images to train for ten classes. It can also be further improved by using hybrid approaches to enhance the performance.

Zhao et al. (2022) proposed Dual Transfer Learning (DTL) by employing Squeeze and Excitation Network (SENet) and ResNet for various plant disease classifications. SENet consists of three stages. In the first stage, it squeezes the data by applying $H \times W \times C$; in the second stage, it uses the unique weights of each channel and develops the dependencies with them, and in the third stage, it adds the new weights generated from the excitation by applying multiplication. The second approach is DTL, employing channel and spatial attention mechanisms to get the most efficient features with ResNet. This work can be further extended to train the model with complex images and then connect with IoT devices to detect in the real scenario.

Fan et al. (2022) proposed a CNN model to classify the apple and coffee plant diseases by extracting the semantic information. Extract the different features parallelly by independently employing Deep Learning and histogram of gradient techniques, fusing those features to extract the local spatial texture features. To minimize the illumination histogram of gradient, normalize the image by using Gamma correction; further, to extract the local features, it calculated the gradient and amplitude of each pixel in the cell and finally fused all the features generated. This work could be further extended with a larger dataset with complex images and can analyze the state-of-the-art augmentation approaches.

Chen et al. (2020a) proposed an approach to detect tomato plant diseases by employing Binary Wavelet Transform (BWT) with retinex to preserve the texture information and enhance the image. BWT is to splits the high and low-frequency features; first, it gathers the low and high-frequency features, and process the obtained features, then inversely transformed the elements to get the final components. Gaussian convolution is used as a denoiser. ABC approach is used to segment the tomato leaf, and the B-RAN model is applied to detect the diseases of the tomato leaf. The images used in this work are captured in a conditioned environment. This work can be further enhanced with state-of-the-art image segmentation and noise removal approaches.

Cucumber plant disease detection was proposed by Wang et al. (2021a), DeepLabV3+, and UNet were employed to separate the complex background and noise of the cucumber plant images and attained 92.85% detection accuracy. The authors used cucumber plant leaf images taken from farmland with complex backgrounds. The cucumber plant dataset has PM disease. Since this method uses two stages to segment the leaf image, this consumes considerable time; hence segmentation time needs to be reduced.

Khan et al. (2022) proposed a lightweight apple leaf disease detection model; this has two stages; in the first stage, the model classifies the image either as healthy or diseased; in the second stage, the classification of the diseases is developed. as part of the research, the authors collected 5201 apple leaf images with ten classes. For the classification, EfficientNet, FR-CNN, and YOLOv4 were used and attained an mAP of 42.5% using the YOLOv4 model. The authors have not used a background removal approach; the dataset is a class imbalance dataset, and some classes have very few images. The system is 2 stage approach; due to two different stages, the computational cost is more.

Zhao et al. (2022a) proposed a plant disease detection approach on corn, potato, and tomato by employing Convolutional Block Attention (CBA) technique with Residual Inception Network (RI-Net). RI-Net increases the computational complexity of the network by merging the residuals into the Inception of the CNN; this increases the number of parameters and enhances the localization of the disease spots by employing the CBA. CBA utilizes the channel and spatial attention approach to extract the most prominent features by considering global and local features. The authors have not used complex images to train the model.

Segmentation can also be used for agriculture-based applications such as harvesting, vegetable disease detection, and pest detection. Pérez-Borrero et al. (2020) proposed instance segmentation based on mask-R-CNN to harvest strawberry fruits by segmenting ripe fruits and background. Coconut tree leaf diseases and pests were recognized by employing watershed and KMC-based segmentation approaches and further applied various Deep Learning approaches for classification (Singh et al., 2021). The mask R-CNN is considerably a slow process; this would lag in farmland disease detection; this could be improved with state-of-the-art approaches.

A hybrid model with the Internet of Things (IoT) makes plant disease detection more efficient and robust. A hybrid plant disease classification approach by employing neural network model with IoT proposed by (Mishra et al., 2021) to detect the plant diseases. Images were collected using IoT, and sensors are complex, with more noise.

The model processed such images by such images by employing a median filter which extracts the leaf structure, and a segmentation approach, which segments the leaf shades on a pixel basis. The model extracted the features using pixel-level and segment-level feature extraction approaches. Finally, employed Ride-NN for classification. This work is to classify whether the leaf is diseased or healthy; this could be extended to find the diseases specific with the improved dataset.

Table 1 describes the contrast with the existing plant leaf disease detection approaches with the following parameters: DLM, dataset, plant name, accuracy, and image types such as RGB, Hyper-Spectral (HS), and Fold Cross-Validation (FCV) used. The selected parameters are the most commonly used parameters, which directed the way toward identifying some of the research gaps. Deep Learning is an emerging technique for plant leaf disease classification. The main limitation of Deep Learning is that it needs a large amount of input data to train the model.

2.4 Gaps in the litearture

- Plant leaf disease detection is essential where developing countries like India. Farmers can quickly get the diseases in plants, if any, in the farmland by capturing images of the plant leaves in the farmland itself; this helps the users to take precautions in the early stage and on time.

- DLMs require a vast amount of processing power since the vast amount of images is used to train the model. These DLMs can be deployed only on the server, and users can make use of this through the Internet. These DLMs cannot deploy on smartphones since the processing power and memory is very less in these systems. Lightweight DLMs are required such that it can function on smartphones.

- To enhance the prediction results of the disease, make use of the multi-network model. It can be used for feature extraction. In feature extraction, different models extract the features separately; it gives the most valuable features, and a multi-network model can be used for the final prediction of the result; again, each DLM predicts different outcomes. Only limited works concentrated on integrating multiple Deep Learning Networks.

- Various image types can be used in the DLMs for plant leaf disease detection, such as color, grayscale, white and black, and hyperspectral images. Most of the works considered color images. Hyperspectral images provide more information as compared to color images.

- Many existing models are designed for large-scale commercial farming, but there's a need for tailored solutions that can be applied to small-scale farming contexts prevalent in developing countries.

Table 2.1: Comparison of existing Plant Leaf Disease Detection Approaches

Sl.No	Authors	Learning Model	Dataset	Plant Name	Accuracy (%)
1	Amin et al. (2022)	Feature Fusion	PV	Multiple	98.56
2	Arun & Umamaheswari (2022)	PWC CNN	PV	Multiple	98.14
3	Astani et al. (2022)	Ensemble	PV	Tomato	95.98
4	Ashwinkumar et al. (2022)	OMNCNN	PDDB	Tomato	98.8p
5	Elfatimi et al. (2022a)	MobileNetV2	Beans	Beans	92
6	Hassan & Maji (2022)	CNN	PV Rice Cassava	Multiple	99.39 99.66 76.59
7	Ji & Wu (2022)	DeepLabV3+	PV	Grape	97.75
8	Joshi et al. (2022)	CNN	Self	Rice	93.25
9	Li et al. (2022a)	B-RAN	PV	Tomato	98.75
10	Mathew & Mahesh (2022)	YOLOv5	Self PV	Pepper	0.907 mAP
11	Nandhini et al. (2022)	GR-CNN	Banana	Banana	93.6
12	Patil & Kumar (2022)	CNN with IoT	Self	Rice	93.6
13	Paymode & Malode (2022)	VGG16	PV	Grape Tomato	98.40 95.71
14	Pérez Roncal et al. (2022)	PLS-DA	Self	Grape	97.17
15	Qi et al. (2022a)	SE-YOLOv5m	Self	Tomato	91.07
16	Kendler et al. (2022)	CNN	Self	Multiple	95.4
17	Khan et al. (2022)	YOLOv4	Self	Apple	88
18	Sai Reddy & Neeraja (2022)	DenseNet	PV	Multiple	97
19	Seetharaman & Mahendran (2022)	R-CNN	Self	Banana	98
20	Shantkumari & Uma (2022)	CNN KNN	PV	Grape	96.60 98.07
21	Syed-Ab-Rahman et al. (2022)	CNN	Citrus Kaggle	Multiple	94.37
22	Vadivel & Suguna (2022)	CNN	PV	Tomato	99.5
23	Zhai et al. (2022)	CNN	PV	Multiple	94.58
24	Abed et al. (2021)	UNet DenseNet121	Self	Beans	98.31
25	Alguliyev et al. (2021)	CNN GRU	PV	Multiple	91.19
26	Chouhan et al. (2021a)	IoT with FBFN	Self	Multiple	90.18

Table 2.1. Comparison of existing Plant Leaf Disease Detection Approaches contd.

Sl.No	Authors	Learning Model	Dataset	Plant Name	Accuracy (%)
27	Delnevo et al. (2021)	IoT with CNN	Self	Multiple	94.58
28	Gajjar et al. (2021)	Hybrid with CNN	PV	Multiple	96.68
29	Gokulnath & Gandhi (2021)	Autoencoder with CNN	PV	Maize	90.08
30	Haider et al. (2021)	DT and CNN	Self	Wheat	97.2
31	Liu et al. (2021)	SeNet154 ResNet152 LSTM	PDD-271	Multiple	90.01
32	Jiang et al. (2021a)	VGG-16	UCI	Rice	97.22
33	Mishra et al. (2021)	Ride-NN with IoT	PV	Multiple	91.56
34	Nagasubramanian et al. (2021)	CNN with IoT	Self	Sugar beet	80.1
35	Shin et al. (2021a)	ResNet50	Strawberry	Strawberry	95.59
36	Singh et al. (2021)	CNN	Coconut	Coconut	96.94
37	Tuncer (2021)	Inception	PV	Multiple	98.88
38	Turkoglu et al. (2021)	Ensemble	Turkey	Multiple	97.56
39	Wang et al. (2021a)	DeepLabV3+	Cucumber	Cucumber	92.85
40	Yu et al. (2021)	K-means Deep Learning	AI Challenger	Corn	93
41	Zhou et al. (2021a)	RRDN	AI Challenger	Tomato	95
42	Zinonos et al. (2021)	LoRa Deep Learning	PV and Self	Grape	99.77
43	Chen et al. (2020a)	ABC	Tomato	Tomato	89
44	Zhang et al. (2020a)	EfficientNet	Cucumber	Cucumber	96
45	Andrushia & Patricia (2020)	ABC	Grape	Grape	93.01
46	Chen et al. (2020)	VGGNet Inception	Rice Maize	Rice Maize	92
47	Cristin et al. (2020)	R-CSA	PV	Multiple	87.7
48	Esgario et al. (2020)	ResNet 50	Coffee	Arabica Coffee	97
49	Ji et al. (2020)	United Model	PV	Grape	98.57
50	S & Vydeki (2020)	DNN with Jaya algorithm	Paddy Leaf	Paddy	98.90

Table 2.1. Comparison of existing Plant Leaf Disease Detection Approaches contd.

Sl.No	Authors	Learning Model	Dataset	Plant Name	Accuracy (%)
51	Khamparia et al. (2020)	DCNN	Kharif Crops	Multiple	93.70
52	Mia et al. (2020)	NN	Self	Multiple	80
53	Aravind & Raja (2020)	TL	Self	Multiple	97.3
54	Zhong & Zhao (2020)	DenseNet-121	AI challenger Global AI Contest	Apple	93.71
55	Coulibaly et al. (2019)	VGG-16	Self	Millet	95.00
56	Cruz et al. (2019)	ResNet-50 ResNet-101	Grapevine Yellows PV	Grape	99.33
57	Liang et al. (2019)	PD2SE-Net	AI challenger Global AI Contest	Multiple	98.00
58	M & Adem (2019)	FR-CNN	Sugar Beat	Sugar beat	95.48
59	Pourazar et al. (2019)	RF	Self	Citrus	95.58
60	Hu et al. (2019)	C-DCGAN and VGG-16	Tea Leaf	Tea	90.00
61	Wu et al. (2019)	ResNet	Self	Soybean	94.29
62	Koushik et al. (2019)	3DDCNN	Soybean Stem	Soybean	95.73
63	Ahila et al. (2019)	LeNet	PV	Maize	97.89
64	G & Arun (2019)	DCNN	PV	Multiple	96.46

- Capturing the images in farmland are more prone to noise. Training the model and classifying the plant leaf diseases on noisy input images is challenging.
- Research should focus on optimizing models to run on resource-constrained devices like smartphones or edge devices to facilitate on-site disease detection.
- Precaution measures or prescriptions for plant diseases are essential, which can be suggested with the classification result.

2.5 Problem Statement and Objectives

2.5.1 Problem Statement

Plant leaf disease detection in real-time is highly essential for farmers to take appropriate action in real-time. **Aim of this research work is to propose a plant leaf disease detection approach**

2.5.2 Research Objectives

- Propose an enhanced / modified Deep Learning-based plant leaf disease detection approach.
- Experimental study on existing noise removal techniques and use best noise removal techniques with proposed approach to measure the detection ability.
- Evaluate the proposed approach by considering two/three plant leaf image dataset. Use performance metrics such as Accuracy, Precision, Recall, F-Score, and detection time to evaluate the performance of the proposed approach.

In the subsequent chapters, this study tries to solve the research objectives. It starts with an empirical study on plant disease detection using CNN, then proceeds with an ensemble-based approach, followed by an adaptive attention mechanism, background removal, and the EfficientNet method.

Chapter 3

Plant Disease Detection using Deep Learning Model

The performance of the deep learning model varies with different datasets with different sets of hyperparameters. There is no standard way or algorithm to fix the number of layers and neurons or which optimizer is influential for a specific dataset. Tuning the hyperparameters results in finding the prominent set of hyperparameters to build deep learning for a particular dataset. In the following sections, plant disease detection based on Multi Convolutional Layer-based Convolutional Neural Network is explained, and further, Ensemble DL-based plant leaf disease detection approach is proposed.

3.1 Multi Convolutional Layer-based Convolutional Neural Network

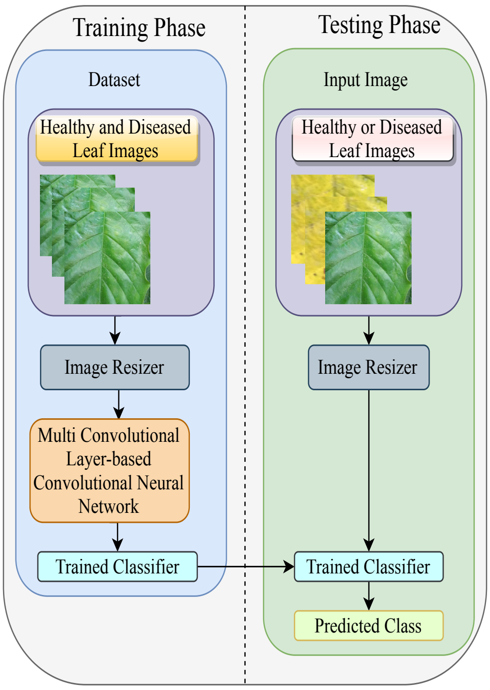


Figure 3.1: Multi Convolutional Layer-based Convolutional Neural Network Classifier

The Multi Convolutional Layer-based Convolutional Neural Network (MCLCNN) classifier used in this work is shown in Figure 3.1. It consists of training and testing phases. The training phase is used to train the MCLCNN. This involves Dataset, Image Resizer, Multi Convolutional Layer-based Convolutional Neural Network, and Trained Classifier. The trained MCLCNN classifier is used in the testing phase to measure its classification ability.

The classifier trained with images of smaller size takes less time to get trained.

However, the classifier is unable to learn several significant features than the one trained with images of larger size. Thus, the MCLCNN is trained with images of size 224×224. The MCLCNN used in this study demands that all input images be of the same size. However, generally raw input images are of distinct sizes. To convert raw images of different sizes into a pre-defined size 224×224, Image Resizer is used, and it is implemented using the Keras package called keras.preprocessing.

There are several hyperparameters available such as the epoch, batch size, dropout, learning rate, and various activation functions such as Binary step function, Linear function, Sigmoid, Tanh, ReLU, Leaky ReLU, and Softmax. These hyperparameters are used to solve complex image classification problems. Epochs are used to train the Deep Learning-based classifiers with the training dataset. Batch size refers to each pass using the number of training examples; batch size has three options: batch mode, mini-batch mode, and stochastic batch mode. Dropout is a regularization technique wherein, during the training phase, the randomly selected neurons are dropped or ignored. Further, this process enables the contribution of these dropped neurons to be removed in the interim during the forward pass; consequently, the weights don't get updated during the backward pass. The use of dropout in the network enables better generalization and avoidance of the overfitting problem. The learning rate tells the amount of weight updated during the training phase and is one of the pivotal hyperparameters. If the learning rate is high, the network converges quickly in local optima; however, the process may get stuck for a low learning rate. The activation function decides whether the neuron should be activated, i.e., the intelligence received by the neuron is relevant to the given task or ignored.

Figure 3.2 shows the internal components of the MCLCNN used in this work. It consists of four convolutional layers with ReLU as an activation function. The first convolutional layer is set with 32 kernels, followed by a batch of normalization layers. Batch normalization is used to normalize the output of the previous layer. The second convolutional layer is set with 32 kernels, followed by the batch normalization layer, max pooling, and dropout layer. The Max pooling layer takes the maximum value of each kernel. A dropout layer is used to prevent overfitting, which ignores the randomly selected neurons. The third convolutional layer is set with 64 kernels, followed by the batch normalization layer and dropout layer. The fourth convolutional layer is set

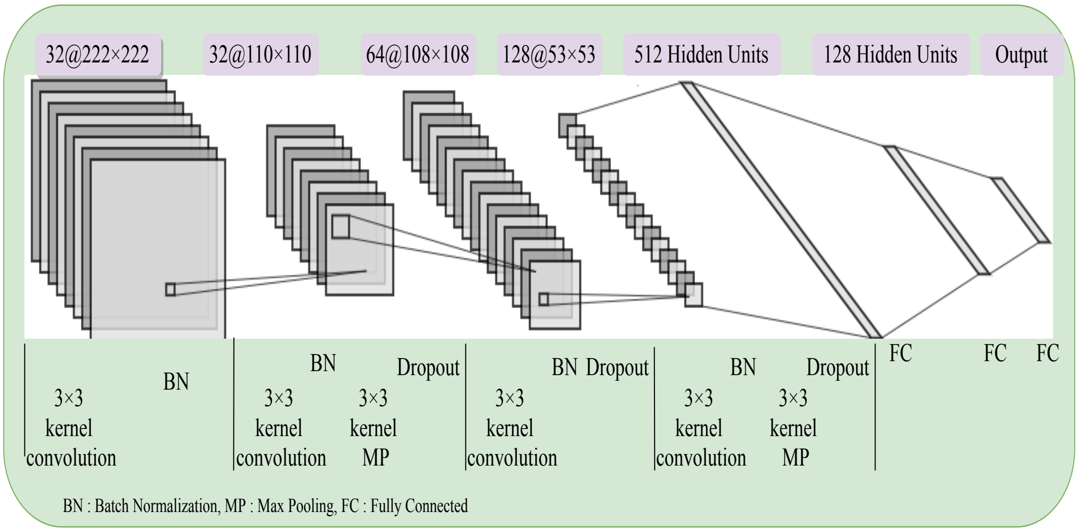


Figure 3.2: Components of Multi Convolutional Layer-based Convolutional Neural Network

with 128 kernels, followed by batch normalization, max pooling, and dropout layer. The MCLCNN has three fully connected layers; firstly, two fully connected layers use ReLU as an activation function with 512 and 128 neurons, respectively. The last fully connected layer acts as an output layer with a Softmax activation function.

3.1.1 Experimental Setup

All experiments were implemented using the Python programming language and executed on an NVIDIA DGX Station server with 4X Tesla V100 and 500 TFLOPS.

3.1.2 Dataset Description

In this empirical work, three different plant leaves datasets of PlantVillage dataset (Hughes & Salathé, 2015) were used, and each dataset consisted of two types of images: as being diseased and healthy. PlantVillage dataset consisted of 54284 leaf images, with 14 distinct plants and 26 distinct diseases. All of them were RGB images of size 256×256. In this experimental work, three distinct datasets of a total of 6,697 images were used, and their details are mentioned in Table 3.1.

Table 3.1: Dataset description

Sl.No	Plant Name	Class	Number of Images	Total Number of Images
1	Peach	Bacterial spot Healthy	2,297 360	2,657
2	Pepper	Bacterial spot Healthy	997 1,478	2,475
3	Strawberry	Healthy Leaf scroch	456 1,109	1,565

3.1.3 Experimental Results

All the original images of size 256×256 were resized into 224×224 using Image Resizer. Each dataset was split into a training dataset and a testing dataset with proportions of 70% and 30%, respectively, and the same proportion was maintained to conduct all experiments. Each dataset was used individually to train and test the performance of the MCLCNN classifier.

A set of experiments was conducted with three different datasets: Peach, Pepper, and Strawberry. For each of these datasets, experiments were carried out for different epochs, such as 50, 75, 100, 125, and 150, with a kernel size of 3×3.

The performance metrics used in this empirical study are Accuracy, Precision, Recall, and F1-Score (Brahimi et al. (2017), Ji et al. (2020), G & Arun (2019), Wu et al. (2019), Ma et al. (2018a)). Accuracy is defined as the number of correctly classified images to the total number of images used.

Classification Accuracy, Precision, Recall, and F1-Score achieved by the MCLCNN classifier for the first set of experiments shown in Table 3.2. The MCLCNN classifier achieved a minimum accuracy of 87.47% for the Peach dataset with 50 epochs. The experiments demonstrated that accuracy also slightly increased as the number of epochs increased. Accordingly, the maximum accuracy attained was 99.25% with 150 epochs. The Pepper plant leaf dataset achieved a minimum of 94.89% and a maximum of 98.38% accuracy, respectively. The Strawberry plant leaf dataset achieved a minimum of 94.04% and a maximum of 98.09% accuracy, respectively. The obtained experimental results show that the MCLCNN classifier performed well, and accuracy did not fall below the minimum accuracy achieved in the experiments, irrespective of the number of epochs. MCLCNN performed well by attaining the highest accuracy of 99.25% for the Peach dataset with the 3×3 kernel size.

3.2 Ensemble Deep Learning-based Approach

Plant diseases are the most crucial factors that reduce crop yield. It is essential to recognize plant diseases in the early stages with a cost-effective approach. None of the works considered multi-models for plant disease detection to the best of our perception. An Ensemble DL-based plant leaf disease detection technique is proposed.

Table 3.2: Performance evaluation of the MCLCNN for three different dataset with 3×3 kernel size

Sl.No.	Performance Metrics	# Epochs	Dataset Name		
			Peach	Pepper	Strawberry
1	Accuracy	50	87.47	95.29	97.66
		75	90.60	94.89	95.32
		100	98.87	95.15	94.04
		125	94.24	97.44	98.09
		150	99.25	98.38	97.45
2	Precision	50	0.77	0.96	0.98
		75	0.91	0.95	0.96
		100	0.99	0.95	0.96
		125	0.96	0.98	0.98
		150	0.99	0.98	0.98
3	Recall	50	0.87	0.95	0.98
		75	0.91	0.95	0.95
		100	0.99	0.95	0.94
		125	0.94	0.97	0.98
		150	0.99	0.98	0.97
4	F1-Score	50	0.82	0.95	0.98
		75	0.91	0.95	0.95
		100	0.99	0.95	0.94
		125	0.95	0.97	0.98
		150	0.99	0.98	0.97

Figure 3.3 describes the generic view of the proposed Ensemble DL-based plant leaf disease detection approach. This has many advantages, such as minimizing the spread in a prediction model's average skill and increasing the average classification performance of every contributing model, minimizing the classification errors made by the individual models, and enhancing the performance of the Ensemble model.

The proposed approach consists of two stages: the training and testing stages. The input images of different sizes cause problems extracting the image's relevant features by the DL-based Models (DLM); some require a similar input size for all input images. Hence, the image resizer is used in the proposed work.

Each DLM is trained separately during the training phase with labeled plant leaf images (training dataset). Each DLM learns the features automatically from the input images, and each DLM hyper-parameters are set to meet the model's expected robustness. Each trained DLM produces the classification output separately for each input image, either as a healthy or an unhealthy. The final classification result is based on

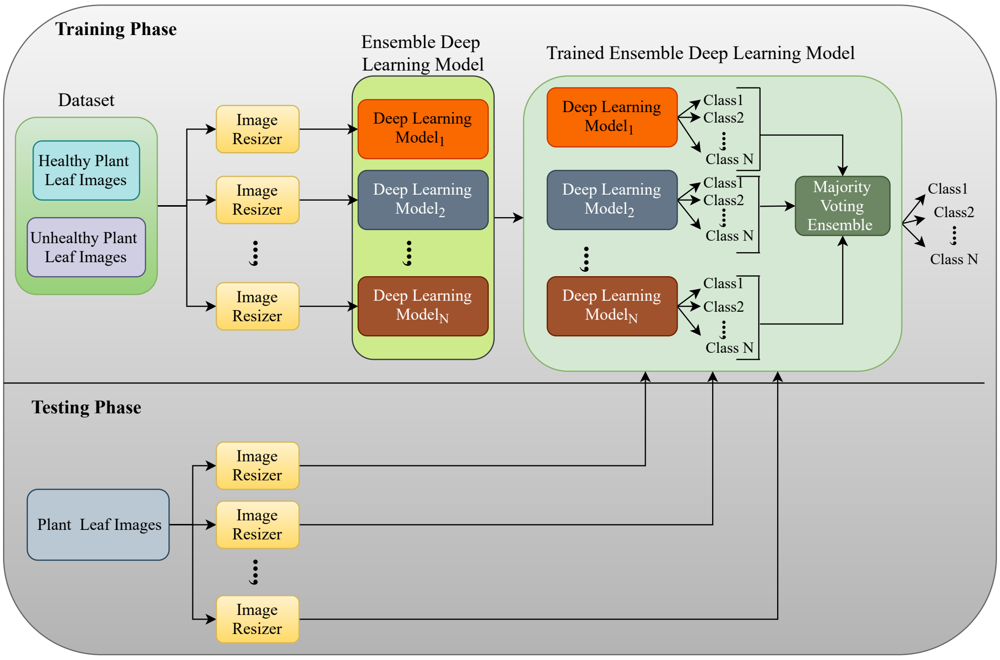


Figure 3.3: Generic view of the Proposed Ensemble Deep Learning-based Approach

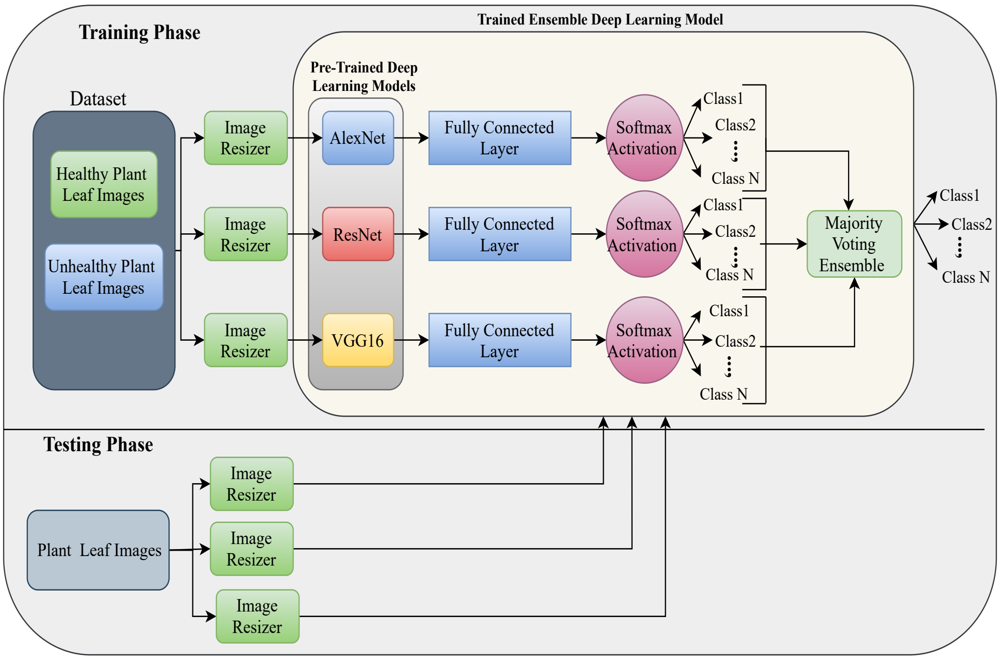


Figure 3.4: Proposed Ensemble Deep Learning-based Approach

the outcome of all DLMs, such as DLM_1, DLM_2, ..., DLM_N. To infer the final result, used a majority voting ensemble, which computes the output of each DLM, and the final prediction outcome is based on the majority of the outcomes of all the DLMs as per Equation 3.1.

$$\text{Output} = \text{Mode}(\chi_A(\text{C}_j(\text{Z}) = \text{i})) \tag{3.1}$$

The proposed approach is implemented as shown in Figure 3.4, in that AlexNet (Krizhevsky et al., 2012), ResNet50 (He et al., 2016), and VGG16 (Simonyan & Zisserman, 2014) pre-trained model weights are used and establish the new network by modifying the final layers of the model by stacking the fully connected layer on top of the pre-trained model with the softmax activation function, and batch size was set to 32. Further, finetuning each pre-trained model with the dataset Z and corresponding labels K with the loss function, which is expressed in Equation 3.2.

$$\text{L(W)} = -\frac{1}{m}\sum_{Z_i=1}^{m}\sum_{k=1}^{K}[\text{Z}_{ik} logP(x_i = k) + (1 - \text{Y}_{ik}) log(1 - P(\text{Z}_i = k))] \tag{3.2}$$

where W indicates weight matrix, m represents training instants, K is class labels, and P is the predicted probability.

All these models were trained and tested independently. Eight different datasets were used, Cardamom, Maize, Grape, Potato, and Tomato datasets were used as multiclass datasets, and Cherry, Pepper, and Strawberry datasets were used as binary class datasets. All the datasets were taken from the PlantVillage dataset (Mohanty et al., 2016a). All the images are colored (RGB) of size 256×256; images were given to the model as per the model's requirement, and each model expects the input in a different dimension.

3.2.1 Effects of Base Models:

The proposed study considers AlexNet, VGG16, and RestNet50; since each model extracts the features differently, each model has a unique technique to extract the features and classify the image. **AlexNet** utilizes the local response normalization by normalizing the local pixel amplifying the stimulated neuron, and it also addresses the outfitting issues by employing dropout layers. **VGG16** replaces the comprehensive kernel filters with collective 3×3 kernel filters; this enhances the model's depth, which encourages extracting the complex features. **ResNet50** resolves the vanishing gradient issue, and ResNet50 extracts the features once, and it does not try to extract the features again; instead, it tries to extract newer features. The Ensemble DL approach utilizes all three models' unique features and enhances the classification rate, and minimizes the miss-classification rate.

3.2.2 Experimental Setup

NVIDIA–DGX-Station server with 4X–Tesla–V100 and 500–TFLOPS was used to execute all the experiments and train the models, and Python version 3.7.0 was used for implementation. Input image size for AlexNet architecture was $227 \times 227 \times 3$; for ResNet50 and VGG16 architecture, the input image size was $224 \times 224 \times 3$. 90% of the dataset was adopted for training, and 10% of the dataset was utilized for external testing. All the experiments are trained with 100 epochs.

3.2.3 Dataset Description

PlantVillage dataset (Mohanty et al., 2016a) is a widely used dataset. It has 54,284 images of 38 different classes of several plant leaves. In this study, 3467 instances of the Maize plant dataset were utilized for training, and 385 instances were used for testing; 4018 and 446 instances of the Grape plant dataset were utilized for training and external testing, respectively. 1937 and 215 instances of the Potato plant dataset were utilized for training and external testing, respectively, and 1715 and 191 instances of the Cherry plant dataset were utilized for training and external testing, respectively. 1970 and 219 instances of the Pepper plant dataset were utilized for training and external testing, respectively, and 1408 and 157 instances of the Strawberry plant dataset were utilized for training and external testing, respectively. 16344 and 1816 instances of the Tomato plant dataset were utilized for training and external testing, respectively.

The cardamom plant dataset is used; it has a total of 1724 images. 1552 and 172 instances of the Cardamom plant dataset were utilized for training and external testing, respectively. Table 3.3 explains the dataset used in this experiment.

Table 3.3: Dataset used in Ensemble DL-based model

Sl.No	Plant	Class	Number of Images
PlantVillage Dataset (Mohanty et al., 2016a)			
1	Cherry	Healthy	854
		Powdery-Mildew	1052
2	Grape	Black Measles	1383
		Black Spot	1180
		Healthy	423
		Spot	1478
3	Maize	Blight	985
		Healthy	1162
		Rust	1192
		Spot	513
4	Pepper	Healthy	997
		Rust	1192
5	Potato	Early Blight	1000
		Healthy	152
		Late Blight	1000
6	Strawberry	Healthy	456
		Scorch	1109
7	Tomato	Bacterial Spot (BS)	2127
		Early Blight (EB)	1000
		Healthy (H)	1591
		Leaf Blight (LB)	1909
		Leaf Mold (LM)	952
		Mosaic Virus (MV)	373
		Septoria Leaf Spot (SP)	1771
		Target Spot (TS)	1404
		Two Spotted Spider Mite (SM)	1676
		Yellow Leaf Curl Virus (CV)	5357
8	Cardamom	Colletotrichum Blight	280
		Healthy	781
		Phyllosticta Leaf Spot	663

3.2.4 Results and Analysis

In this work, Accuracy, F1-Score, Precision, and Recall were used as performance evaluation metrics which are written from equation 3.3 to equation 3.6 respectively (Chen et al., 2021a); (Abdu et al., 2020).

$$\text{Accuracy} = \frac{\text{TP + TN}}{\text{FP + FN + TP + TN}} \times 100 \tag{3.3}$$

F1-Score is the harmonic mean of Precision and Recall, which provides information on the stability between Precision and Recall.

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.4}$$

Precision is the proportion of positive predictions that are true.

$$\text{Precision} = \frac{\text{TP}}{\text{FP + TP}} \tag{3.5}$$

Recall is the proportion of literal positive which was predicted correctly.

$$\text{Recall} = \frac{\text{TP}}{\text{TP + FN}} \tag{3.6}$$

TP: True Positive; TN: True Negative; FP: False Positive; FN: False Negative;

Table 3.4 describes the performance attained by the proposed model for the multi-class dataset. The proposed model attained the highest accuracy, 97.75% for the Maize dataset, 99.25% for the Grape dataset, and 99.53% for the Potato dataset. Table 3.4 also describes the performance achieved by the proposed approach for the binary class dataset—99.59% accuracy for the Pepper dataset and 100% accuracy for the Cherry and Strawberry datasets.

Table 3.4 also describes the performance achieved by the proposed model for the Cardamom plant dataset; it attained a detection accuracy of 99.41%.

By utilizing the unique benefits of AlexNet, VGG16, and ResNet50, experimental results showed by the proposed Ensemble approach infer that the miss classification rate is significantly less with different datasets.

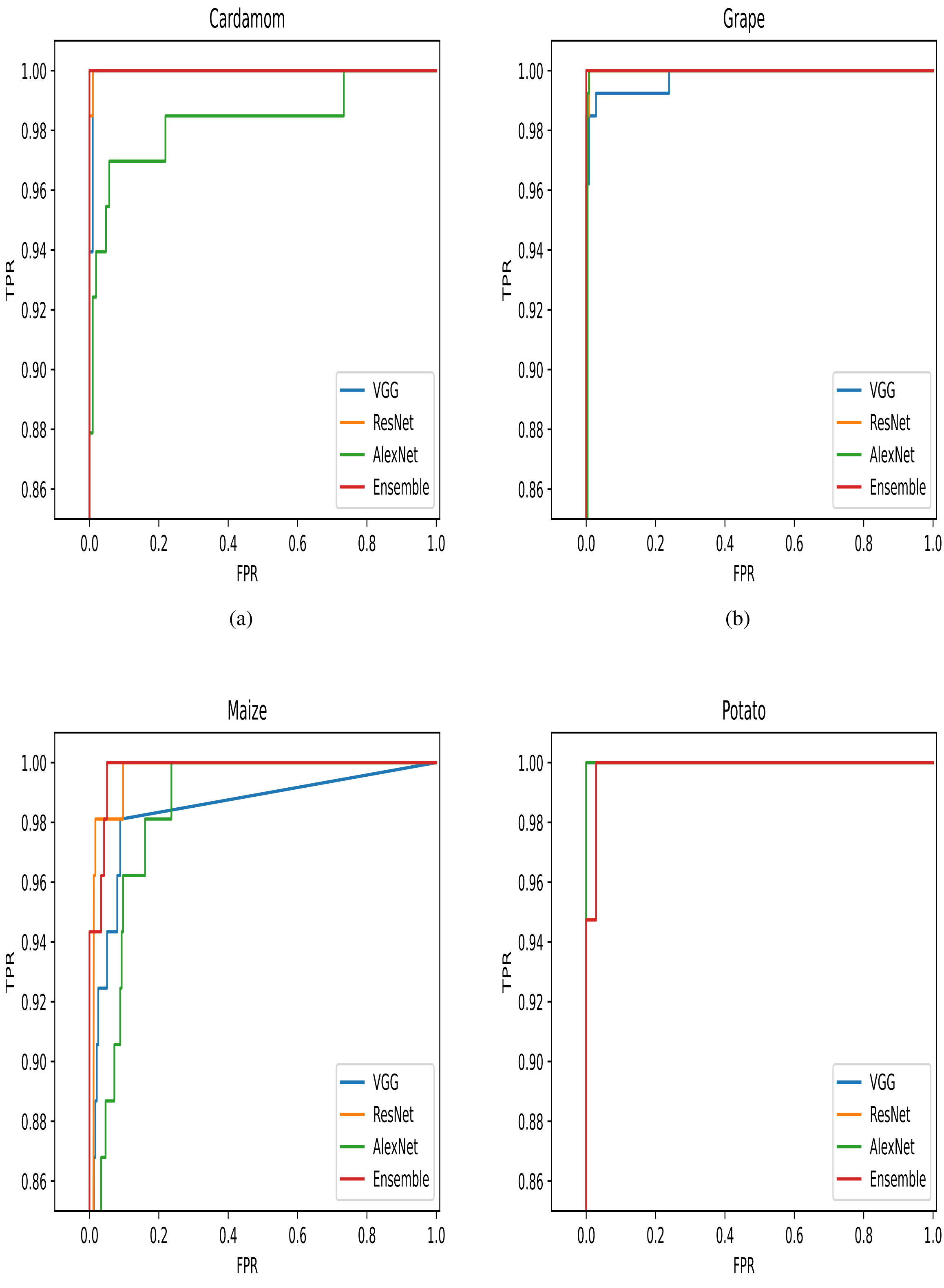
The Receiver Operating Characteristic (ROC) is a likelihood curve that evaluates classification on various threshold settings. It is a plot of the True Positive Rate (TPR) versus the False Positive Rate (FPR). The ROC curve is also used as the performance evaluation metric in this work. Performance is evaluated by plotting the ROC curve for all three DL-based models, such as AlexNet, ResNet50, and VGG16, and the proposed Ensemble model for all the Eight datasets used in this experiment. The ROC

Table 3.4: Performance Evaluation

Sl.No	Plant Dataset	Performance Metrics	Model			
			AlexNet	ResNet50	VGG16	Proposed Approach (Ensemble Model)
Performance Evaluation for Multiclass Dataset						
1	Cardamom	Precision	0.96	0.98	0.98	0.99
		Recall	0.95	0.98	0.98	0.99
		F1-Score	0.95	0.98	0.98	0.99
		Accuracy(%)	95.32	98.24	97.66	99.41
2	Grape	Precision	0.96	0.98	0.98	0.99
		Recall	0.95	0.98	0.98	0.99
		F1-Score	0.95	0.98	0.98	0.99
		Accuracy (%)	95.01	98.50	97.75	99.25
3	Maize	Precision	0.92	0.95	0.94	0.96
		Recall	0.92	0.95	0.94	0.96
		F1-Score	0.92	0.95	0.94	0.96
		Accuracy (%)	94.48	96.85	96.32	97.75
4	Potato	Precision	0.99	0.96	0.96	0.99
		Recall	0.97	0.95	0.94	0.95
		F1-Score	0.98	0.96	0.95	0.97
		Accuracy(%)	99.06	98.59	98.12	99.53
5	Tomato	Precision	0.93	0.94	0.95	0.95
		Recall	0.93	0.93	0.94	0.93
		F1-Score	0.93	0.93	0.93	0.94
		Accuracy(%)	93.24	93.41	93.61	94.41
Performance Evaluation for Binary Class Dataset						
6	Cherry	Precision	1	1	1	1
		Recall	1	1	1	1
		F1-Score	1	1	1	1
		Accuracy(%)	100	100	100	100
7	Pepper	Precision	0.99	0.98	0.99	0.99
		Recall	0.99	0.98	0.99	0.99
		F1-Score	0.99	0.98	0.99	0.99
		Accuracy (%)	99.59	98.77	99.18	99.59
8	Strawberry	Precision	1	1	1	1
		Recall	1	1	1	1
		F1-Score	1	1	1	1
		Accuracy(%)	100	100	100	100

Table 3.5: Comparison of proposed Ensemble DL-based Model with State-of-the-art Methods

Sl.No	Plant	Techniques	Model	Accuracy (%)
1	Apple and Cucumber (Zhang et al., 2018)	Fusion of superpixel	K-means clustering	92.15
2	Multiple plant (Too et al., 2019)	Finetune the hyperparameters	DenseNet	99.75
3	Maize (Ahila et al., 2019)	PCA and Modified LeNet	Deep CNN	97.89
4	Multiple (Kamal et al., 2019)	Depthwise separable convolution	MobileNet	98.34
5	Cucumber (Ma et al., 2018a)	Comprehensive color feature	Deep CNN	93.40
6	Cucumber (Wang et al., 2021a)	DeepLab-V3	UNet	92.85
7	Cucumber (Zhang et al., 2019)	Dilated convolution and spatial resolution	Global pooling dilated CNN	95.18
8	Rice (Lu et al., 2017)	Sparse auto encoder	CNN	95.48
9	Wheat (Picon et al., 2019)	Super pixel segmentation	ResNet	96
10	Apple (Jiang et al., 2019)	Single shot multi box detector	DNN	78.8
11	Coffee (Esgario et al., 2020)	Multitask framework	CNN	97
12	Peach (Zhang et al., 2019)	Transfer Learning	AlexNet	100
13	Grape (Kerkech et al., 2018)	Color spaces and Vegetation indices	CNN	95.8
14	Multiple (Khamparia et al., 2019)	Encoder Network	Deep CNN	100
15	Coffee (Manso et al., 2019)	Color Spaces	Extreme Learning	99.09
16	Grape (Cruz et al., 2019)	Transfer Learning	ResNet	99.33
17	Grape (Gutiérrez et al., 2021)	Color Spaces	CNN	94.00
Proposed Ensemble Deep Learning-based Model				
18	Cardamom Cherry Grape Maize Pepper Potato Strawberry Tomato	Proposed Ensemble Model	Ensemble Model	99.41 100 99.25 97.75 99.59 99.53 100 94.41



(a) (b)

(c) (d)

Figure 3.5: ROC curves: a) ROC-curve for Cardamom dataset, b) ROC-curve for Grape dataset, and c) ROC-curve for Maize dataset, d) ROC curves for Potato dataset

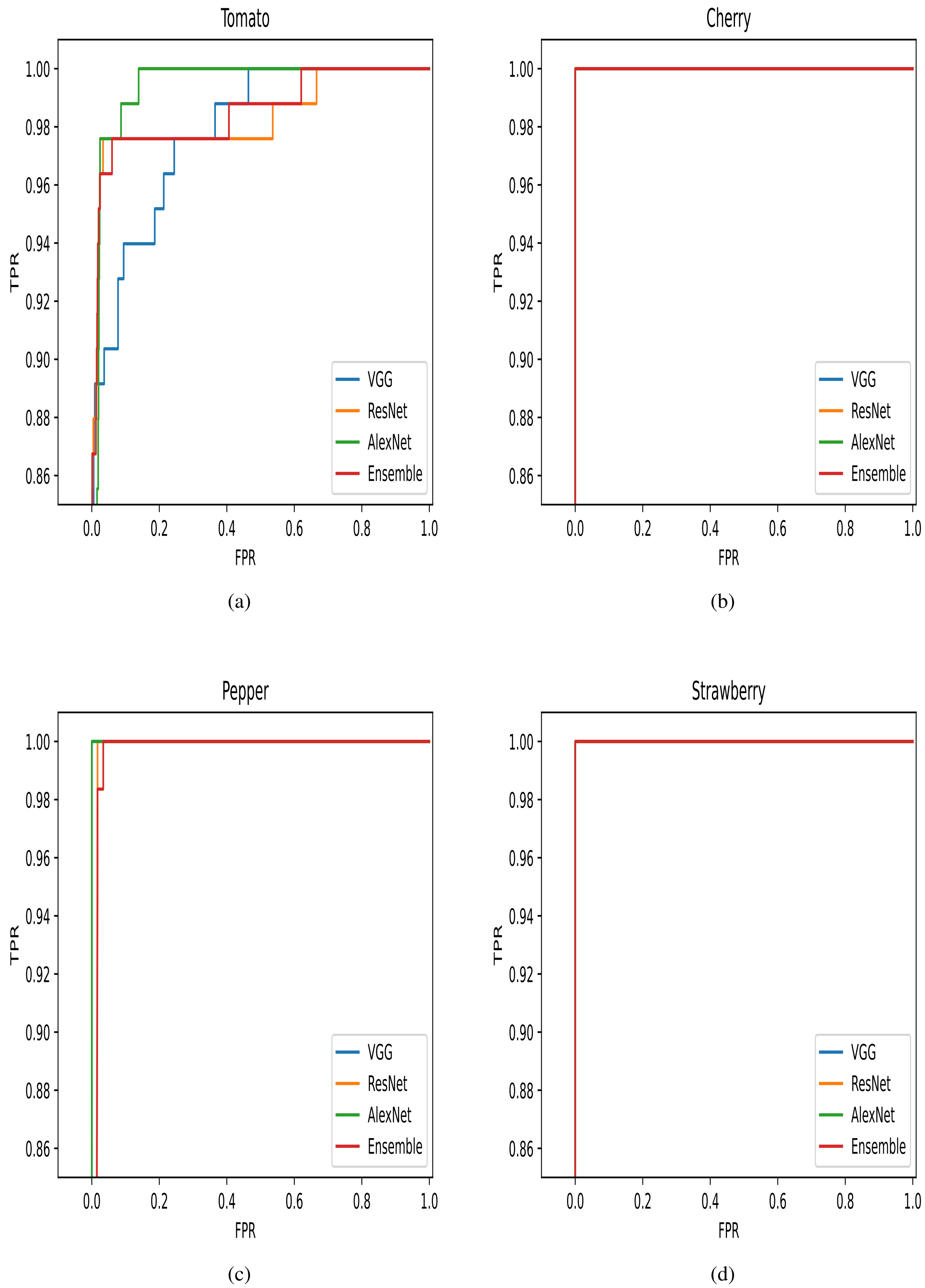


Figure 3.6: ROC curves: a) ROC-curve for Tomato dataset, b) ROC-curve for Cherry dataset, c) ROC-curve for Pepper dataset, and d) ROC-curve for Strawberry dataset

Table 3.6: External testing time for 10% dataset of multiclass and binary class datasets

Sl.No	Dataset Name	Testing Time
Multiclass dataset		
1	Cardamom	2 minutes, 2 seconds
2	Maize	5 minutes 26 seconds
3	Grape	4 minutes 25 seconds
4	Potato	2 minutes, 19 seconds
5	Tomato	50 minutes, 10 seconds
Binary class dataset		
6	Cherry	16 milliseconds
7	Pepper	2 minutes, 29 seconds
8	Strawberry	1 minute, 47 seconds

curve exhibits that the proposed Ensemble model outperformed the other three DL-based models. Figure 3.5a-3.5d and Figure 3.6 a) shows the ROC curve for multiclass datasets such as the Cardamom, Grape, Maize, Potato, and Tomato plant datasets, respectively. Figure 3.6b-3.6d shows the ROC curve for binary class datasets such as the Cherry, Pepper, and Strawberry datasets, respectively. The X axis is FPR, and the Y axis is TPR in both Figure. 3 and Figure. 4. The proposed approach outperformed compared with state-of-the-art methods. Table 3.5 compares the proposed work with other works using DL-based plant leaf disease detection; it shows that the proposed Ensemble DL-based plant disease detection approach performed better compared to other state-of-the-art models.

None of the works consider the testing time for plant disease detection to the best of our knowledge. The proposed Ensemble model's external testing time for multiclass datasets and the binary class dataset is shown in Table 3.6.

3.3 Summary

A robust, low-cost, and real-time plant disease detection approach is essential to ascertain an early-stage plant disease. Selecting appropriate hyperparameters is one of the challenges; this work proposed an empirical study on different plant datasets to find the optimal epoch size to get a precise result. In this work, it is observed that the performance of the MCLCNN is nearly the same with slight variation. Further, Ensemble DL-based plant leaf disease detection approach is proposed. It addressed the challenges such as class miss-classification, image capturing conditions, and classification time. AlexNet, ResNet50, and VGG16 were used as base models in the proposed approach since each base models are unique in nature to classify the images, and each model extracts independent features. The main objective of the proposed approach is to minimize the miss-classification rate; this is achieved by employing aforesaid three different DL-based models; the final classification outcome is based on the majority of the classifier's outcomes. This work outperformed the state-art-of-art approaches with 100% and 99.53% detection accuracy for binary class dataset and multiclass dataset, respectively.

In this work, the ensemble model is used with three different transfer learning models, which makes the model computationally costly, and it is essential to minimize the computational complexity; the next chapter addresses this challenge by using the adaptive attention mechanism.

Chapter 4

Plant Disease Classification using Multilevel Feature Fusion with Adaptive Channel and Pixel Attention Mechanism

Smart Agriculture is evolving; it has various activities such as marketing, weather details collections, climate information, soil fertility, water management, plant disease and pest management, and many more. Adopting smart agriculture improves the quality and quantity of the product; it saves the farmers time and minimizes irrelevant activities. In this chapter, Tomato Plant Disease Classification using Multilevel Feature Fusion with Adaptive[1] Channel, Spatial and Pixel Attention Mechanism is explained.

4.1 Tomato Plant Disease Classification

The tomato plant dataset taken from PlantVillage (Hughes & Salathé, 2015) is imbalanced. The performance of the classifier deteriorates when the classifier is trained with the imbalanced dataset. Another reason could be that diseases such as early blight and leaf blight exhibit similar features; in such cases, the trained classifier may produce a high miss-classification rate. Classification is defiant due to the imbalanced data, similar features with other classes, and contrasting information in the data. To cover these limitations, in this work, the tomato plant disease classification approach is proposed by using Multilevel Feature Fusion Network (MFFN) with Adaptive Channel and Pixel Attention Mechanism (ACPAM), and Adaptive Channel, Spatial, and Pixel Attention Mechanism (ACSPAM) as shown in Figure 4.1. Algorithm 4.1 describes the feature extraction using the channel, spatial, and pixel attention approaches.

Firstly, the MFFN extracts the multilevel features, then combines them, afterwards use the combined features for classification. To extract the features, adaptive selection of kernel sizes is employed for dimensionality reduction and applied combined ReLU.

In the proposed approach, ResNet50 is used as the base model. Upon the base

[1] The term "adaptive" in this context refers to the ability of the Channel, Spatial, pixel attention operation to adaptively and selectively focus on important features while discarding less relevant ones.

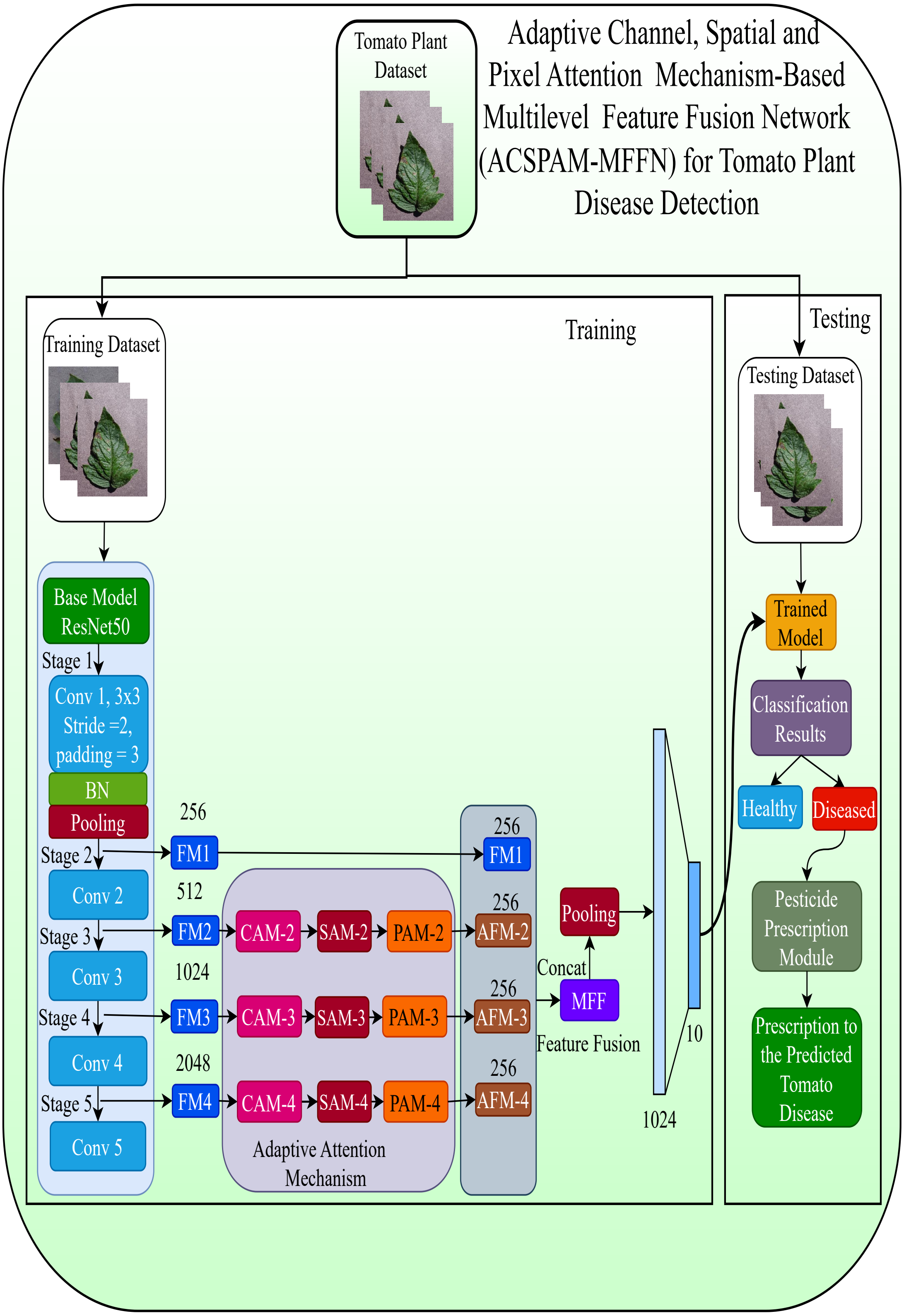


Figure 4.1: Adaptive Channel Spatial and Pixel Attention Based Multilevel Feature Fusion Network for Tomato Plant Disease Detection

Algorithm 4.1 Generating Channel Spatial and Pixel Attention Feature Map

Require: $FeatureMap(FM) = [\lambda_1, \lambda_2, \lambda_3, ..., \lambda_n]$
Ensure: Channel Spatial and Pixel Attention Feature Map
  $Y \leftarrow ReLU(Global(Conv3 \times 3(FM)))$
  $X \leftarrow ReLU(Global(Conv1 \times 1(FM)))$
  $GF \leftarrow [GP(\lambda_1), GP(\lambda_2), GP(\lambda_3), ..., GP(\lambda_n)]$
  for each *Channel* **do**
    $CWAF \leftarrow CWAF \oplus Sigmoid(GF.W)$
  end for
  for each *Element* **do**
    $EFM \leftarrow EFM \oplus CWAF \otimes X$
  end for
  for each *Element* **do**
    $CAFM \leftarrow CAFM \oplus EFM \oplus Y$
  end for

  $SAFM \leftarrow Sigmoid(Conv(AvgPool(CAFM); MaxPool(CAFM)))$
  $PAFM \leftarrow Sigmoid(Conv(ReLu(Conv(SAFM))))$
  for each *Element* **do**
    $AFM \leftarrow AFM \oplus CFAM \otimes PAFM$
  End For

model, stacked five more layers to extract the local and global features by making use of the stacked layers. The numbers on various Feature Maps (FM) in a neural network represent the number of channels or depth dimensions in each feature map. The term ”feature map” refers to the output of a particular layer in the neural network, and each feature map corresponds to a specific set of learned features or patterns detected by the network. the increasing number of feature maps in deep neural networks is essential for hierarchical feature learning, capturing complex representations, and increasing the model's capacity to learn from the data effectively. It allows the network to progressively learn more abstract and informative features, leading to better performance on various tasks, such as image classification, object detection, and segmentation.

In this work, initially, The FM generated by the base model are fed to the subsequent layer. In the first stage, the layer takes convolution layer kernel set as 3×3 with a stride two and padding three, followed by Batch Normalization (BN), and pooling, the output fed to the next stage. The FM generated by the subsequent stages from stage 2 is extracted on the basis of a layer separately and employs a Channel Attention Mechanism (CAM) and Pixel Attention Mechanism (PAM) to exploit the Depthwise Separable

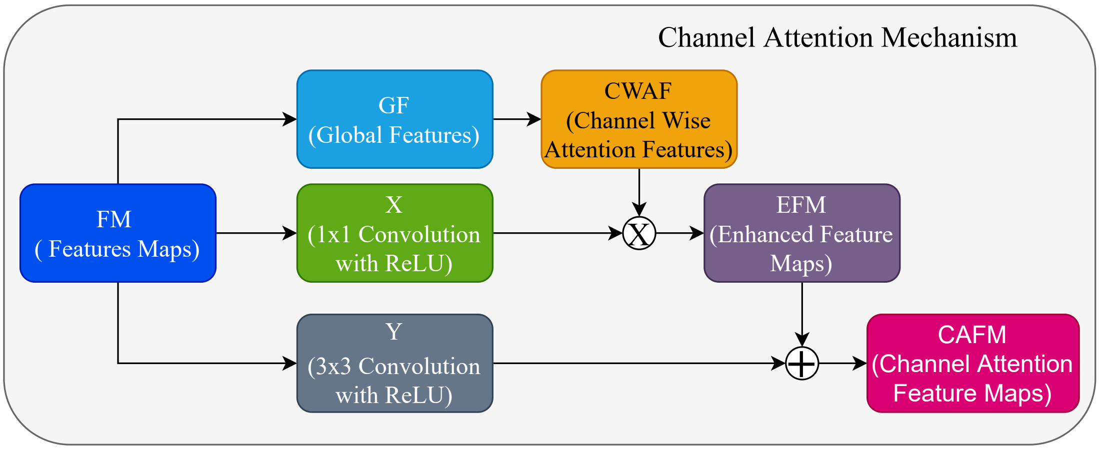


Figure 4.2: Channel Attention Mechanism

Convolution (DSC) to create Adaptive Feature Maps (AFM). In DSC, the input tensor's spatial dimensions are convolved separately for each channel of the input. Instead of applying a single 3D kernel across all channels, a 2D kernel is applied individually to each channel. This process reduces the number of parameters required, as each channel has its own small kernel, resulting in fewer computations. The adaptive nature of DSC allows it to efficiently learn and represent complex patterns and features present in the data, making it particularly useful in scenarios with limited computational resources. This efficiency and adaptability make DSC a fundamental building block for designing lightweight and efficient neural network architectures, especially for applications on mobile and edge devices where resource constraints are a concern. The FM generated by stage 2 and all the AFM caused by the subsequent stages are fused and applied pooling layer upon this, followed by a fully-connected layer with Softmax activation function (Figure 4.1). The Feature Maps FM is represented by Equation 4.1.

$$FM = [\lambda_1, \lambda_2, \lambda_3, ..., \lambda_n] \tag{4.1}$$

where λ_n indicates the FM of the n^{th} channel.

Figure 4.2 describes the CAM, it takes the input from the different stages of the feature extraction module with the base model as ResNet50 in a multilevel approach.

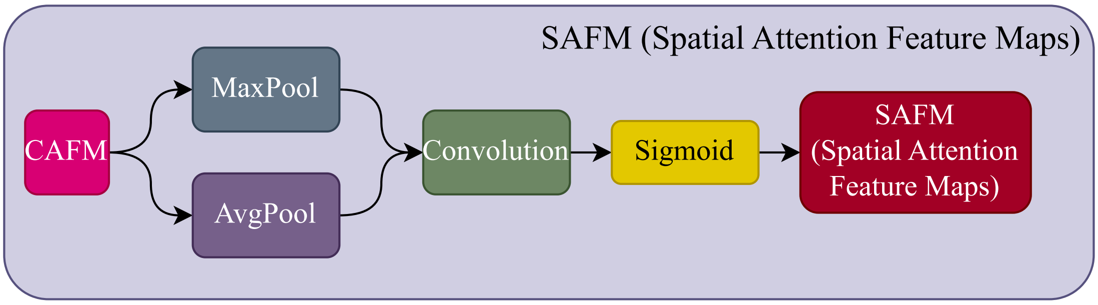


Figure 4.3: Spatial Attention Mechanism

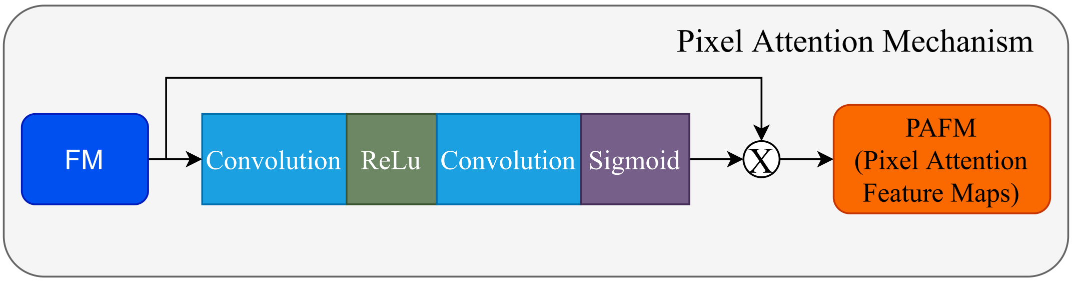


Figure 4.4: Pixel Attention Mechanism

Global[2] and local[3] features of the image provide rich information that are required for classification. Combining local and global features enhances classification performance. Most of the features can be extracted from the superficial layers. Deeper the network, the deeper features contain more ambiance knowledge.

Figure 4.3 describes the SAM, it takes the Channel Attention Features Maps (CAFM) as input and applied AvgPoll and MAxPool parallelly then concatenated the output with convolution, further applying sigmoid activation. Figure 4.4 describes the PAM, it takes the input as feature maps and applies the convolution, ReLU, convolution, and sigmoid in sequence, then, element wise multiplication is applied with the obtained results and the feature maps.

The FM Y is obtained after applying max pooling on the original FM with 3×3 convolution and ReLU activation function. This is generated to find saliency maps with dilated convolution; this enhances the amenable area and retains the spatial information.

[2]After the depthwise convolution, the pointwise convolution combines the output of the depthwise convolution across all channels using a 1×1 kernel. This step enables the model to capture global features by aggregating information from all spatial locations and channels. The pointwise convolution effectively summarizes the local features into a more compact and informative representation, capturing global context from the entire image.

[3]In attention process spatial information independently for each input channel. This allows the model to capture local patterns, edges, and textures present in specific regions of the input image. The independent processing of each channel helps the model identify and focus on local features in the data.

The FM Y is calculated by using Equation 4.2.

$$Y = ReLU(Global(Conv3 \times 3(FM))) \tag{4.2}$$

The FM is used to generate two new FM with 1×1 convolution. One is Global Features (GF) by using global average pooling, and another one is max pooling with ReLU activation function.

Each channel has unique information, the most valuable information to get the local features of the image. To extract such information, first obtained the feature map GF with 1×1 convolution.

$$X = ReLU(Global(Conv1 \times 1(FM))) \tag{4.3}$$

The Global Pooling (GP) is defined as follows

$$\text{GP}(\lambda_n) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \lambda_n(i, j) \tag{4.4}$$

The GF obtained as follows, by stacking all the Global pooling result

$$GF = [GP(\lambda_1), GP(\lambda_2), GP(\lambda_3), ..., GP(\lambda_n)] \tag{4.5}$$

GF has all the Global features of the image, to get more local features channel-wise information, Channel Wise Attention Features (CWAF) are obtained from GF by connecting all the layers with sigmoid activation as follows:

$$CWAF = Sigmoid(GF.W) \tag{4.6}$$

where W indicates the weight for each channel.

Next, obtain the Enhanced Feature Maps (EFM) by employing the element-wise multiplication operation with X; it is denoted as follows

$$EFM = CWAF \otimes X \tag{4.7}$$

where $\otimes$ denotes element wise multiplication.

CAFM are generated using the EFM and the Y with element-wise addition operation as follows.

$$CAFM = EFM \oplus Y \tag{4.8}$$

where $\oplus$ denotes element wise addition.

Spatail Attention Mechanism (SAM) is applied on the feature maps obtained by the CAM. Using the spatial relationships between features, create a spatial attention map. In contrast to channel attention, which concentrates on what is informative, spatial attention concentrates on information location. This applies AvgPool and MaxPool operations along the channel axis to create an effective feature descriptor before concatenating them to calculate the spatial attention. It is demonstrated that applying pooling operations along the channel axis effectively highlights informative regions.

$$SAFM = Sigmoid(Conv(AvgPool(CAFM); MaxPool(CAFM))) \tag{4.9}$$

Pixel attention is added upon the features maps obtained from spatial attention mechanism, since each pixel has certain features.

$$PAFM = Sigmoid(Conv(ReLu(Conv(SAFM)))) \tag{4.10}$$

$$AFM = SFAM \otimes PAFM \tag{4.11}$$

where $\otimes$ denotes element-wise multiplication.

ACSPAM was used to generate the AFM. In supremacy with multilevel features, ACSPAM, and multilevel feature extraction are combined. Features generated by ACPAM and the Multilevel Feature Fusion (MFF) are combined by employing combined

Table 4.1: Tomato Plant Leaf Dataset

Sl.No	Disease	# of Images	Total # of Images
1	Bacterial Spot (BS)	2127	18,160
2	Early Blight (EB)	1000	
3	Healthy (H)	1591	
4	Leaf Blight (LB)	1909	
5	Leaf Mold (LM)	952	
6	Mosaic Virus (MV)	373	
7	Septoria Leaf Spot (SP)	1771	
8	Target Spot (TS)	1404	
9	Two Spotted Spider Mite (SM)	1676	
10	Yellow Leaf Curl Virus (CV)	5357	

ReLU, as shown in Equation 4.12.

$$MFF = CombinedReLU(Concat[FM2, AFM3, AFM4, AFM5]) \tag{4.12}$$

where *Concat* denoted concatenation.

Global pooling is applied on MFF and applied fully connected layer upon it with softmax activation function.

4.1.1 Experimental Study and Result Analysis

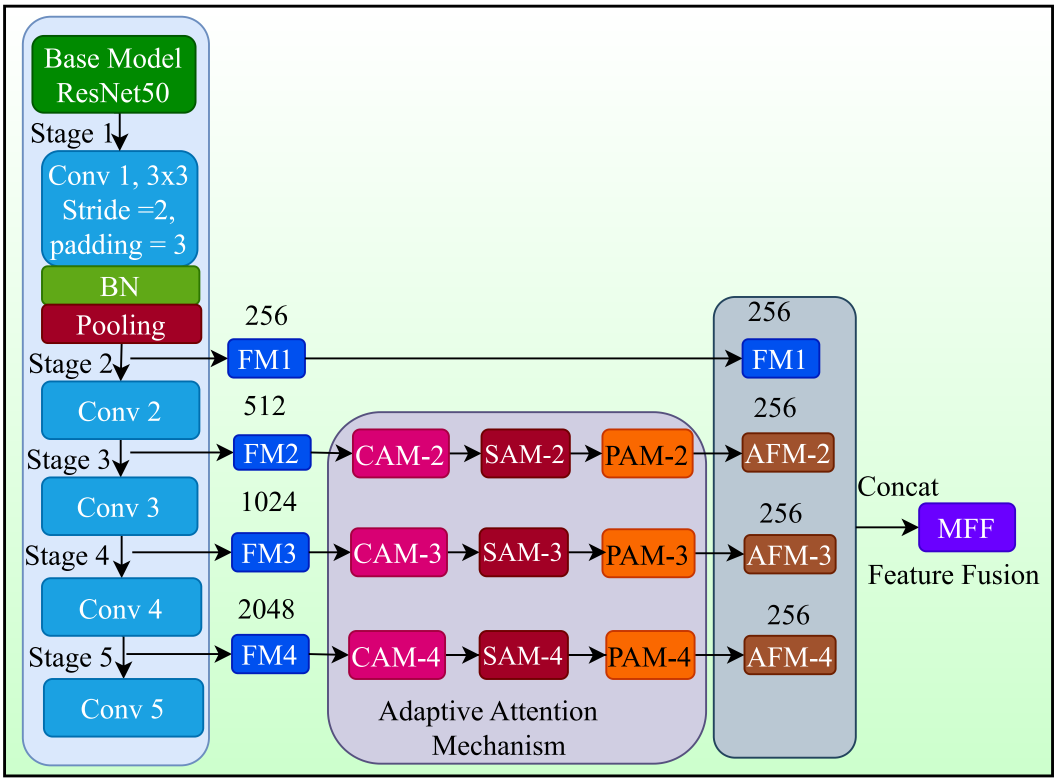
Dataset

In this work, ten classes of tomato plant leaf datasets are used, which are taken from PlantVillage (Mohanty et al., 2016b). The dataset utilized in this study is described in Table 4.1.

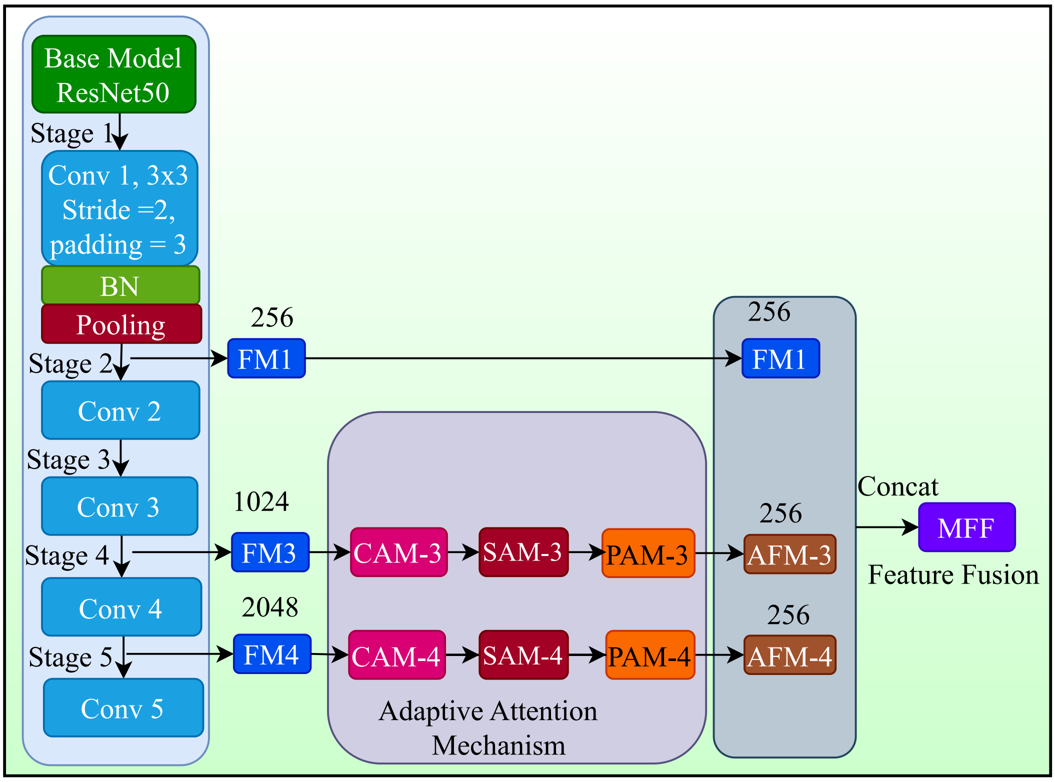
For the study, the dataset was split into 81% for training, 9% for validation, and 10% for external testing. All the images that are used in this work are resized to 224×224 to feed into the base model. All the experiments are executed with 100 epochs, batch size of 4, with a learning rate set to 0.0003.

Results and Analysis

Figure 4.6a-4.6c shows the confusion matrix of the proposed CAM method with two layers, three layers, four layers on tomato plant dataset for 10 classes. The proposed approach with 4 layers outperformed the other approaches with a classification accuracy

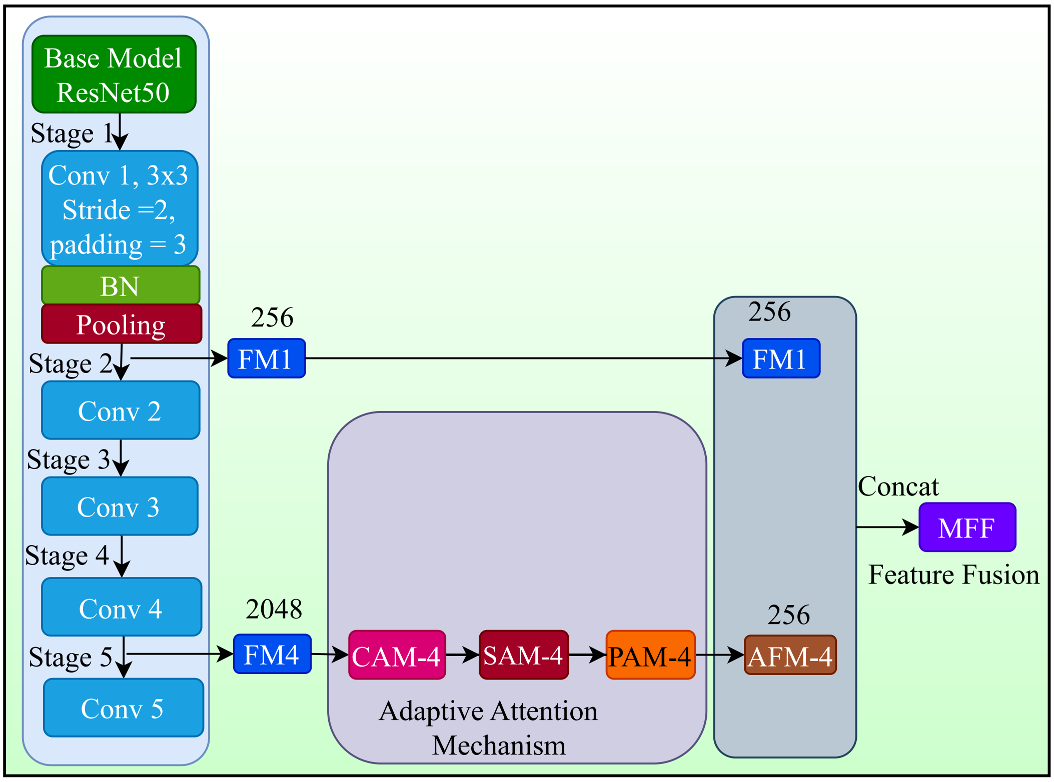


(a) ACSPAM-MFFN with 4 Layers



(b) ACSPAM-MFFN with 3 Layers

Figure 4.5: Multilevel Feature Fusion with Different Layers

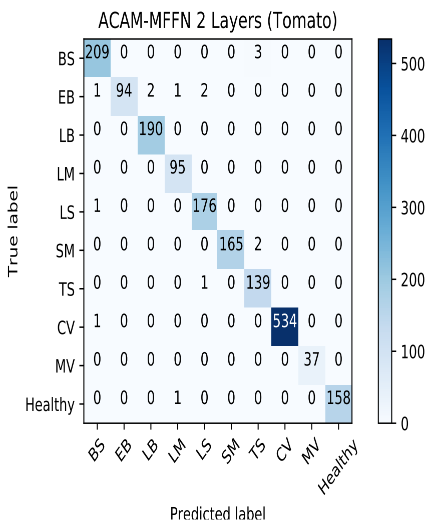


(c) ACSPAM-MFFN with 2 Layers

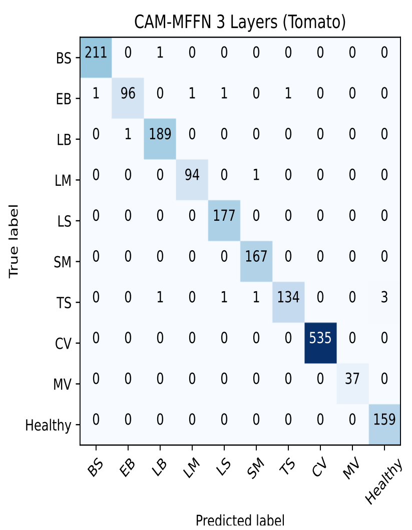
Figure 4.5: Multilevel Feature Fusion with Different Layers Contd.

of 99.39%. Figure 4.7a-4.7c shows the confusion matrix of the proposed PAM method with two layers, three layers, four layers on tomato plant dataset for 10 classes. The proposed approach with 3 layers outperformed the other approaches with a classification accuracy of 99.44%. Figure 4.8a-4.8c shows the confusion matrix of the proposed AC-PAM method with two layers, three layers, and four layers on the tomato plant dataset for 10 classes. The proposed approach with 4 layers outperformed the other approaches with a classification accuracy of 99.50%., and the results are optimal for the class imbalance dataset. Figure 4.9a-4.9c shows the confusion matrix for the proposed ACSPAM method with two, three, and four layers on the tomato plant dataset for 10 classes. Figure 4.9d shows the confusion matrix for Cardamom plant dataset with ACSPAM, and Figure 4.9e shows the confusion matrix for Grape plant dataset with ACSPAM

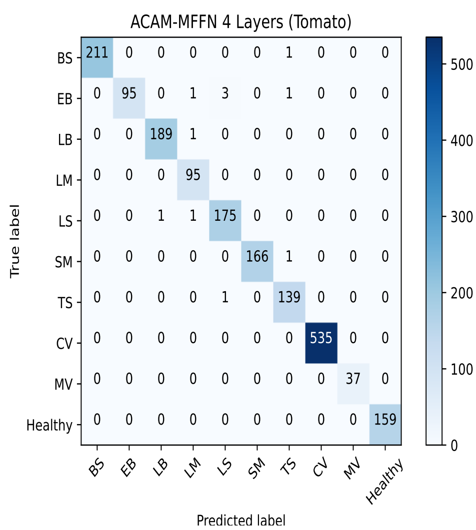
Figure 4.10 shows the accuracy and loss achieved during training and validation of the proposed CAM-MFFN approach, Figure 4.10a refers the accuracy against number of epochs for CAM-MFFN with four layers, Figure 4.10b refers the loss against a number of epochs for CAM-MFFN with four layers, likewise Figure 4.10c to Figure



(a) CAM-MFFN with 2 Layers

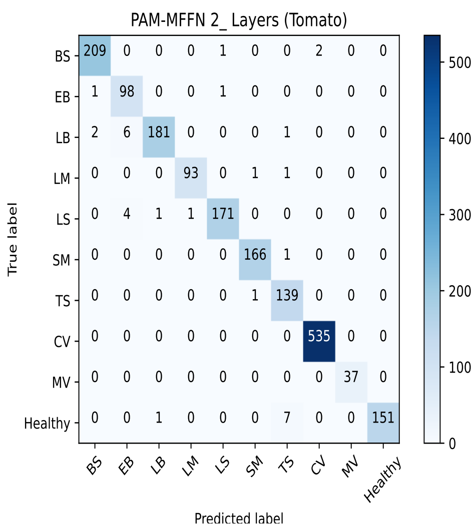


(b) CAM-MFFN with 3 Layers

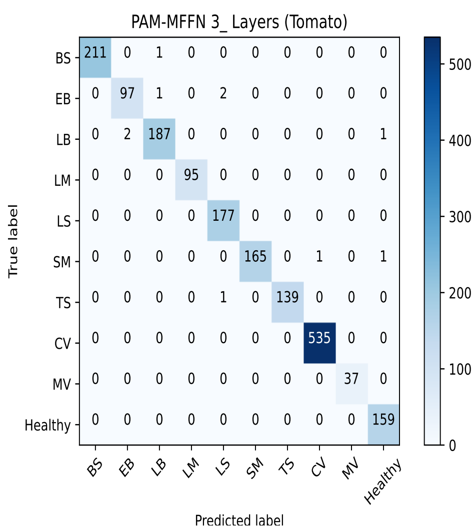


(c) CAM-MFFN with 4 Layers

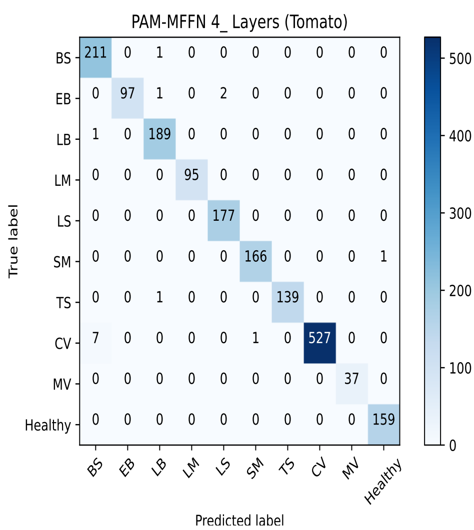
Figure 4.6: Confusion Matrix of the proposed CAM-MFFN Tomato Plant Leaf Classification



(a) PAM-MFFN with 2 Layers

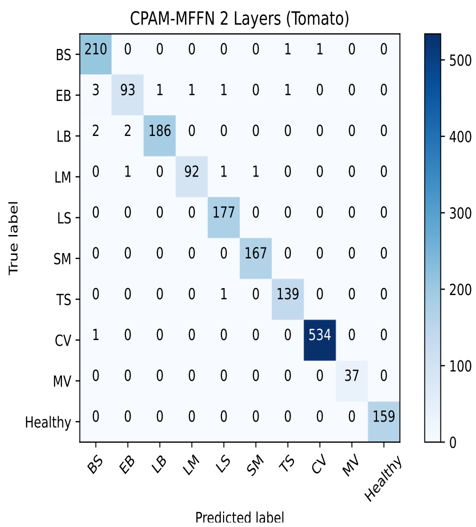


(b) PAM-MFFN with 3 Layers

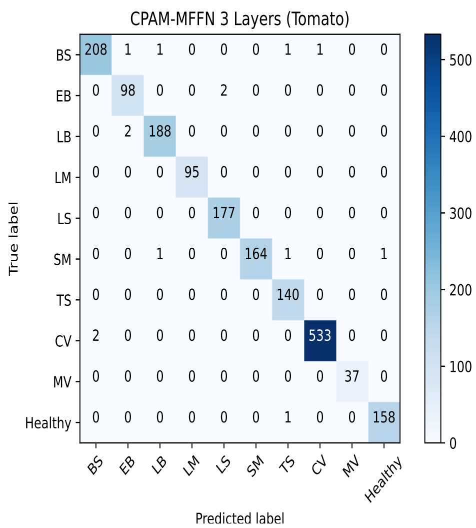


(c) PAM-MFFN with 4 Layers

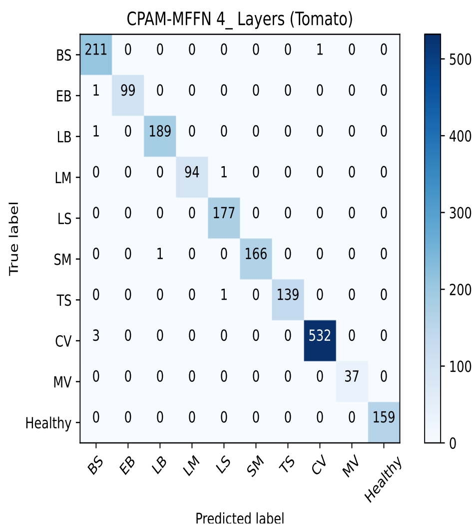
Figure 4.7: Confusion Matrix of the proposed PAM-MFFN Tomato Plant Leaf Classification



(a) ACPAM-MFFN with 2 Layers

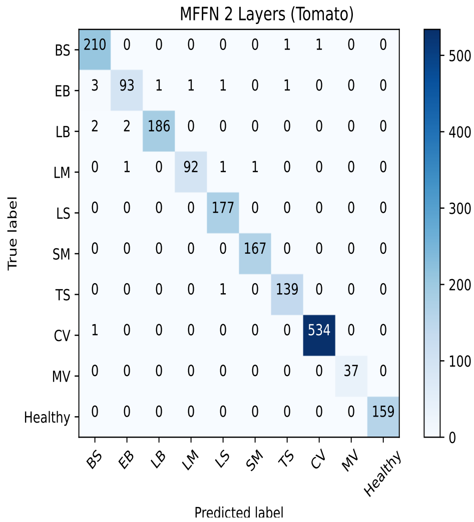


(b) ACPAM-MFFN with 3 Layers

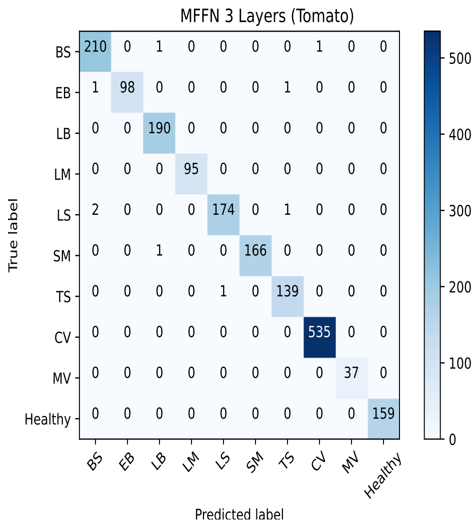


(c) ACPAM-MFFN with 4 Layers

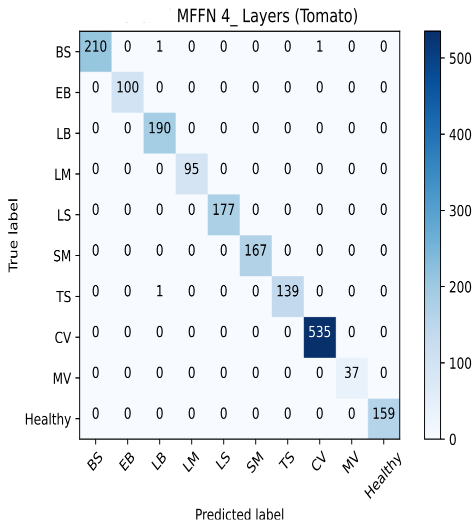
Figure 4.8: Confusion Matrix of the proposed ACPAM-MFFN Tomato Plant Leaf Classification



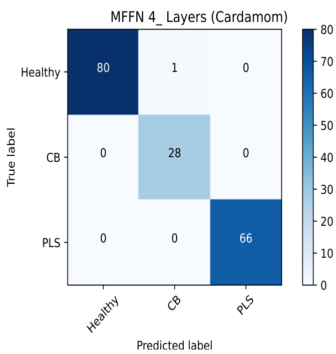
(a) ACSPAM-MFFN with 2 Layers



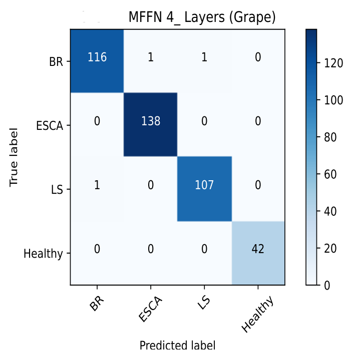
(b) ACSPAM-MFFN with 3 Layers



(c) ACSPAM-MFFN with 4 Layers

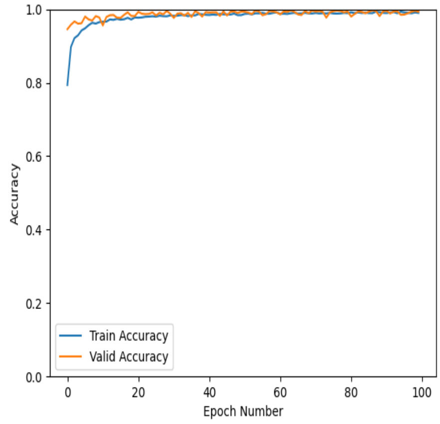


(d) ACSPAM-MFFN with 4 Layers

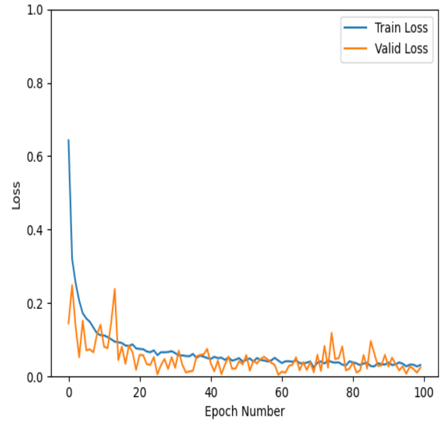


(e) ACSPAM-MFFN with 4 Layers

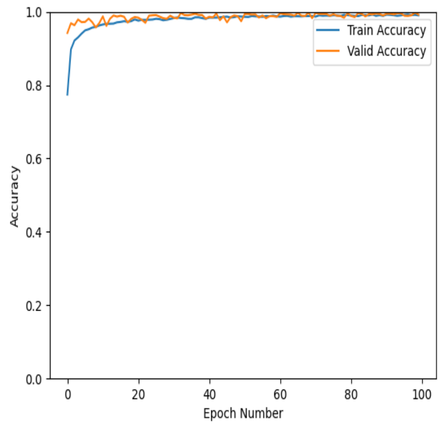
Figure 4.9: Confusion Matrix of the proposed ACSPAM-MFFN Tomato, Cardamom, and Grape Plant Leaf Classification



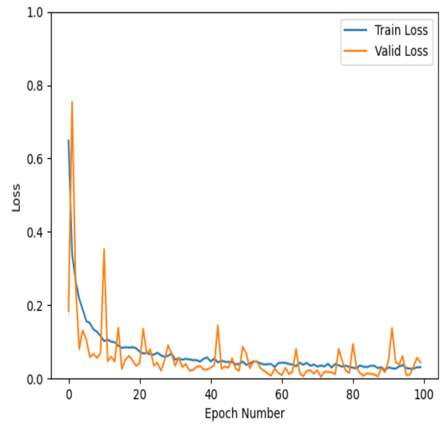
(a) Accuracy for CAM-MFFN with 4 layers



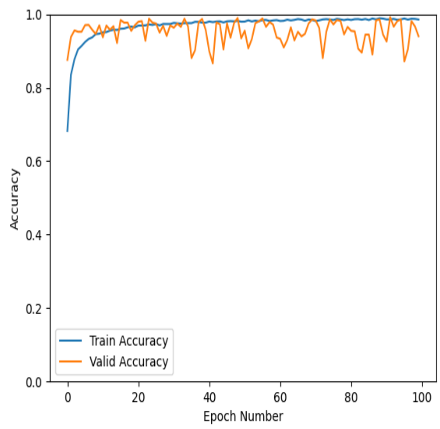
(b) Loss for CAM-MFFN with 4 layers



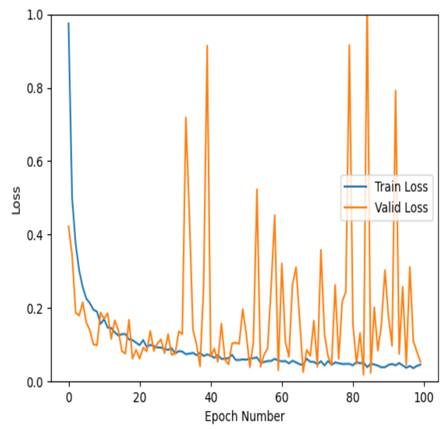
(c) Accuracy for CAM-MFFN with 3 layers



(d) Loss for CAM-MFFN with 3 layers

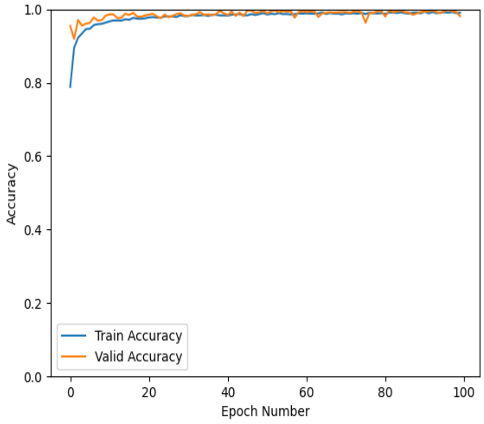


(e) Accuracy for CAM-MFFN with 2 layers

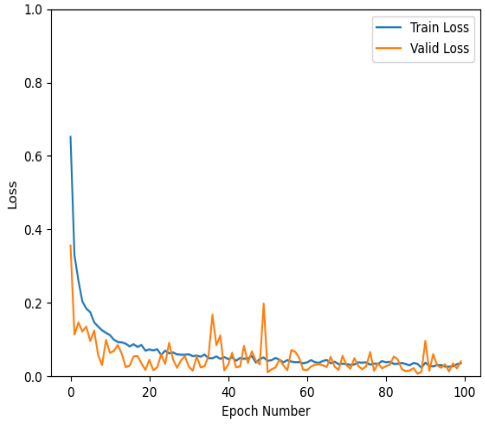


(f) Loss for CAM-MFFN with 2 layers

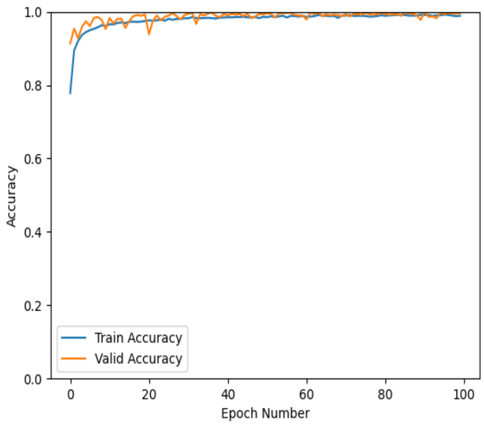
Figure 4.10: Accuracy and Loss of CAM-MFFN on Tomato Plant Dataset



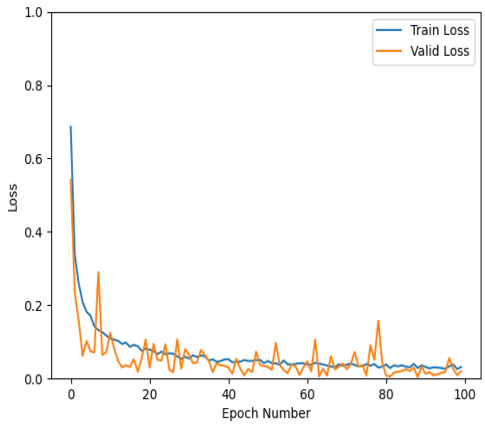
(a) Accuracy for PAM-MFFN with 4 layers



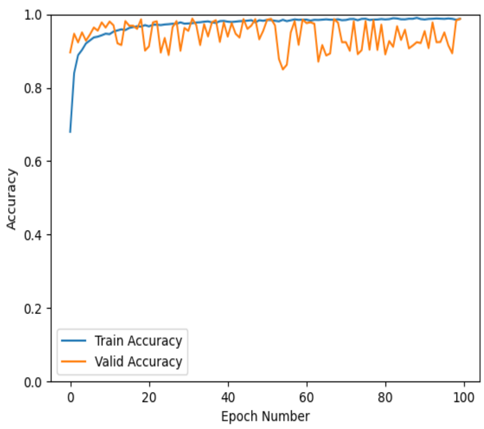
(b) Loss for PAM-MFFN with 4 layers



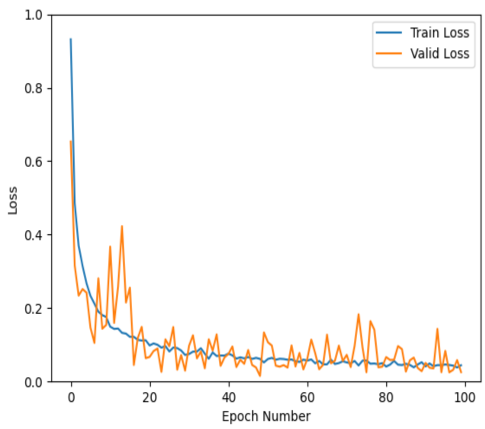
(c) Accuracy for PAM-MFFN with 3 layers



(d) Loss for PAM-MFFN with 3 layers

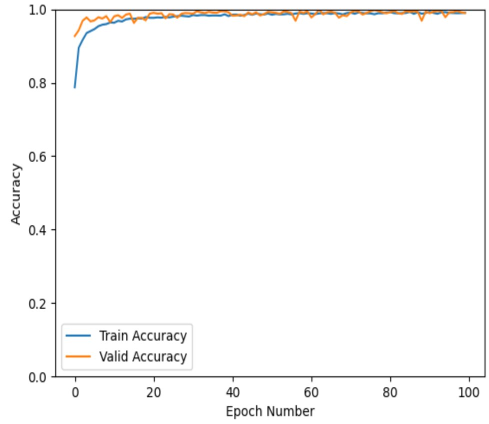


(e) Accuracy for PAM-MFFN with 2 layers

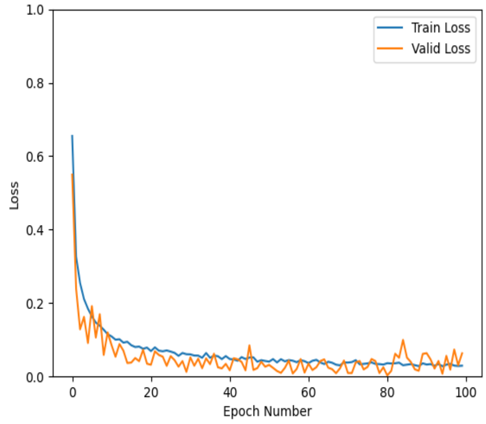


(f) Loss for PAM-MFFN with 2 layers

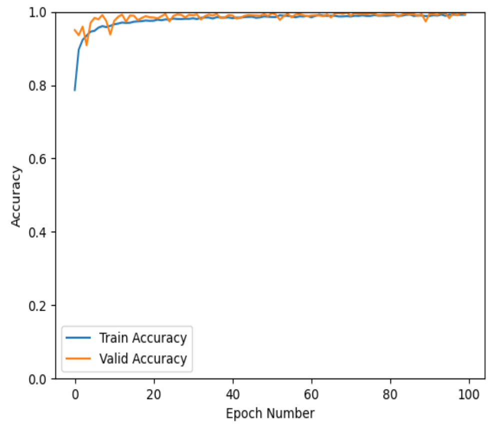
Figure 4.11: Accuracy and Loss of PAM-MFFN on Tomato Plant Dataset



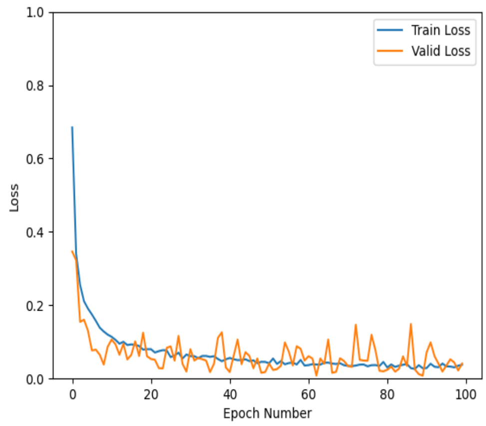
(a) Accuracy for ACPAM-MFFN with 4 layers



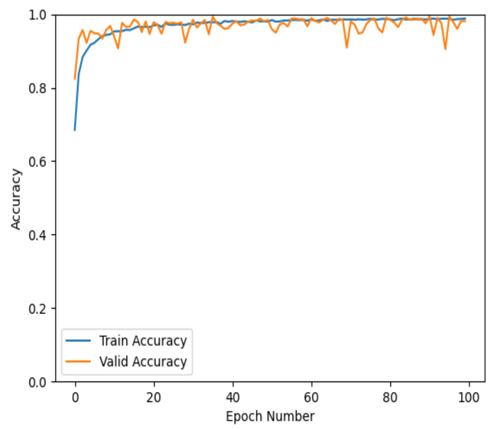
(b) Loss for ACPAM-MFFN with 4 layers



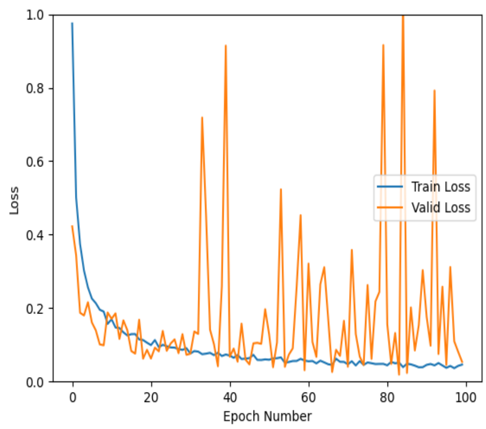
(c) Accuracy for ACPAM-MFFN with 3 layers



(d) Loss for ACPAM-MFFN with 3 layers

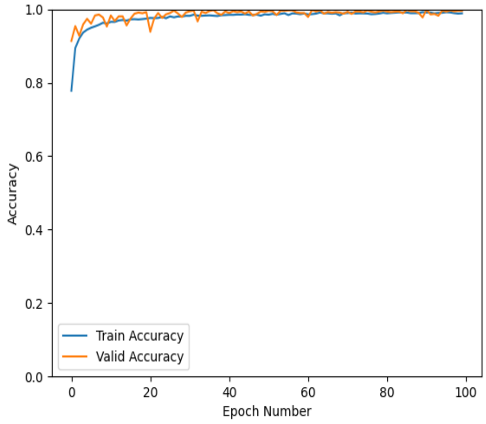


(e) Accuracy for ACPAM-MFFN with 2 layers

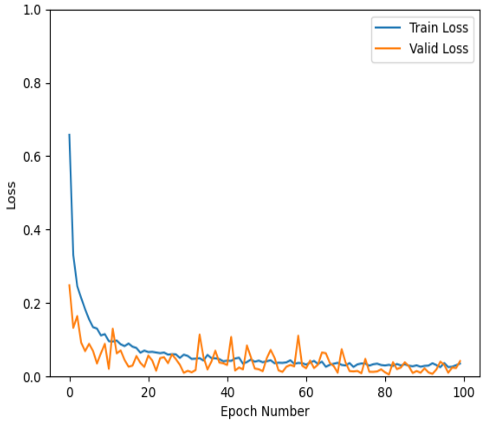


(f) Loss for ACPAM-MFFN with 2 layers

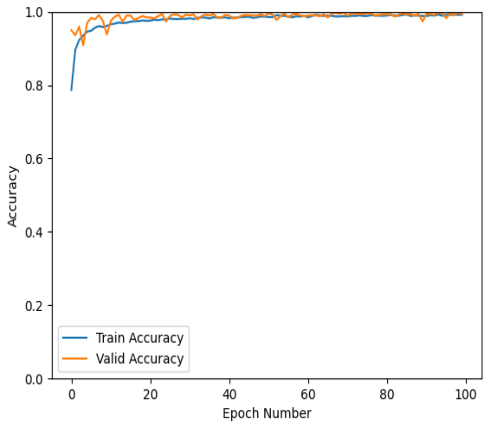
Figure 4.12: Accuracy and Loss of the proposed ACPAM-MFFN on Tomato Plant Dataset



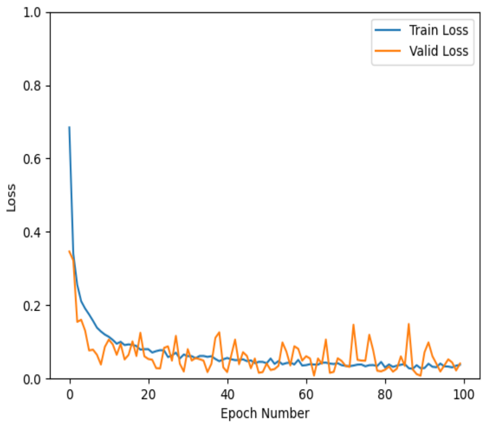
(a) Accuracy for ACSPAM-MFFN with 4 layers



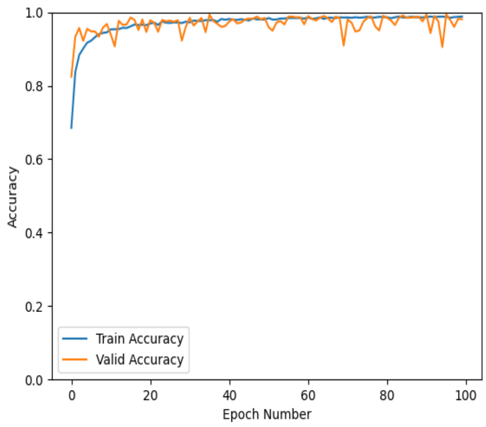
(b) Loss for ACSPAM-MFFN with 4 layers



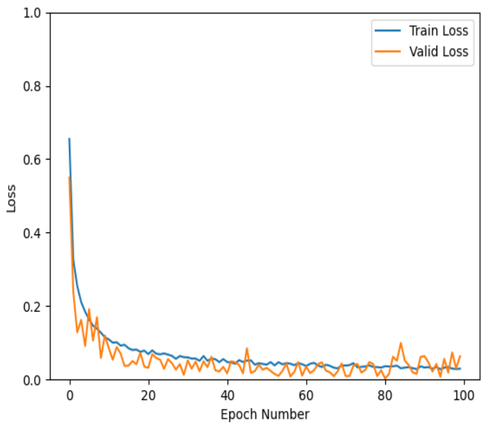
(c) Accuracy for ACSPAM-MFFN with 3 layers



(d) Loss for ACSPAM-MFFN with 3 layers



(e) Accuracy for ACSPAM-MFFN with 2 layers



(f) Loss for ACSPAM-MFFN with 2 layers

Figure 4.13: Accuracy and Loss of the proposed Approach on Tomato Plant Dataset

10f for CAM-MFFN with three layers and two layers. As shown the Figure 10a to Figure 10f CAM-MFFN with four layers performs better as compared to the other two approaches, since the validation accuracy improves with less fluctuation similarly, validation loss also minimizing as the number of epochs increases. But with CAM-MFFN with three layers and two layers more fluctuation in the validation accuracy and validation loss, that too with only 2 layers the validation loss fluctuation is more. It shows the training accuracy gradually increases as the number of epochs increases and is saturated after 90 epochs, it also shows that loss also converged after 90 epochs. Similarly, Figure 4.11 and 4.12 shows the accuracy and loss against the number of epochs for the PAM-MFFN and ACPAM-MFFN approaches. Figure 4.13 shows the accuracy and loss against the number of epochs for the ACSPAM-MFFN with two, three, and four layers. As observed in Figures 10, 11, 12, and 13, ACSPAM-MFFN performed best, which has fewer fluctuations in the validation accuracy and performed steadily in terms of train and validation loss.

Table 4.2 analyses the proposed approach using performance metrics such as Accuracy, Precision, Recall, and F-Score. The results depict that, the proposed ACSPAM-MFFN with 4 layers outperforms by achieving Precision, Recall, and F1-Score as 1.

Table 4.3 compares the results of all the above studies for tomato plant disease detection. ACSPAM-MFFN with four layers outperforms other methods by attaining the best external testing accuracy of 99.83%. Table 4.3 shows the proposed approach results with state-of-the-art techniques on the tomato plant dataset and other plant datasets. The results shown in Table 4.3 state the proposed approach outperformed the state-of-the-art approaches, by attaining a validation accuracy of 99.88% and external testing accuracy of 99.83%, by taking 501.42 seconds to generate the report for 1812 testing images. The proposed approach atained bext testing accuracy of 99.26%, 99.42% and 99.83% on Grape, Cardamom, and Tomato plant dataset respectively

4.1.2 Pesticide Prescription Module

To minimize crop loss and enhance the crop's quality with good yield, proper precaution measures need to be taken at the right time; as the classification models detect certain diseases in the plant, further suggestions to control disease to be given. To the best of our knowledge, most of the work in the literature focused on the classification of

Table 4.2: Performance Evaluation of the proposed ACSPAM

Sl.No	Plant Name	Accuracy	Precision	Recall	F-1Score
1	Tomato	99.83	1	1	1
2	Grape	99.26	0.99	0.99	0.99
3	Cardamom	99.42	0.99	1	0.99

plant disease; this work proposed tomato plant disease classification with a pesticide prescription module. The trained ACSPAM-MFFN approach is tested externally. As a test case, this work considers tomato plant leaf images taken from the test dataset.

In this work, we have collected the control measures for tomato plant leaf diseases (NIPHM, 2014)(Marissa et al., 2021)(Bayer & Seminis., 2021), which are described in Table 4.4.

4.1.3 External Testing Phase

The trained ACSPAM-MFFN performance is analyzed in the testing phase with tomato plant leaf images. Once the trained model is loaded, the tomato plant leaf image from the testing dataset is sent to the trained model. It classifies the tomato leaf image as healthy or diseased. If the image is diseased, it sends the diseased name to the pesticide prescription module to get the appropriate pesticide details for the detected tomato plant disease. Figure 4.14 describes the test case results suggested by the pesticide prescription module of the proposed approach. All the appropriate control measures for tomato plant diseases are described in Table 4.4. Algorithm 4.2 describes the working of testing the proposed approach with a prescription suggestion.

4.2 Summary

It is difficult to obtain a balanced dataset due to the unavailability of diseases in certain stages, geographical diseases, etc. It is essential to produce a precise result with the class imbalance dataset. In this regard, this work focused on to handle the class imbalance dataset and also handling the diseases with similar symptoms. Proposed ACSPAM with MFFN, which outperforms other models by obtaining the best validation accuracy of 99.88% and testing accuracy of 99.83% for tomato plant dataset. Analyze the performance of the proposed approach on cardamom and grape plant dataset, it attained a testing accuracy of 99.42, and 98.76 respectively. Figure 4.15 describes the generic in-farm testing of the plant disease detection approach.

Table 4.3: Comparison of proposed Tomato Plant Leaf Classification with other state-of-the-art approaches on Tomato and other plant datasets

Sl.No.	Plant Name	Method	Accuracy (%)
1	Multiple (Arun & Umamaheswari, 2022)	PC	98.14
2	Wheat (Rangarajan et al., 2022)	DarkNet 19	100 (F1-Score)
3	Cucumber (Wang et al., 2021b)	DeepLabv3 and U-Net	92.85
4	Rice (Wang et al., 2021)	DSC	94.65
5	Cucumber (Zhang et al., 2020b)	EfficientNet	96
6	Multiple (Chen et al., 2021b)	CNN	93.75
7	Strawberry (Shin et al., 2021b)	ResNet50	98.11
8	Paddy (Jiang et al., 2021b)	VGG16	97.22
9	Wheat (Jiang et al., 2021b)	VGG16	98.75
10	Cucumber (Ma et al., 2018b)	DCNN	93.4
11	Cucumber (Nanehkaran et al., 2020)	CNN	75.59
12	Paddy (Jain & Dharavath, 2021)	SVM	95
13	Beans (Elfatimi et al., 2022b)	MobileNetV2	99.4
14	Multiple (Kour & Arora, 2019)	Particle SO and SVM	95.2
	Tomato Plant Disease Detection		
15	Tomato (Li et al., 2022b)	RAN	98.75
16	Tomato (Schor et al., 2016)	PCA	95.2
17	Tomato (Wu et al., 2022)	CNN	94.5
18	Tomato (Chen et al., 2020b)	ABC and RAN	89
19	Tomato (Wu et al., 2020a)	GoogleNet	94.33
20	Tomato (Zhou et al., 2021b)	Residual Network	95
21	Tomato (Liu & Wang, 2020)	DarkNet53	92.39
22	Tomato (Zhao et al., 2022b)	CNN	95.20
23	Tomato (Qi et al., 2022b)	Modified YOLOV5	91.07
24	Tomato (Abbas et al., 2021b)	DenseNet121	97.11
25	**Proposed Approach**	**ACSPAM-MFFN**	**99.83**
	Grape Plant Disease Detection		
26	**Proposed Approach**	**ACSPAM-MFFN**	**99.26**
	Cardamom Plant Disease Detection		
27	**Proposed Approach**	**ACSPAM-MFFN**	**99.42**

Table 4.4: Tomato Plant Diseases and Control Measures.

Sl.No.	Disease Name	Prescription	Dilution
1	Bacterial Spot (NIPHM, 2014)	Streptomycin Sulfate 9% and Tetracycline Hydrocholride 1% SP	40-100ppm
2	Early Blight (NIPHM, 2014)	Azoxytrobin 23% SC Captan 50% WP Copper oxycholride 50% WP	200ml in 200 L 1000gm in 100 to 200 1000gm in 300 to 400L
3	Mosaic Virus (NIPHM, 2014)	1) Gather all the contaminated leaves and plant parts, then destroy them by fire. 2) Manoeuvre 4 to 5 insect sticky or pest sticky traps for an acre, such as yellow or blue pan. 3) Maneuver the light traps at least one for an acre during 6 pm and 10 pm.	–
4	Late Blight (NIPHM, 2014)	Mancozeb 35% Mancozeb 75% WP OR Zineb 75% WP SC	200ml in 200L 600 to 800gm in 300L
5	Leaf Mold (NIPHM, 2014)	Benjovindiflutpr Difenoconazle	10.5-13.5 fl.oz
6	Septoria Leaf Spot (NIPHM, 2014)	Mancozeb 75% WP	600-800gm in 300L
7	Target Spot (Bayer & Seminis., 2021)	Products with Chlorothalonil, Mancozeb, and Copper Oxychloride	600-800gm in 300L
8	Spider Mite (Bayer & Seminis., 2021)	Fenazaquin 10% Spiromesifen 22.9% EC	500ml in 200L 250ml in 200L
9	Yellow Leaf Curl (NIPHM, 2014)	Dimethoate 30% EC Imidacloprid 17.8	396ml in 200-400L 60-70mll in 200L 80gm in 160L
		Organic Treatment: Neem Seed Kernel extract 5% or Azadirachtin 5% SL	80gm in 160L

```
  (fc): Linear(in_features=1024, out_features=10, bias=True)
  (logSoftmax): LogSoftmax(dim=1)
)
/data/apps/python3/lib/python3.8/site-packages/torch/nn/modules
l to include dim=X as an argument.
  input = module(input)
The leaf has Bacterial spot disease
Prescritpion to class BS
Streptomycin Sulfate 9% and  Tetracycline Hydrocholride 1% SP
Dilution: 40-100ppm
```

(a) Bacterial Spot

```
  (fc): Linear(in_features=1024, out_features=10, bias=True)
  (logSoftmax): LogSoftmax(dim=1)
)
/data/apps/python3/lib/python3.8/site-packages/torch/nn/modules/container
l to include dim=X as an argument.
  input = module(input)
The leaf has Early Blight disease
Prescritpion to class EB
Azoxytrobin  23% SC in 200 liters  of water per acre
Dilution:200ml in  200 liters
OR
Captan 50% WP,   Dilution: 1000 gm in 100-100 liters  of water per acre
```

(b) Early Blight

```
  )
  (fc): Linear(in_features=1024, out_features=10, bias=True)
  (logSoftmax): LogSoftmax(dim=1)
)
/data/apps/python3/lib/python3.8/site-packages/torch/nn/modules/container.py
l to include dim=X as an argument.
  input = module(input)
The leaf has Late Blight disease
Prescritpion to class LB
Mancozeb 35% SC,  Dilution: 200ml in  200 liters
OR
 Mancozeb 75% WP, Dilution: 600 - 800 gm in 300 liters  of water per acre
```

(c) Late Blight

```
  (fc): Linear(in_features=1024, out_features=10, bias=True)
  (logSoftmax): LogSoftmax(dim=1)
)
/data/apps/python3/lib/python3.8/site-packages/torch/nn/modul
l to include dim=X as an argument.
  input = module(input)
The leaf has Leaf spot disease
Prescritpion to class LS
Mancozeb 75% WP
Dilution: 600 - 800 gm in 300 liters  of water per acre
```

(d) Leaf Spot

Figure 4.14: Testing Result and Suggested Prescription

Algorithm 4.2 Testing and Pesticide Prescription Module

Require: Tomato plant leaf image
Ensure: Class name and suggested prescriptions
  Step 1: Load the trained tomato plant disease detection model.
  Step 2: Capture real-time Tomato plant leaf image.
  Step 3: Send the real-time plant image to trained model.
  Step 4: Pre-process the plant leaf image.
  Step 5: Trained model produces the classification result.
  if Healthy **then**
    Send the result as Healthy.
  else if Disease 1: **then**
    Sends the prescription to the user for disease 1
  else if Disease 2: **then**
    Sends the prescription to the user for disease 2
    .
    .
    .
    ,
  else if Disease 9: **then**
    Sends the prescription to the user for disease 9
  end if

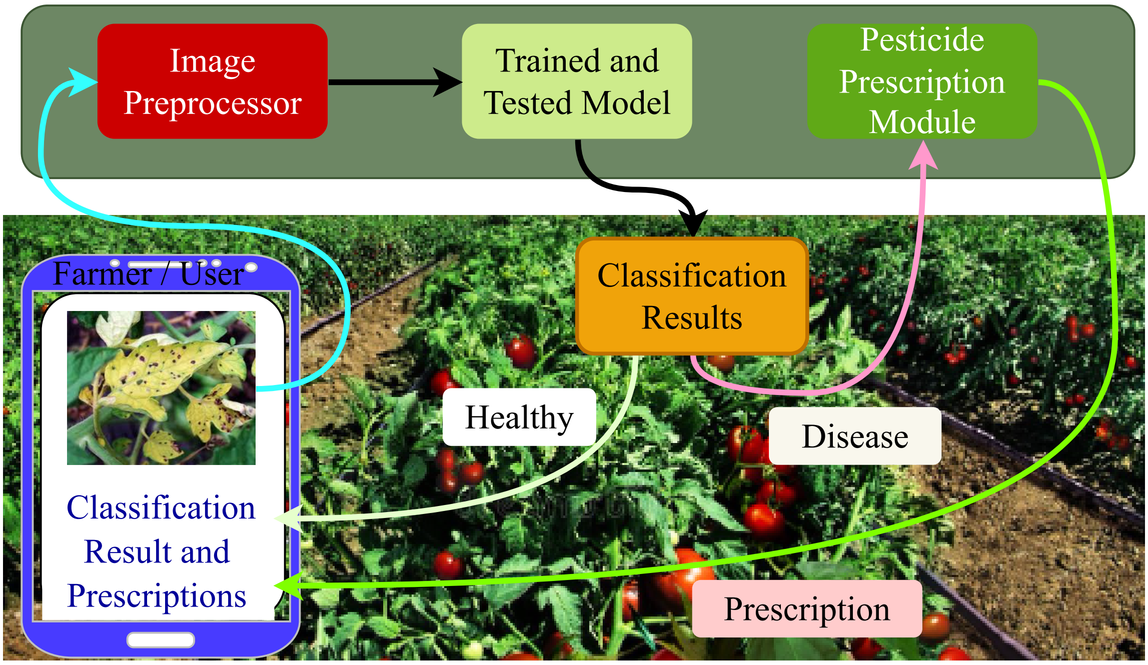


Figure 4.15: In-farm Tomato Plant Disease Diagnosis.

The datasets used in this study are taken from PlantVillage, where images are captured in a controlled environment and have no complex background. It is essential to collect the dataset in a complex environment, where the images should have some complex background and low-resolution images. Addressing the limitations of controlled environments, carefully curated diverse datasets from various sources, encompassing challenging lighting conditions. Preprocessing techniques were applied to clean noisy images and enhance quality. Findings underscore the importance of diverse data for practical applications in complex scenarios, paving the way for advanced techniques in a subsequent chapter.

Chapter 5

Plant Disease Detection with Complex Background and Noisy Images

The spice queen is cardamom. In the evergreen woods of Karnataka, Kerala, Tamil Nadu, and the North-Eastern states of India, it is a native plant. The third-largest producer of cardamom is India. In addition to being used as a flavour, cardamom is frequently employed in allopathic and ayurvedic treatment (Manju et al., 2018). It is a money mint crop; modern technology for agro production has been developed and widely accepted in all cardamom growing territories in India. Still, the spread of various pests and diseases remains a challenge that is considered as a significant production barrier experienced by the cardamom sector. Small cardamom is affected by a host of pathogenic bacteria, which seriously damages the crop and is often harmful. Diseases infected with cardamom plants such as colletotrichum blight and leaf spot have emerged dramatically in fields where crop management is not considered (Manju et al., 2018). Detecting plant diseases is difficult since photos are captured in real-time settings from farm fields, which has a complex background. A method for detecting cardamom plant leaf disease utilising photos with complicated backgrounds is proposed in this article.

5.1 Dataset Description

Cardamom Dataset 2021: In this work 1724 cardamom plant leaf images of three classes, such as Colletotrichum Blight and Phyllosticta Leaf Spot and healthy category are collected. The Indian Cardamom Research Institute officers at Regional Station

Table 5.1: Dataset used in the proposed research

Sl.No	Class	# of Images	Train/Test Split	Total # of images
This research generated a dataset of cardamom plant leaves.				
1	Colletotrichum Blight	280	252/28	1724
2	Healthy	781	700/81	
3	Phyllosticta Leaf Spot	663	597/66	
PlantVillage dataset (Grape) (Hughes & Salathé, 2015)				
4	Black Rot	1180	1062/118	4062
5	ESCA	1383	1245/138	
6	Healthy	423	381/42	
7	Isariopsis Leaf Spot	1076	968/108	

Sakaleshpur in the state of Karnataka, a division of Spices Board India, assist in labelling these. From April to June 2021, all photographs are taken during the daylight hours of 10 a.m. to 5 p.m. The Cardamom dataset collected during the year 2021 is described in Table 5.1. The illnesses listed in Table 5.1 are common to the cardamom plant and have an impact on the crop's growth and productivity. In this work, each image is additionally captured in a farm field scenario without the use of any technical tools, maintaining all the archive information and removing the background from the image. The original photos of cardamom plant leaves have a complicated background with various dimensions and photographing circumstances. Images of three distinct varieties of cardamom plant leaves are shown in Figure 5.1.

PlantVillage dataset (Hughes & Salathé, 2015): One of the commonly used and openly accessible datasets in the field of classifying plant diseases is called PlantVillage. It has approximately 54,284 photos, all of which have annotations. It is difficult to spot unfavourable circumstances in these pictures, including the complicated background. The Plant Village dataset's grape dataset is used in this investigation. Table 5.1 provides information on the dataset utilised in this experiment.

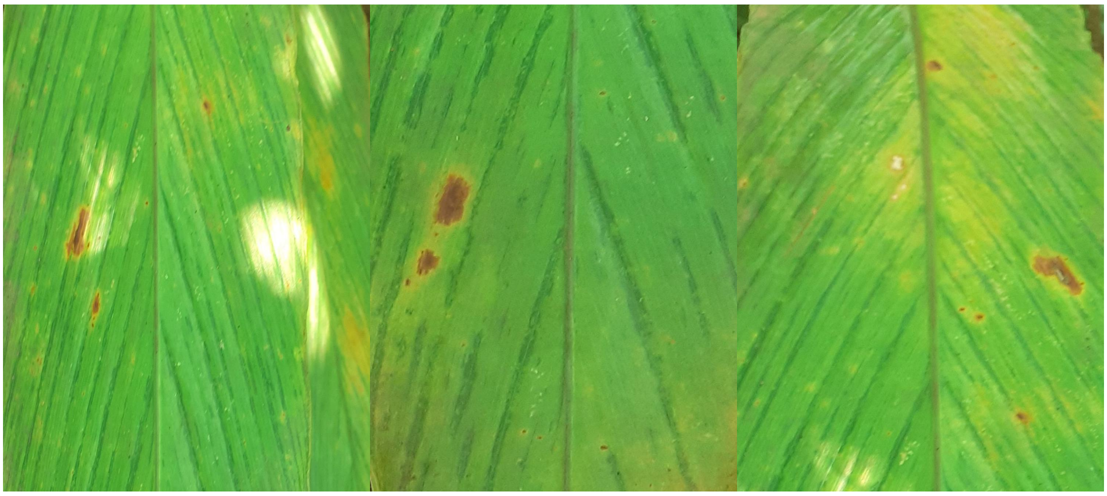
5.2 Proposed Method

This research suggested a method for detecting cardamom plant leaf illness by removing the image's complicated background and noise using U^2-Net and EfficientNetV2 deep learning-based models as a classifier.

5.2.1 Background Removal

Cardamom plant leaf images are of RGB, collected with a complex background with different dimensions and resolution, and the leaf is surrounded by several other factors, generally in the environment.

While in most cases, computer vision algorithms remove the background from an image, such as image thresholding in OpenCV and grab hut techniques (Cruz et al., 2019). These techniques help when the background color differs from the interesting object; in such cases, it is easy to remove the background by utilizing green and blue screens to eradicate the foundation and replace it with another scene.



a). Cardamom colletotrichum leaf blight

b). Cardamom healthy

c). Cardamom phyllosticta leaf spot

Figure 5.1: Cardamom plant leaf images: a) Cardamom colletotrichum leaf blight, b) cardamom healthy, c) Cardamom phyllosticta leaf spot

Without intentional pre- or post-processing, removing a backdrop from an image is a very stimulating exercise. It can be extremely difficult to get a correct form if the object's colour is really close to the background because of the soft edges or shadows.

Background and noise elimination procedure used in this research is U^2-Net (Qin et al., 2020) and Figure 5.2 depicts it. It takes an input image, generates a mask of the region of interest, and then performs a bitwise operation on the original image and the mask generated by the U^2-Net. Figure 5.3 depicts the U^2-Net architecture as a twofold interlaced U-structure. It is divided into three sections. The first component is a six-staged encoder that employs ReSidual U-Block (RSU) (Equation 5.1):

$$H_{RSU} = U(FM(x)) + FM(x) \tag{5.1}$$

Where x is input, FM is Feature Map, and U is U-structure.

In order to extricate local features, a convolutional input layer provides the intermediate activation map $FM(x)$. The next component of the RSU block is the encoder decoder, which is similar to a U-Net and accepts $FM(x)$ as input. The multiscale contingent qualities $U(FM(x))$ are extracted and encoded. It extracts the multiscale features from gradually downsampled activation maps to reduce the loss during direct up sampling. By using incremental up sampling, concatenation, and convolution, it encodes them into high-aspiration activation maps.

The profusion connection, which mixes local characteristic with multiscale characteristics, is combined by residual connection. $FM(x) + U(FM(x)$.

The five-stage decoder in the second component of the U^2-Net design makes use of the dilated form of RSU.

Finally, combine the encoder and decoder phases to create saliency probability maps. Saliency maps display the special characteristics of each pixel. Using a saliency map, you can separate the fascinating content from the backdrop.

5.2.2 Classification Models

CNN consists of various layers, such as convolutional, pooling, and fully connected layers (Chen et al., 2020). The convolutional layer is an essential part of CNN since it

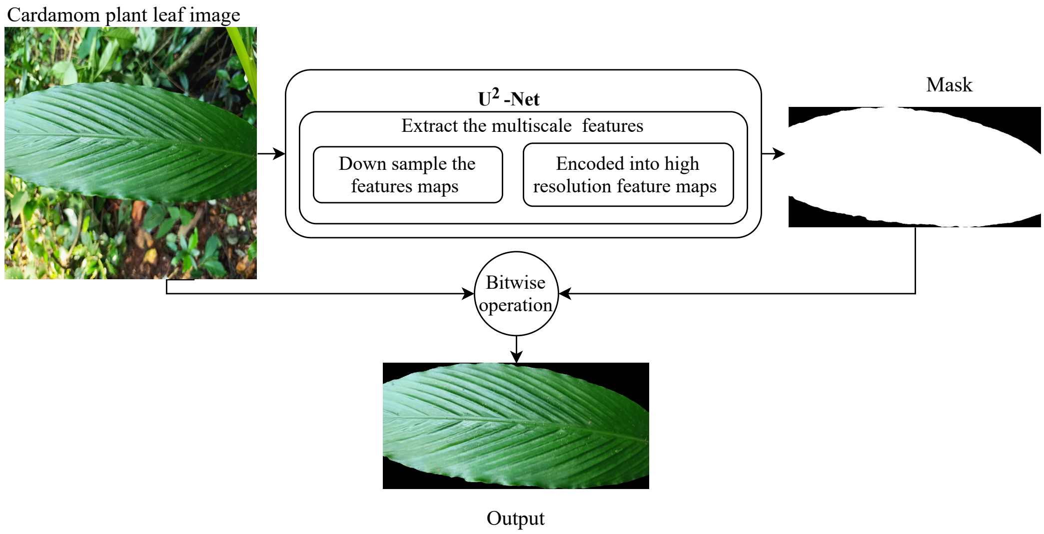


Figure 5.2: Background Removal by using U^2-Net

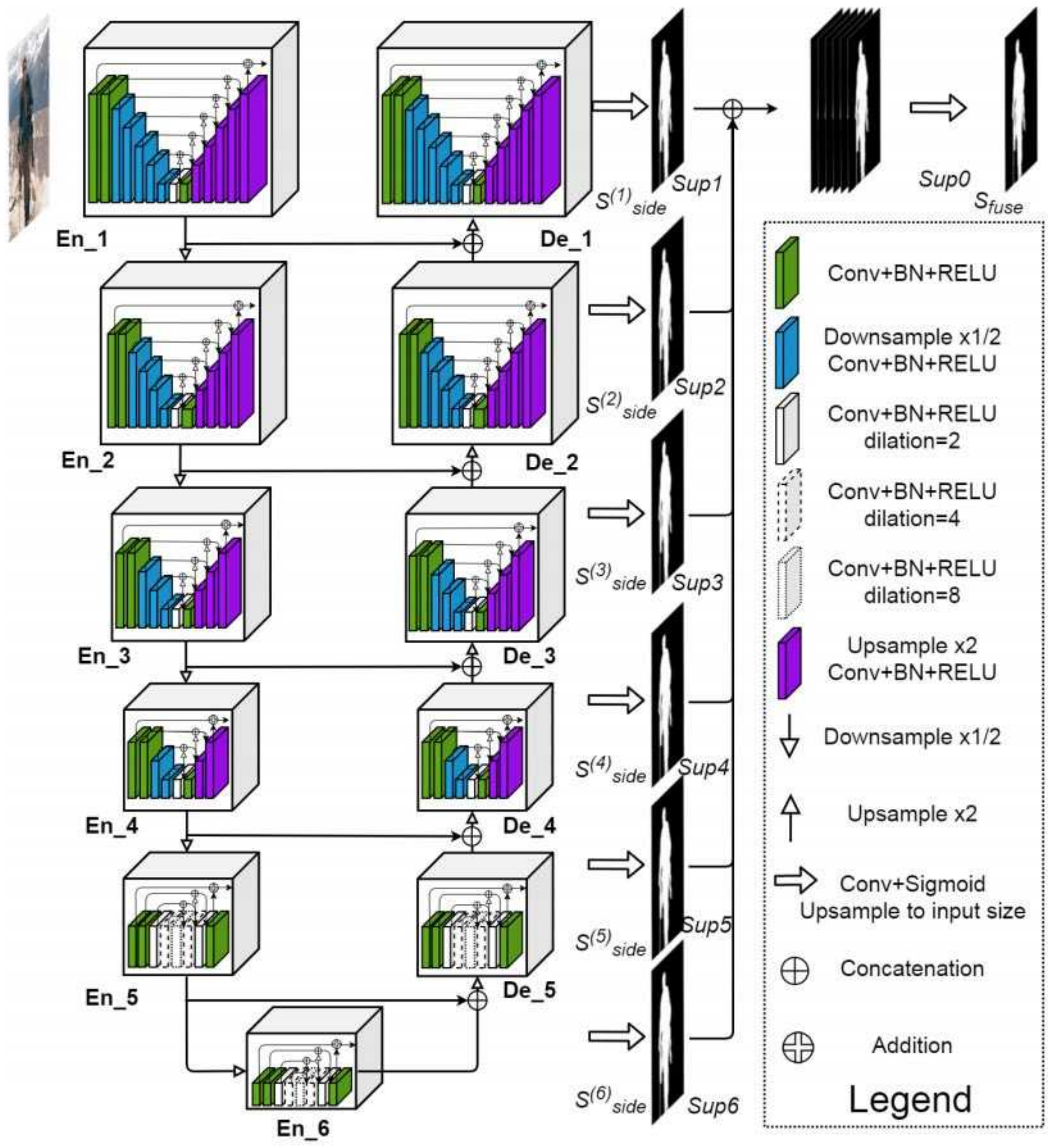


Figure 5.3: U^2-Net Architecture (Qin et al., 2020)

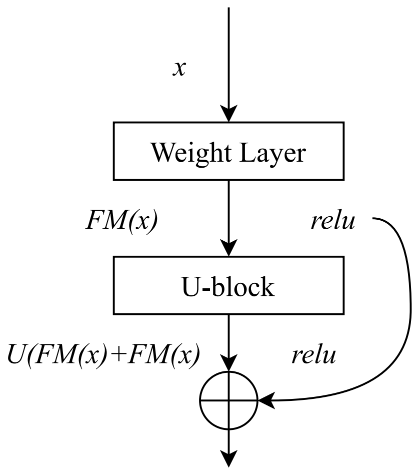


Figure 5.4: Residual U-block

extracts the detailed information of the input images using different convolution kernels. Several convolutional layers extract the set of feature maps known as color and edges of the input image. Feature map function is defined in Equation 5.2 (Chen et al., 2020).

$$FM_i = f(FM_{i-1}W_i + b_i) \tag{5.2}$$

FM denotes the Feature Map, W denotes the weight, b is the offset vector, and f(.) defines the ReLu activation function defined in Equation 5.3 (Cruz et al., 2019).

$$ReLu(z) = Max(0, z) \tag{5.3}$$

Where z is the input.

By reducing the spatial dimension and convolution, the pooling layer lowers the likelihood of overfitting. It is specified in Equation 5.4. (Chen et al., 2020).

$$y_i^l = down(y_i^{l-1}, s) \tag{5.4}$$

y_i^l indicates the feature vector, s defines the pooling size,

and down(.) indicates the down sampling.

Finally, the one or more fully connected layers defined, which flatten the network by connecting all the previous layer neurons, final fully connected layer predicts the class label, where Softmax activation function is used in the pre-trained models which

Table 5.2: Performance comparisons of the proposed approach.

Sl.No	Plant Name	Performance Metrics	CNN	EN	EN-V2-S	EN-V2-M	EN-V2-L
1	Cardamom	Accuracy (%)	91.30	94.10	95.59	88.44	98.26
		F1-Score	0.91	0.94	0.96	0.88	0.98
		Precision	0.91	0.94	0.96	0.88	0.98
		Recall	0.91	0.94	0.96	0.88	0.98
2	Grape	Accuracy (%)	94.24	97.81	96.44	93.72	96.45
		F1-Score	0.94	0.98	0.96	0.94	0.96
		Precision	0.94	0.98	0.96	0.95	0.96
		Recall	0.94	0.98	0.96	0.94	0.96
	EN-	EfficientNet					

are used in this work. Softmax is defined in the Equation 5.5 (Chen et al., 2020).

$$\text{Softmax}(y_i) = \frac{\exp(y_i)}{\sum_j \exp(y_j)} \tag{5.5}$$

Where y denotes input vector

$\exp(y_i)$ denotes the exponential function for input vector.

$\exp(y_j)$ denotes the exponential function for output vector.

In CNN, hyperparameters are chosen prior to training, whereas weights and biases are adjusted as the model is trained. Hyperparameters can be divided into two categories: those that address network structure and those that address training. The hyperparameters that deal with network structure include kernel sizes and the number of layers in the model; kernel size affects feature extraction on a wide scale, and deeper the layers had a greater classification rate. The hyperparameters batch size, learning rate, and dropout are related to training. Equation 5.6 represents the loss function.

EfficientNet is a family of CNNs that was proposed by Tan & Le (2019); it scales CNN parameters like as depth, or how many layers deep the network has, breadth, or resolution, or how high an image's resolution should be. EfficientNet increases the CNN's dimensionality by using a compound scaling algorithm. It scales up the baseline network, mobile inverted bottleneck convolution (MBConv), to become EfficientNet. Another member of the CNN family is EfficientNetV2, which has a lower learning curve and higher performance efficacy. It uses Fused-MBConv, which is quicker than previous models and up to 6.8x smaller, to improve training and efficacy (Tan & Le, 2021).

Table 5.3: Performance evaluation on external testing for cardamom plant leaf dataset

Sl.No	Category	Performance Metrics	CNN	EN	EN-V2-S	EN-V2-M	EN-V2-L
Cardamom plant dataset							
1	Colletotrichum Blight	Accuracy (%)	96.48	100	100	92.85	100
		F1-Score	0.98	0.98	1.00	0.95	0.98
		Precision	1.00	0.97	1.00	0.96	0.97
		Recall	0.96	1.00	1.00	0.93	1.00
2	Healthy	Accuracy (%)	98.76	98.76	98.76	97.53	98.76
		F1-Score	0.98	0.98	0.98	0.96	0.98
		Precision	0.98	.0.98	0.98	0.95	0.98
		Recall	0.98	0.99	0.99	0.98	0.99
3	Phyllosticta leaf spot	Accuracy (%)	98.48	96.96	96.96	96.96	96.96
		F1-Score	0.98	0.98	0.98	0.98	0.98
		Precision	0.98	100	0.98	0.98	1.00
		Recall	0.99	0.97	0.97	0.97	0.97
	EN-	EfficientNet					

Table 5.4: Performance evaluation on external testing for grape plant leaf dataset

Sl.No.	Category	Performance Metrics	CNN	EN	EN-S	EN-M	EN-L
Grape plant dataset							
1	Black rot	Accuracy (%)	100	97.45	97.45	97.45	96.61
		F1-Score	0.97	0.96	0.95	0.95	0.93
		Precision	0.94	0.95	0.93	0.93	0.90
		Recall	1.00	0.97	0.97	0.97	0.97
2	ESCA	Accuracy (%)	98.55	97.82	92.75	95.65	95.65
		F1-Score	0.99	0.98	0.95	0.97	0.95
		Precision	1.00	0.99	0.95	0.99	0.95
		Recall	0.99	0.98	0.93	0.96	0.96
3	Healthy	Accuracy (%)	100	100	97.61	97.61	100
		F1-Score	0.99	1.00	0.99	0.98	1.00
		Precision	0.98	100	1.00	0.98	1.00
		Recall	1.00	1.00	0.98	0.98	1.00
4	Leaf spot	Accuracy (%)	93.51	96.29	98.14	96.29	89.98
		F1-Score	0.97	0.97	0.98	0.97	0.94
		Precision	1.00	0.98	0.97	0.97	0.99
		Recall	0.94	0.96	0.98	0.96	0.90
	EN-	EfficientNet					

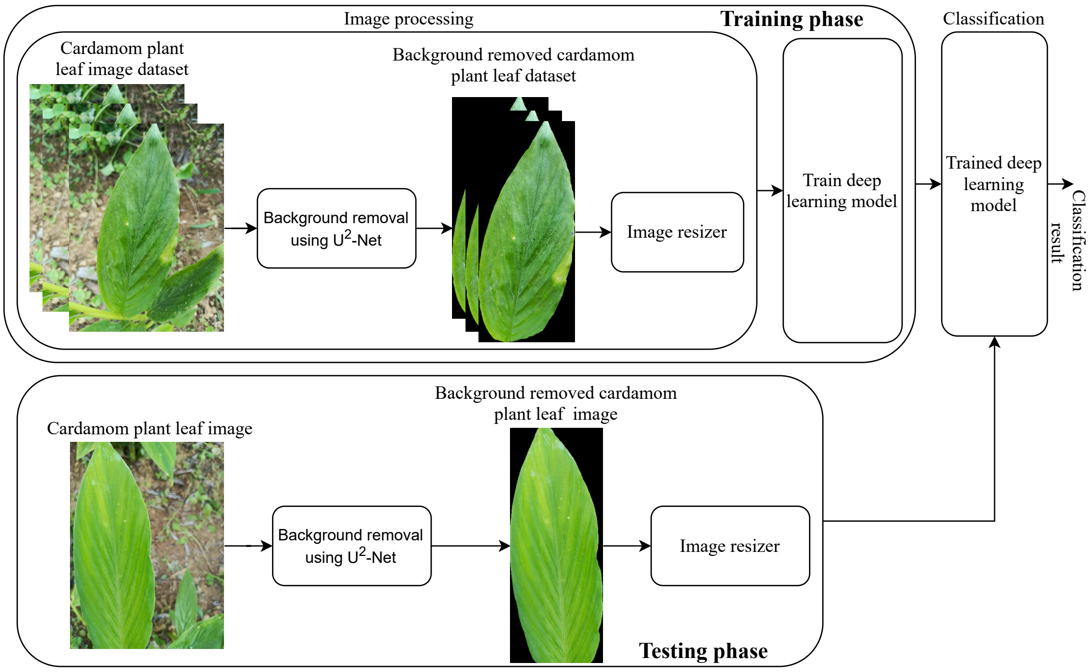


Figure 5.5: Proposed cardamom plant leaf disease detection approach

The proposed pipeline of cardamom plant leaf disease detection approach has four stages. The first stage depicts the dataset preparation; in this stage, collected the noisy cardamom plant leaf images from the cardamom plantation and labeled them. The second stage is used to remove the background of the leaf image and noise.

The third stage is training the deep learning-based model from scratch using the generated dataset. The next and final stage is the performance evaluation of the trained model. The proposed approach is shown in Figure 5.5. This has two phases; the first is, the training phase, which is used as the processing stage used to remove the complex background from the input image by using U^2-Net, background removed images are further processed to resize the images using an image resizer and fed into the next stage. The next stage is employed to train deep learning-based models such as CNN, EfficientNet, and EfficientNetV2. The different versions of EfficientNetV2 used in this work are EfficientNetV2-S (Small with 22 Million Parameters), EfficientNetV2-M (Medium with 54 Million Parameters), and EfficientNetV2-L (Large with 120 Million Parameters). Finally, the trained model produces the classification results. In the testing phase, the cardamom plant leaf image is fed to a trained deep learning-based model after completion of pre-processing operation such as background removal and resizing the image, the trained deep learning-based model produce the classification results.

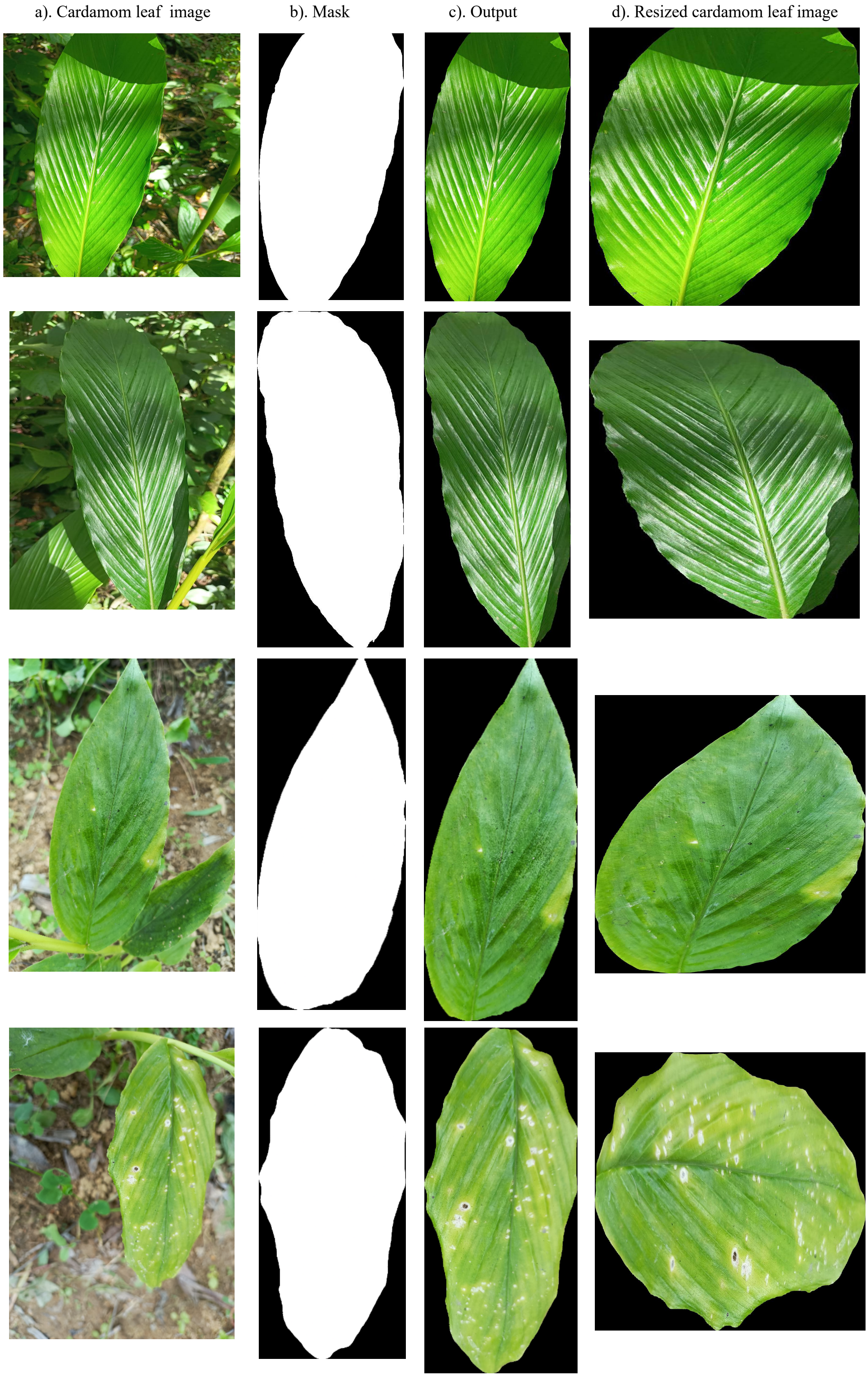


Figure 5.6: Background removal of cardamom plant leaf images: a). Cardamom plant leaf images, b). Mask, c). Output, d). Resized cardamom plant leaf images.

$$L(M) = -\frac{1}{n}\sum_{Y_i=1}^{n}\sum_{c=1}^{C}[Y_{ic}logP(x_i = c) + (1 - Y_{ic})log(1 - P(Y_i = c))] \tag{5.6}$$

where M is the weight matrix, n is training samples, C represents class labels, and P is the predicted probability.

Table 5.5: Cross fold validation results (Accuracy (%))

Sl.No	Plant Name	CNN	EfficientNet	EfficientNetV2-L
1	Cardamom	87.59	87.02	91.42
2	Grape	94.60	96.00	94.62

5.2.3 Experimental Results

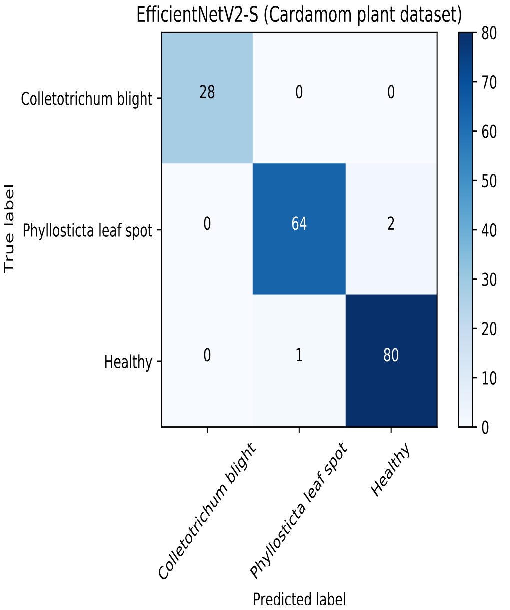
As discussed in Section 5.1 all the cardamom plant leaf images are captured in complex background, and all of them are of different dimensions; to remove the complex background from the image U^2-Net is used (Qin et al., 2020). Figure 5.6 describes the background removal of cardamom plant leaf images using U^2-Net. Figure 5.6 a) shows the original cardamom plant leaf images, Figure 5.6 b) shows the mask generated by the U^2-Net, and Figure 5.6 c). Output generated by the Background removal approach, and Figure 5.6 d) shows the resized images using an image resizer.

Accuracy (Equation 3.3), Precision (Positive predictive value) (Equation 3.5), Recall (sensitivity) (Equation 3.6), and F1-Score(balanced F-score) (Equation 3.4) were used as the performance metrics in this experiential study (Chouhan et al. (2021b); Zhao et al. (2022b); Qi et al. (2022b)) .

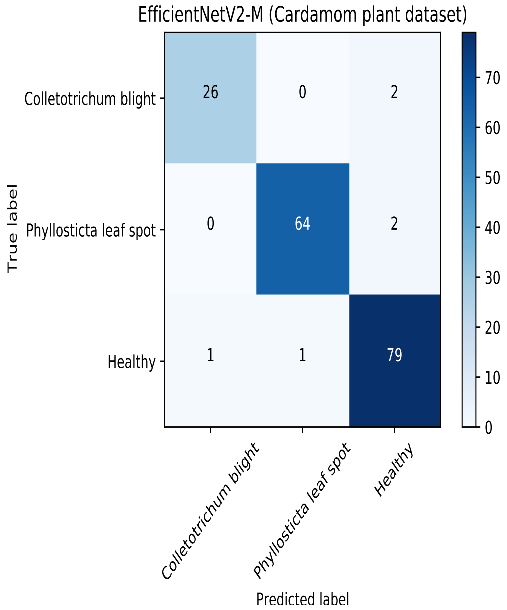
All the original input images are resized to 224×224 for all the three deep learning-based models used in this study. In the experiment, 90% of the dataset was used for training and 10% for testing. A set of experiments was conducted to measure the performance of the proposed approach for 100 epochs on the cardamom plant dataset. The same set of experiments was conducted by using a publicly available grape dataset to measure the performance of the proposed approach. Further, another set of experiments was conducted by using other deep learning models such as CNN and EfficientNet.The proposed approach's performance evaluation is shown in Table 5.2. On the cardamom plant dataset, CNN achieves a maximum detection accuracy of 91.30%, while on the grape dataset, it achieves a maximum detection accuracy of 94.24%. On the cardamom and grape plant datasets, EfficientNet achieved a maximum detection accuracy of 94.10% and 97.81%, respectively. On the cardamom and grape plant datasets,

Table 5.6: Comparison of the proposed model with state-of-the-art methods

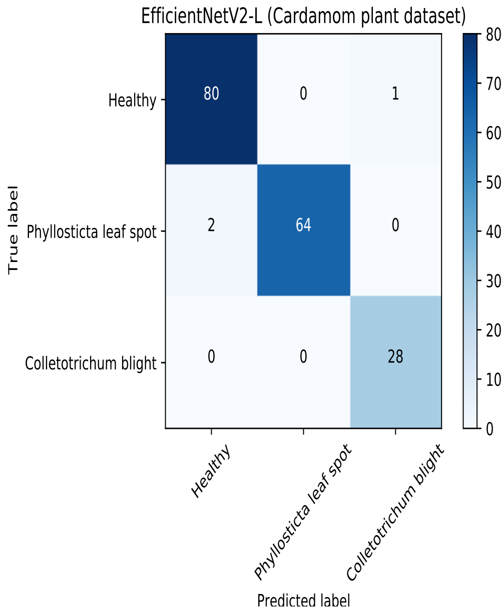
Sl.No	Plant	Approach	Accuracy (%)
1	Coffee (Manso et al., 2019)	Artificial Neural Network	95.8
2	Cucumber (Zhang et al., 2017)	K-means and sparse representation	85.7
3	Multiple (Singh & Misra, 2017)	Genetic Algorithm	96.7
4	Citrus (Pourazar et al., 2019)	Vision sensor and SVM	97
5	Multiple (Hang et al., 2019)	VGG16 and InceptionNet	91.7
6	Maize (Sibiya & Sumbwanyambe, 2019)	CNN	92.85
7	Strawberry (Shin et al., 2021a)	SqueezeNet	92.61
8	Tomato (Karthik et al., 2020)	CNN	98
9	Cucumber (Zhang et al., 2019)	CNN	95.18
10	Multiple (Ma et al., 2018a)	DCNN	93.4
11	Coffee (Esgario et al., 2020)	ResNet50	95.24
12	Soybean (Wu et al., 2020b)	ResNet	94.29
13	Apple (Zhong & Zhao, 2020)	DenseNet121	92.29
14	Paddy (Jiang et al., 2021a)	VGG16	97.22
15	Coconut (Singh et al., 2021)	MobileNet	82.10
Proposed Model			
16	Cardamom	EfficientNetV2	98.26
17	Grape	EfficientNetV2	96.44



(a) EfficientNetV2-S

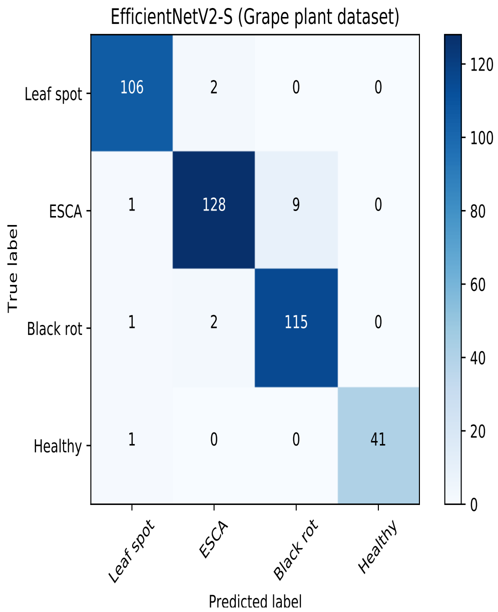


(b) EfficientNetV2-M

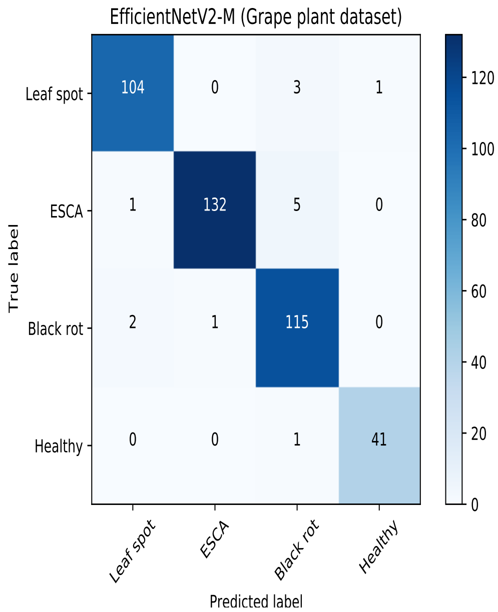


(c) EfficientNetV2-L

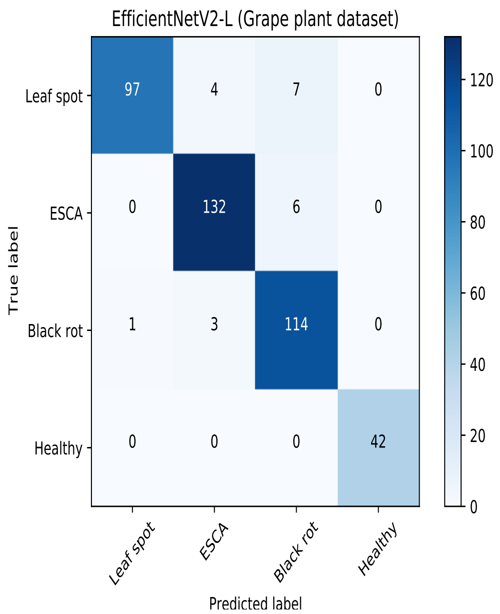
Figure 5.7: Confusion matrix for cardamom plant dataset (external testing): a) EfficientNetV2-S model, b) EfficientNetV2-M model, c) EfficientNetV2-L model



(a) EfficientNetV2-S



(b) EfficientNetV2-M



(c) EfficientNetV2-L

Figure 5.8: Confusion matrix for grape plant dataset (external testing): a) EfficientNetV2-S model, b) EfficientNetV2-M model, c) EfficientNetV2-L model

EfficientNetV2-S achieved a maximum detection accuracy of 95.59% and 96.44%, respectively. On the cardamom and grape plant datasets, EfficientNetV2-M achieved a maximum detection accuracy of 94.10% and 97.81%, respectively. On the cardamom and grape plant datasets, EfficientNetV2-L achieved a maximum detection accuracy of 98.26% and 96.45%, respectively.

The two types of testing are generally internal and external testing. Internal testing divides the dataset into a train set and a test set. The training set is used to train the model, and the test is used to assess the model's performance in terms of accuracy, precision, and recall. The trained model is tested externally using a dataset that was obtained independently, i.e., a dataset that was not used for training or internal testing. In order to demonstrate the generalizability of the model, external testing is a task that requires the trained model to use other datasets that are distinct from the original dataset used to train the model.

To understand the behavior of the trained models, external testing is essential; this examine the dialects learned during training, and this helps to measure the performance of the trained models. A set of external testing was conducted using trained models such as CNN, EfficientNet, and EfficinetNetV2 models on cardamom plant and grape plant dataset. Table 5.3 describes the performance evaluation on external testing for the cardamom plant dataset. In the external testing, EfficientNetV2-S outperformed compared to other models for the cardamom plant dataset. Figure 5.7 shows the confusion matrix for external testing on trained EfficientNetV2 models for the cardamom plant dataset. Table 5.4 describes the performance evaluation on external testing for grape plant dataset. In the external testing. Figure 5.8 shows the confusion matrix for external testing on trained EfficientNetV2 models for the grape plant dataset.

A set of experiments was conducted with cross-fold validation on CNN, EfficientNet, and EfficientNetV2-L models; Table 5.5 describes the cross-fold validation with 100 epochs for each fold on cardamom and grape plant leaf datasets. The efficientNetV2-L model outperforms compared to the other two models for the cardamom plant dataset with 91.42% detection accuracy.

A wide set of experiments was conducted on CNN, EfficientNet, and EfficientNetV2 models, and the results shown in Table 5.2 to Table 5.4 depicts that the EfficientNetV2 model outperforms as compared with other models and attained almost consistent result on grape plant dataset, and EfficientNetV2-L attained a maximum result of 98.26% cardamom plant dataset. Table 5.6 shows the comparison of the proposed approach with state-of-the-art methods. Employing U^2-Net with EfficientNetV2 outperforms cardamom plant leaf disease detection with 98.26% detection accuracy.

5.3 Summary

To make it near real-time complex background images to be processed and handled. In this regard, this work focused on leaf images taken in complex environments such as noisy images and with background. Using U^2-Net obtain the enhanced leaf images without background and further applied a EfficientNetV2 for classification of the cardamom plant leaf images. This work overcame the challenge of complex background images for plant leaf disease classification, and outperformed compared to state-of-the-art methods, and achieved 98.28% (EfficientNetV2-S model) detection accuracy for the cardamom plant dataset on external testing.

This study shows the importance of the background removal of the images taken on farmland, which is essential since the images taken in farmland contain unwanted background, which deteriorates the results due to the foreign objects. After removing the background and noise from the dataset or images, a classification model is applied to classify the images. In this study, EfficinetNetV2 is used, and obtain the result as 98.28% detection accuracy.

Chapter 6

Conclusions and Future Work

Plant diseases are a significant threat to crop quality and yield. It is indispensable to detect the disease on time with a robust, low-cost plant disease detection approach and suggest appropriate control measures.

6.1 Conclutions

Ensemble DL-based plant leaf disease detection approach is proposed. It addressed the challenges such as class miss-classification, image capturing conditions, and classification time. AlexNet, ResNet50, and VGG16 were used as base models in the proposed approach since each base models are unique in nature to classify the images, and each model extracts independent features. The main objective of the proposed approach is to minimize the miss-classification rate; this is achieved by employing aforesaid three different DL-based models; the final classification outcome is based on the majority of the classifier's outcomes. In this work, external testing time is also considered for binary and multiclass datasets. The Ensemble DL-based approach outperformed the state-of-the-art methods by attaining maximum detection accuracy of 100% for the binary class dataset and 99.53% for the multi-class dataset.

Tomato plant disease classification is proposed by using MFFN with ACSPAM. A Series of experiments are conducted to measure the robustness of the proposed tomato plant disease classification. In the classification of the diseases, the diseases with similar symptoms and class-imbalanced data lead to poor performance of the classifier. To contemplate this, an ACSPAM with MFFN is used. It attained the best validation accuracy of 99.88% and testing accuracy of 99.5% compared to state-of-the-art methods.

An efficient plant leaf disease detection approach is essential to detect plant diseases with noisy images and complex backgrounds. In this regard, the cardamom plant leaf disease detection approach is proposed, where the cardamom plant leaf dataset was collected from farmland with a complex background. Segmenting and detecting diseases in real-time images is a challenging task, as the images are associated with other factors, such as the background of the image, environmental factors, such as lighting, and the angle of the capturing conditions. In the proposed method, the U^2-Net architecture is employed to remove the complex background, which produces results without deteriorating the quality of the original image. For classification, in this work, CNN, EfficientNet, and EfficientNetV2 models were trained instead of using the pre-trained weights for EfficientNet, and EfficientNetV2. EfficientNetV2-S and EfficientNetV2-L models outperformed the other models; EfficientNetV2-L achieved 98.26% detection accuracy for the cardamom plant dataset, and EfficientNetV2-S achieved 98.28% detection accuracy for the cardamom plant dataset on external testing.

To generalize to the specific model, Cardamom and Grape datasets are used in all three approaches; all the approaches produce promising results, while ACSPAM-MLFF performed better compared to all the models as observed with different datasets and U^2-Net is essential to remove the background and noise in the images taken in the farmland.

6.2 Future work

There are ample of opportunities to work on plant disease detection.

- Can be extended to detect plant nutrition deficiency.
- Can be strengthened to identify the plant disease's severity, such as the early/middle/final stage of the plant disease.
- Can be extended further to train the classification model such that it works for real-time images and provide the prescriptions in real-time.
- Can be deployed in the cloud-computing environment so that farmers can use it irrespective of time and place.
- This work can also be extended to detect the weeds in the agricultural lands and control measures.

References

Abbas, A., Jain, S., Gour, M., & Vankudothu, S. (2021a). Tomato plant disease detection using transfer learning with c-gan synthetic images. *Computers and Electronics in Agriculture*, *187*, 106279.

Abbas, A., Jain, S., Gour, M., & Vankudothu, S. (2021b). Tomato plant disease detection using transfer learning with c-gan synthetic images. *Computers and Electronics in Agriculture*, *187*, 106279.

Abdu, A. M., Mokji, M. M., & Sheikh, U. U. (2020). Automatic vegetable disease identification approach using individual lesion features. *Computers and Electronics in Agriculture*, *176*, 105660.

Abed, S. H., Al-Waisy, A. S., Mohammed, H. J., & Al-Fahdawi, S. (2021). A modern deep learning framework in robot vision for automated bean leaves diseases detection. *International Journal of Intelligent Robotics and Applications*, *5*(2), 235–251.

Aditya, S. (2022). Understanding the malnutrition crisis in india. *Feeding India*. https://www.feedingindia.org/blog/understanding-the-malnutrition-crisis-in-india/.

Ahila, P. R., Arivazhagan, S., Arun, M., & Mirnalini, A. (2019). Maize leaf disease classification using deep convolutional neural networks. *Neural Computing and Applications*, *31*, 8887—-8895.

Ahmad, M., Abdullah, M., Moon, H., & Han, D. (2021). Plant disease detection in imbalanced datasets using efficient convolutional neural networks with stepwise transfer learning. *IEEE Access*, *9*, 140565–140580.

Alguliyev, R., Imamverdiyev, Y., Sukhostat, L., & Bayramov, R. (2021). Plant disease detection based on a deep model. *Soft Computing*, *25*(21), 13229–13242.

Amin, H., Darwish, A., Hassanien, A. E., & Soliman, M. (2022). End-to-end deep learning model for corn leaf disease classification. *IEEE Access*, *10*, 31103–31115.

Andrushia, A. D. & Patricia, A. T. (2020). Artificial bee colony optimization (abc) for grape leaves disease detection. *Evolving Systems*, *11*(1), 105–117.

Aravind, K. R. & Raja, P. (2020). Automated disease classification in (selected) agricultural crops using transfer learning. *Automatika: Journal for Control, Measurement, Electronics, Computing & Communications*, *61*(2).

Arun, A. R. & Umamaheswari, S. (2022). Effective multi-crop disease detection using pruned complete concatenated deep learning model. *Expert Systems with Applications*.

Ashwinkumar, S., Rajagopal, S., Manimaran, V., & Jegajothi, B. (2022). Automated plant leaf disease detection and classification using optimal mobilenet based convolutional neural networks. *Materials Today: Proceedings*, *51*, 480–487.

Astani, M., Hasheminejad, M., & Vaghefi, M. (2022). A diverse ensemble classifier for tomato disease recognition. *Computers and Electronics in Agriculture*, *198*, 107054.

Bagheri, N., Mohamadi-Monavar, H., Azizi, A., & Ghasemi, A. (2018). Detection of fire blight disease in pear trees by hyperspectral data. *European Journal of Remote Sensing*, *51*(1), 1–10.

Barbedo, J. G. A. (2019). Plant disease identification from individual lesions and spots using deep learning. *Biosystems Engineering*, *180*, 96–107.

Barburiceanu, S., Meza, S., Orza, B., Malutan, R., & Terebes, R. (2021). Convolutional neural networks for texture feature extraction. applications to leaf disease classification in precision agriculture. *IEEE Access*, *9*, 160085–160103.

Bayer, G. & Seminis. (2021). Agronomic spotlight target spot of tomato. https://www.vegetables.bayer.com/ca/en-ca/resources/agronomic-spotlights/target-spot-of-tomato.html.

Brahimi, M., Boukhalfa, K., & Moussaoui, A. (2017). Deep learning for tomato diseases: classification and symptoms visualization. *Applied Artificial Intelligence*, *31*(4), 299–315.

Bravo, C., Moshou, D., Oberti, R., West, J., McCartney, A., Bodria, L., & Ramon, H. (2004). Foliar disease detection in the field using optical sensor fusion. *Agricultural Engineering International: the CIGR Journal of Scientific Research and Development*, *6*. Manuscript FP 04 008.

Chen, J., Chen, J., Zhang, D., Nanehkaran, Y. A., & Sun, Y. (2021a). A cognitive vision method for the detection of plant disease images. *Machine Vision and Applications*, *32*(1), 1–18.

Chen, J., Chen, J., Zhang, D., Nanehkaran, Y. A., & Sun, Y. (2021b). A cognitive vision method for the detection of plant disease images. *Machine Vision and Applications*, *32*(1), 1–18.

Chen, J., Chen, J., Zhang, D., Sun, Y., & Nanehkaran, Y. A. (2020). Using deep transfer learning for image-based plant disease identification. *Computers and Electronics in Agriculture*, *173*, 105393.

Chen, X., Zhou, G., Chen, A., Yi, J., Zhang, W., & Hu, Y. (2020a). Identification of tomato leaf diseases based on combination of abck-bwtr and b-arnet. *Computers and Electronics in Agriculture*, *178*, 105730.

Chen, X., Zhou, G., Chen, A., Yi, J., Zhang, W., & Hu, Y. (2020b). Identification of tomato leaf diseases based on combination of abck-bwtr and b-arnet. *Computers and Electronics in Agriculture*, *178*, 105730.

Chouhan, S. S., Singh, U. P., & Jain, S. (2021a). Automated plant leaf disease detection and classification using fuzzy based function network. *Wireless Personal Communications*, *121*(3), 1757–1779.

Chouhan, S. S., Singh, U. P., & Jain, S. (2021b). Automated plant leaf disease detection and classification using fuzzy based function network. *Wireless Personal Communications*, *121*(3), 1757–1779.

Chug, A., Bhatia, A., Singh, A. P., & Singh, D. (2022). A novel framework for image-based plant disease detection using hybrid deep learning approach. *Soft Computing*, 1–26.

Coulibaly, S., Kamsu-Foguem, B., Kamissoko, D., & Traore, D. (2019). Deep neural networks with transfer learning in millet crop images. *Computers in Industry*, *108*, 115–120.

Cristin, R., Kumar, B. S., Priya, C., & Karthick, K. (2020). Deep neural network based rider-cuckoo search algorithm for plant disease detection. *Artificial intelligence review*, *53*(7), 4993–5018.

Cruz, A., Ampatzidis, Y., Pierro, R., Materazzi, A., Panattoni, A., De Bellis, L., & Luvisi, A. (2019). Detection of grapevine yellows symptoms in vitis vinifera l. with artificial intelligence. *Computers and electronics in agriculture*, *157*, 63–76. https://github.com/DrAlbertCruz/Salento-Grapevine-Yellows-Dataset.

Deepa, N. & Nagarajan, N. (2021). Kuan noise filter with hough transformation based reweighted linear program boost classification for plant leaf disease detection. *Journal of Ambient Intelligence and Humanized Computing*, 1–14.

Delnevo, G., Girau, R., Ceccarini, C., & Prandi, C. (2021). A deep learning and social iot approach for plants disease prediction toward a sustainable agriculture. *IEEE Internet of Things Journal*, *9*(10), 7243–7250.

Deng, L. & Yu, D. (2014). Deep learning–methods and applications. *Foundations and Trends in Signal Processing*, *7*(3-4), 197–387.

Dyrmann, M., Karstoft, H., & Midtiby, H. S. (2016). Plant species classification using deep convolutional neural network. *Biosystems engineering*, *151*, 72–80.

Ebrahimi, M., Khoshtaghaza, M. H., Minaei, S., & Jamshidi, B. (2017). Vision-based pest detection based on svm classification method. *Computers and Electronics in Agriculture*, *137*, 52–58.

Elfatimi, E., Eryigit, R., & Elfatimi, L. (2022a). Beans leaf diseases classification using mobilenet models. *IEEE Access*, *10*, 9471–9482.

Elfatimi, E., Eryigit, R., & Elfatimi, L. (2022b). Beans leaf diseases classification using mobilenet models. *IEEE Access*.

Esgario, J. G., Krohling, R. A., & Ventura, J. A. (2020). Deep learning for classification and severity estimation of coffee leaf biotic stress. *Computers and Electronics in Agriculture*, *169*, 105162. https://github.com/esgario/lara2018.

Faithpraise, F., Birch, P., Young, R., Obu, J., Faithpraise, B., & Chatwin, C. (2013). Automatic plant pest detection and recognition using k-means clustering algorithm and correspondence filters. *International Journal of Advanced Biotechnology and Research*, *4*(2), 189–199.

Fan, X., Luo, P., Mu, Y., Zhou, R., Tjahjadi, T., & Ren, Y. (2022). Leaf image based plant disease identification using transfer learning and feature fusion. *Computers and Electronics in Agriculture*, *196*, 106892.

Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. *Computers and Electronics in Agriculture*, *145*, 311–318.

Furlanetto, R. H., Nanni, M. R., Mizuno, M. S., Crusiol, L. G. T., & da Silva, C. R. (2021). Identification and classification of asian soybean rust using leaf-based hyperspectral reflectance. *International Journal of Remote Sensing*, *42*(11), 4177–4198.

G, G. & Arun, P. (2019). Identification of plant leaf diseases using a nine-layer deep convolutional neural network. *Computers & Electrical Engineering*, *76*, 323–338.

Gajjar, R., Gajjar, N., Thakor, V. J., Patel, N. P., & Ruparelia, S. (2021). Real-time detection and identification of plant leaf diseases using convolutional neural networks on an embedded platform. *The Visual Computer*, 1–16.

Gokulnath, B. & Gandhi, U. D. (2021). Regularized deep clustering approach for effective categorization of maize diseases. *Journal of Ambient Intelligence and Humanized Computing*, 1–10.

Gutiérrez, S., Hernández, I., Ceballos, S., Barrio, I., Díez-Navajas, A. M., & Tardaguila, J. (2021). Deep learning for the differentiation of downy mildew and spider mite in grapevine under field conditions. *Computers and Electronics in Agriculture*, *182*, 105991.

Haider, W., Rehman, A.-U., Durrani, N. M., & Rehman, S. U. (2021). A generic approach for wheat disease classification and verification using expert opinion for knowledge-based decisions. *IEEE Access*, *9*, 31104–31129.

Hang, J., Zhang, D., Chen, P., Zhang, J., & Wang, B. (2019). Classification of plant leaf diseases based on improved convolutional neural network. *Sensors*, *19*(19), 4161.

Hassan, S. M. & Maji, A. K. (2022). Plant disease identification using a novel convolutional neural network. *IEEE Access*, *10*, 5390–5401.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Identity mappings in deep residual networks. In *European conference on computer vision*, 630–645. Springer.

Hu, G., Wu, H., Zhang, Y., & Wan, M. (2019). A low shot learning method for tea leaf's disease identification. *Computers and Electronics in Agriculture*, *163*, 104852.

Hua, S., Xu, M., Xu, Z., Ye, H., & Zhou, C. (2022). Multi-feature decision fusion algorithm for disease detection on crop surface based on machine vision. *Neural Computing and Applications*, *34*(12), 9471–9484.

Hughes, D. & Salathé, M. (2015). An open access repository of images on plant health to enable the development of mobile disease diagnostics. *https://arxiv.org/abs/1511.08060*. https://github.com/spMohanty/PlantVillage-Dataset.

Jain, S. & Dharavath, R. (2021). Memetic salp swarm optimization algorithm based feature selection approach for crop disease detection system. *Journal of Ambient Intelligence and Humanized Computing*, 1–19.

Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, *31*(3), 685–695.

Ji, M. & Wu, Z. (2022). Automatic detection and severity analysis of grape black measles disease based on deep learning and fuzzy logic. *Computers and Electronics in Agriculture*, *193*, 106718.

Ji, M., Zhang, L., & Wu, Q. (2020). Automatic grape leaf diseases identification via unitedmodel based on multiple convolutional neural networks. *Information Processing in Agriculture*, *7*(3), 418–426.

Jiang, P., Chen, Y., Liu, B., He, D., & Liang, C. (2019). Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. *IEEE Access*, *7*, 59069–59080.

Jiang, Z., Dong, Z., Jiang, W., & Yang, Y. (2021a). Recognition of rice leaf diseases and wheat leaf diseases based on multi-task deep transfer learning. *Computers and Electronics in Agriculture*, *186*, 106184.

Jiang, Z., Dong, Z., Jiang, W., & Yang, Y. (2021b). Recognition of rice leaf diseases and wheat leaf diseases based on multi-task deep transfer learning. *Computers and Electronics in Agriculture*, *186*, 106184.

Joshi, P., Das, D., Udutalapally, V., Pradhan, M. K., & Misra, S. (2022). Ricebios: Identification of biotic stress in rice crops using edge-as-a-service. *IEEE Sensors Journal*, *22*(5), 4616–4624.

Kamal, K., Yin, Z., Wu, M., & Wu, Z. (2019). Depthwise separable convolution architectures for plant disease classification. *Computers and Electronics in Agriculture*, *165*, 104948.

Karthik, R., Hariharan, M., Anand, S., Mathikshara, P., Johnson, A., & Menaka, R. (2020). Attention embedded residual cnn for disease detection in tomato leaves. *Applied Soft Computing*, *86*, 105933.

Kendler, S., Aharoni, R., Young, S., Sela, H., Kis-Papo, T., Fahima, T., & Fishbain, B. (2022). Detection of crop diseases using enhanced variability imagery data and convolutional neural networks. *Computers and Electronics in Agriculture*, *193*, 106732.

Kerkech, M., Hafiane, A., & Canals, R. (2018). Deep leaning approach with colorimetric spaces and vegetation indices for vine diseases detection in uav images. *Computers and electronics in agriculture*, *155*, 237–243.

Khamparia, A., Saini, G., Gupta, D., Khanna, A., Tiwari, S., & de Albuquerque, V. H. C. (2019). Seasonal crops disease prediction and classification using deep convolutional encoder network. *Circuits, Systems, and Signal Processing*, *39*, 818—-836.

Khamparia, A., Singh, A., Luhach, A. K., Pandey, B., & Pandey, D. K. (2020). Classification and identification of primitive kharif crops using supervised deep convolutional networks. *Sustainable Computing: Informatics and Systems*, *28*, 100340.

Khan, A. I., Quadri, S., Banday, S., & Shah, J. L. (2022). Deep diagnosis: A real-time apple leaf disease detection system based on deep learning. *Computers and Electronics in Agriculture*, *198*, 107093.

Kour, V. P. & Arora, S. (2019). Particle swarm optimization based support vector machine (p-svm) for the segmentation and classification of plants. *IEEE Access*, *7*, 29374–29385.

Koushik, N., Jones, S., Singh, A. K., Sarkar, S., Singh, A., & Ganapathysubramanian, B. (2019). Plant disease identification using explainable 3d deep learning on hyperspectral images. *Plant Methods*, *15*(1), 98.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Commun. ACM*, *60*(6), 84–90.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84–90.

Kumar, M., Kumar, A., & Palaparthy, V. S. (2020). Soil sensors-based prediction system for plant diseases using exploratory data analysis and machine learning. *IEEE Sensors Journal*, *21*(16), 17455–17468.

LeCun, Y. (2015). Lenet-5, convolutional neural networks. *URL: http://yann. lecun. com/exdb/lenet*, *20*(5), 14.

Li, M., Zhou, G., Chen, A., Yi, J., Lu, C., He, M., & Hu, Y. (2022a). Fwdgan-based data augmentation for tomato leaf disease identification. *Computers and Electronics in Agriculture*, *194*, 106779.

Li, M., Zhou, G., Chen, A., Yi, J., Lu, C., He, M., & Hu, Y. (2022b). Fwdgan-based data augmentation for tomato leaf disease identification. *Computers and Electronics in Agriculture*, *194*, 106779.

Liang, Q., Xiang, S., Hu, Y., Coppola, G., Zhang, D., & Sun, W. (2019). Pd2se-net: Computer-assisted plant disease diagnosis and severity estimation network. *Computers and electronics in agriculture*, *157*, 518–529.

Lincy, B. R. J. (2021). Detection of plant leaf diseases using recent progress in deep learning-based identification techniques. *Annals of Computer Vision Impelled Technology*.

Liu, J. & Wang, X. (2020). Tomato diseases and pests detection based on improved yolo v3 convolutional neural network. *Frontiers in plant science*, *11*, 898.

Liu, X., Min, W., Mei, S., Wang, L., & Jiang, S. (2021). Plant disease recognition: A large-scale benchmark dataset and a visual region and loss reweighting approach. *IEEE Transactions on Image Processing*, *30*, 2003–2015.

Lu, J., Hu, J., Zhao, G., Mei, F., & Zhang, C. (2017). An in-field automatic wheat disease diagnosis system. *Computers and electronics in agriculture*, *142*, 369–379.

Lu, Y., Yi, S., Zeng, N., Liu, Y., & Zhang, Y. (2017). Identification of rice diseases using deep convolutional neural networks. *Neurocomputing*, *267*, 378–384.

M, Metin, O. & Adem, K. (2019). Automatic detection and classification of leaf spot disease in sugar beet using deep learning algorithms. *Physica A: Statistical Mechanics and its Applications*, *535*, 122537.

M, P. R., Arthur, K., C, S. M., & J, P. M. (2021). Plant disease. *Encyclopedia Britannica*. https://www.britannica.com/science/plant-disease.

Ma, J., Du, K., Zheng, F., Zhang, L., Gong, Z., & Sun, Z. (2018a). A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network. *Computers and electronics in agriculture*, *154*, 18–24.

Ma, J., Du, K., Zheng, F., Zhang, L., Gong, Z., & Sun, Z. (2018b). A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network. *Computers and electronics in agriculture*, *154*, 18–24.

Mahlein, A., Kuska, M. T., Behmann, J., Polder, G., & Walter, A. (2018). Hyperspectral sensors and imaging technologies in phytopathology: state of the art. *Annual review of phytopathology*, *56*, 535–558.

Manju, M., Mushrif, S. K., Shankarappa, T., Hegde, L., Gowda, A., Lokesh, M., & Naik, N. (2018). Evaluation of new fungicides against the leaf blight disease of cardamom [elettaria cardamomum maton] in arecanut based intercropping system. *J. Pharmacognosy Phytochem.*, 283–285.

Manso, G. L., Knidel, H., Krohling, R. A., & Ventura, J. A. (2019). A smartphone application to detection and classification of coffee leaf miner and coffee leaf rust. *arXiv preprint arXiv:1904.00742*.

Marissa, S., Anna, J., Michelle, G., & Angela, O. (2021). Tomato leaf mold. *https://extension.umn.edu/disease-management/tomato-leaf-mold*.

Mathew, M. P. & Mahesh, T. Y. (2022). Leaf-based disease detection in bell pepper plant using yolo v5. *Signal, Image and Video Processing*, *16*(3), 841–847.

Mia, M. R., Roy, S., Das, S. K., & Rahman, M. A. (2020). Mango leaf disease recognition using neural network and support vector machine. *Iran Journal of Computer Science*, *3*(3), 185–193.

Mishra, M., Choudhury, P., & Pati, B. (2021). Modified ride-nn optimizer for the iot based plant disease detection. *Journal of Ambient Intelligence and Humanized Computing*, *12*, 691–703.

Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016a). Using deep learning for image-based plant disease detection. *Frontiers in plant science*, *7*, 1419.

Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016b). Using deep learning for image-based plant disease detection. *Frontiers in plant science*, *7*, 1419.

Moshou, D., Bravo, C., Oberti, R., West, J., Bodria, L., McCartney, A., & Ramon, H. (2005). Plant disease detection based on data fusion of hyper-spectral and multi-spectral fluorescence imaging using kohonen maps. *Real-Time Imaging*, *11*(2), 75–83.

Nagasubramanian, G., Sakthivel, R. K., Patan, R., Sankayya, M., Daneshmand, M., & Gandomi, A. H. (2021). Ensemble classification and iot-based pattern recognition for crop disease monitoring system. *IEEE Internet of Things Journal*, *8*(16), 12847–12854.

Nandhini, M., Kala, K., Thangadarshini, M., & Verma, S. M. (2022). Deep learning model of sequential image classifier for crop disease detection in plantain tree cultivation. *Computers and Electronics in Agriculture*, *197*, 106915.

Nanehkaran, Y., Zhang, D., Chen, J., Tian, Y., & Al-Nabhan, N. (2020). Recognition of plant leaf diseases based on computer vision. *Journal of Ambient Intelligence and Humanized Computing*, 1–18.

Nations, U. (2015). World population prospects: The 2015 revision. *United Nations Econ Soc Aff*. https://www.un.org/en/development/desa/publications/world-population-prospects-2015-revision.html.

NIPHM, G. o. I. (2014). Aesa based ipm package for tomato. *National Institute of Plant Health Management, Rajendranagar, Hyderabad* ....

Park, K., ki Hong, Y., hwan Kim, G., & Lee, J. (2018). Classification of apple leaf conditions in hyper-spectral images for diagnosis of marssonina blotch using mrmr and deep neural network. *Computers and Electronics in Agriculture*, *148*, 179–187.

Patil, R. R. & Kumar, S. (2022). Rice-fusion: A multimodality data fusion framework for rice disease diagnosis. *IEEE Access*, *10*, 5207–5222.

Paymode, A. S. & Malode, V. B. (2022). Transfer learning for multi-crop leaf disease image classification using convolutional neural network vgg. *Artificial Intelligence in Agriculture*, *6*, 23–33.

Pérez-Borrero, I., Marin-Santos, D., Gegundez-Arias, M. E., & Cortés-Ancos, E. (2020). A fast and accurate deep learning method for strawberry instance segmentation. *Computers and Electronics in Agriculture*, *178*, 105736.

Pérez Roncal, C., Arazuri Garín, S., López Molina, C., Jarén Ceballos, C., Santesteban García, G., & López Maestresalas, A. (2022). Exploring the potential of hyperspectral imaging to detect esca disease complex in asymptomatic grapevine leaves. *Computers And Electronics In Agriculture, 2022, 196 (106863), 1-12*.

Picon, A., Alvarez-Gila, A., Seitz, M., Ortiz-Barredo, A., Echazarra, J., & Johannes, A. (2019). Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild. *Computers and Electronics in Agriculture*, *161*, 280–290.

Pourazar, H., Samadzadegan, F., & Dadrass Javan, F. (2019). Aerial multispectral imagery for plant disease detection: Radiometric calibration necessity assessment. *European Journal of Remote Sensing*, *52*(sup3), 17–31.

Punja, Z. K., Utkhede, R. S., et al. (2004). Biological control of fungal diseases on vegetable crops with fungi and yeasts. *Fungal biotechnology in agricultural, food, and environmental applications*, 157–171.

Qi, J., Liu, X., Liu, K., Xu, F., Guo, H., Tian, X., Li, M., Bao, Z., & Li, Y. (2022a). An improved yolov5 model based on visual attention mechanism: Application to recognition of tomato virus disease. *Computers and Electronics in Agriculture*, *194*, 106780.

Qi, J., Liu, X., Liu, K., Xu, F., Guo, H., Tian, X., Li, M., Bao, Z., & Li, Y. (2022b). An improved yolov5 model based on visual attention mechanism: Application to recognition of tomato virus disease. *Computers and Electronics in Agriculture*, *194*, 106780.

Qin, X., Zhang, Z., Huang, C., Dehghan, M., Zaiane, O. R., & Jagersand, M. (2020). U2-net: Going deeper with nested u-structure for salient object detection. *Pattern recognition*, *106*, 107404.

Rangarajan, A. K., Whetton, R. L., & Mouazen, A. M. (2022). Detection of fusarium head blight in wheat using hyperspectral data and deep learning. *Expert Systems with Applications*, *208*, 118240.

Rao, U. S., Swathi, R., Sanjana, V., Arpitha, L., Chandrasekhar, K., Naik, P. K., et al. (2021). Deep learning precision farming: grapes and mango leaf disease detection by transfer learning. *Global Transitions Proceedings*, *2*(2), 535–544.

Rosa-Márquez, E., Fornaris-Rullán, G., et al. (2003). Survey on fungal diseases affecting vegetable crops in southern puerto rico. *JOURNAL OF AGRICULTURE-UNIVERSITY OF PUERTO RICO*, *87*, 155–160.

S, R. & Vydeki, D. (2020). Recognition and classification of paddy leaf diseases using optimized deep neural network with jaya algorithm. *Information processing in agriculture*, *7*(2), 249–260.

Sai Reddy, B. & Neeraja, S. (2022). Plant leaf disease classification and damage detection system using deep learning models. *Multimedia Tools and Applications*, 1–20.

Saleem, R., Shah, J. H., Sharif, M., Yasmin, M., Yong, H.-S., & Cha, J. (2021). Mango leaf disease recognition and classification using novel segmentation and vein pattern technique. *Applied Sciences*, *11*(24), 11901.

Sarker, I. H. (2021). Deep learning: a comprehensive overview on techniques, taxonomy, applications and research directions. *SN Computer Science*, *2*(6), 1–20.

Sathiya, V., Josephine, M., & Jeyabalaraja, V. (2022). An automatic classification and early disease detection technique for herbs plant. *Computers and Electrical Engineering*, *100*, 108026.

Schor, N., Bechar, A., Ignat, T., Dombrovsky, A., Elad, Y., & Berman, S. (2016). Robotic disease detection in greenhouses: combined detection of powdery mildew and tomato spotted wilt virus. *IEEE Robotics and Automation Letters*, *1*(1), 354–360.

Seetharaman, K. & Mahendran, T. (2022). Leaf disease detection in banana plant using gabor extraction and region-based convolution neural network (rcnn). *Journal of The Institution of Engineers (India): Series A*, *103*(2), 501–507.

Shantkumari, M. & Uma, S. (2022). Grape leaf image classification based on machine learning technique for accurate leaf disease detection. *Multimedia Tools and Applications*, 1–11.

Shin, J., Chang, Y. K., Heung, B., Nguyen-Quang, T., Price, G. W., & Al-Mallahi, A. (2021a). A deep learning approach for rgb image-based powdery mildew disease detection on strawberry leaves. *Computers and Electronics in Agriculture*, *183*, 106042.

Shin, J., Chang, Y. K., Heung, B., Nguyen-Quang, T., Price, G. W., & Al-Mallahi, A. (2021b). A deep learning approach for rgb image-based powdery mildew disease detection on strawberry leaves. *Computers and Electronics in Agriculture*, *183*, 106042.

Shrestha, A. & Mahmood, A. (2019). Review of deep learning algorithms and architectures. *IEEE access*, *7*, 53040–53065.

Sibiya, M. & Sumbwanyambe, M. (2019). A computational procedure for the recognition and classification of maize leaf diseases out of healthy leaves using convolutional neural networks. *AgriEngineering*, *1*(1), 119–131.

Simonyan, K. & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 1–14. https://arxiv.org/abs/1409.1556.

Singh, P., Verma, A., & Alex, J. S. R. (2021). Disease and pest infection detection in coconut tree through deep learning techniques. *Computers and Electronics in Agriculture*, *182*, 105986.

Singh, R. K., Tiwari, A., & Gupta, R. K. (2022). Deep transfer modeling for classification of maize plant leaf disease. *Multimedia Tools and Applications*, *81*(5), 6051–6067.

Singh, U. P., Chouhan, S. S., Jain, S., & Jain, S. (2019). Multilayer convolution neural network for the classification of mango leaves infected by anthracnose disease. *IEEE Access*, *7*, 43721–43729.

Singh, V. & Misra, A. K. (2017). Detection of plant leaf diseases using image segmentation and soft computing techniques. *Information processing in Agriculture*, *4*(1), 41–49.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, *15*(1), 1929–1958.

Syed-Ab-Rahman, S. F., Hesamian, M. H., & Prasad, M. (2022). Citrus disease detection and classification using end-to-end anchor-based deep learning model. *Applied Intelligence*, *52*(1), 927–938.

Tan, M. & Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, 6105–6114. PMLR.

Tan, M. & Le, Q. V. (2021). Efficientnetv2: Smaller models and faster training. *arXiv preprint arXiv:2104.00298*.

Too, E. C., Yujian, L., Njuki, S., & Yingchun, L. (2019). A comparative study of fine-tuning deep learning models for plant disease identification. *Computers and Electronics in Agriculture*, *161*, 272–279.

Tsai, C.-W., Hsia, C.-H., Yang, S.-J., Liu, S.-J., & Fang, Z.-Y. (2020). Optimizing hyperparameters of deep learning in predicting bus passengers based on simulated annealing. *Applied soft computing*, *88*, 106068.

Tuncer, A. (2021). Cost-optimized hybrid convolutional neural networks for detection of plant leaf diseases. *Journal of Ambient Intelligence and Humanized Computing*, 1–12.

Turkoglu, M., Hanbay, D., & Sengur, A. (2019). Multi-model lstm-based convolutional neural networks for detection of apple diseases and pests. *Journal of Ambient Intelligence and Humanized Computing*, 1–11.

Turkoglu, M., Yanikoğlu, B., & Hanbay, D. (2021). Plantdiseasenet: Convolutional neural network ensemble for plant disease and pest detection. *Signal, Image and Video Processing*, *16*(2), 301–309.

Vadivel, T. & Suguna, R. (2022). Automatic recognition of tomato leaf disease using fast enhanced learning with image processing. *Acta Agriculturae Scandinavica, Section B—Soil & Plant Science*, *72*(1), 312–324.

Wang, C., Du, P., Wu, H., Li, J., Zhao, C., & Zhu, H. (2021a). A cucumber leaf disease severity classification method based on the fusion of deeplabv3+ and u-net. *Computers and Electronics in Agriculture*, *189*, 106373.

Wang, C., Du, P., Wu, H., Li, J., Zhao, C., & Zhu, H. (2021b). A cucumber leaf disease severity classification method based on the fusion of deeplabv3+ and u-net. *Computers and Electronics in Agriculture*, *189*, 106373.

Wang, Y., Wang, H., & Peng, Z. (2021). Rice diseases detection and classification using attention based neural network and bayesian optimization. *Expert Systems with Applications*, *178*, 114770.

Wang, Z., Wang, J., & Wang, Y. (2018). An intelligent diagnosis scheme based on generative adversarial learning deep neural networks and its application to planetary gearbox fault pattern recognition. *Neurocomputing*, *310*, 213–222.

Wu, Q., Chen, Y., & Meng, J. (2020a). Dcgan-based data augmentation for tomato leaf disease identification. *IEEE Access*, *8*, 98716–98728.

Wu, Q., Chen, Y., & Meng, J. (2020b). Dcgan-based data augmentation for tomato leaf disease identification. *IEEE Access*, *8*, 98716–98728.

Wu, Q., Zhang, K., & Meng, J. (2019). Identification of soybean leaf diseases via deep learning. *Journal of The Institution of Engineers (India): Series A*, *100*(4), 659–666.

Wu, Y., Feng, X., & Chen, G. (2022). Plant leaf diseases fine-grained categorization using convolutional neural networks. *IEEE Access*.

Yu, H., Liu, J., Chen, C., Heidari, A. A., Zhang, Q., Chen, H., Mafarja, M., & Turabieh, H. (2021). Corn leaf diseases diagnosis based on k-means clustering and deep learning. *IEEE Access*, *9*, 143824–143835.

Zhai, D., Shi, R., Jiang, J., & Liu, X. (2022). Rectified meta-learning from noisy labels for robust image-based plant disease classification. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, *18*(1s), 1–17.

Zhang, K., Xu, Z., Dong, S., Cen, C., & Wu, Q. (2019). Identification of peach leaf disease infected by xanthomonas campestris with deep learning. *Engineering in Agriculture, Environment and Food*, *12*(4), 388–396.

Zhang, P., Yang, L., & Li, D. (2020a). Efficientnet-b4-ranger: A novel method for greenhouse cucumber disease recognition under natural complex environment. *Computers and Electronics in Agriculture*, *176*, 105652.

Zhang, P., Yang, L., & Li, D. (2020b). Efficientnet-b4-ranger: A novel method for greenhouse cucumber disease recognition under natural complex environment. *Computers and Electronics in Agriculture*, *176*, 105652.

Zhang, S., Wang, H., Huang, W., & You, Z. (2018). Plant diseased leaf segmentation and recognition by fusion of superpixel, k-means and phog. *Optik*, *157*, 866–872.

Zhang, S., Wu, X., You, Z., & Zhang, L. (2017). Leaf image based cucumber disease recognition using sparse representation classification. *Computers and electronics in agriculture*, *134*, 135–141.

Zhang, S., Zhang, S., Zhang, C., Wang, X., & Shi, Y. (2019). Cucumber leaf disease identification with global pooling dilated convolutional neural network. *Computers and Electronics in Agriculture*, *162*, 422–430.

Zhao, X., Li, K., Li, Y., Ma, J., & Zhang, L. (2022). Identification method of vegetable diseases based on transfer learning and attention mechanism. *Computers and Electronics in Agriculture*, *193*, 106703.

Zhao, Y., Chen, Z., Gao, X., Song, W., Xiong, Q., Hu, J., & Zhang, Z. (2021). Plant disease detection using generated leaves based on doublegan. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*.

Zhao, Y., Sun, C., Xu, X., & Chen, J. (2022a). Ric-net: A plant disease classification model based on the fusion of inception and residual structure and embedded attention mechanism. *Computers and Electronics in Agriculture*, *193*, 106644.

Zhao, Y., Sun, C., Xu, X., & Chen, J. (2022b). Ric-net: A plant disease classification model based on the fusion of inception and residual structure and embedded attention mechanism. *Computers and Electronics in Agriculture*, *193*, 106644.

Zhong, Y. & Zhao, M. (2020). Research on deep learning in apple leaf disease recognition. *Computers and Electronics in Agriculture*, *168*, 105146.

Zhou, C., Zhou, S., Xing, J., & Song, J. (2021a). Tomato leaf disease identification by restructured deep residual dense network. *IEEE Access*, *9*, 28822–28831.

Zhou, C., Zhou, S., Xing, J., & Song, J. (2021b). Tomato leaf disease identification by restructured deep residual dense network. *IEEE Access*, *9*, 28822–28831.

Zinonos, Z., Gkelios, S., Khalifeh, A. F., Hadjimitsis, D. G., Boutalis, Y. S., & Chatzichristofis, S. A. (2021). Grape leaf diseases identification system using convolutional neural networks and lora technology. *IEEE Access*, *10*, 122–133.

Publications

Journal Papers

1. **Sunil C K**, Jaidhar C D, and Nagamma Patil (2021), "Cardamom plant disease detection approach using efficientNetV2 ", *IEEE Access*, Volume: 10, Pages 789–804, DOI 10.1109/ACCESS.2021.3138920.

2. **Sunil C K**, Jaidhar C D, and Nagamma Patil (2022), "Binary class and multi-class plant disease detection using ensemble deep learning-based approach ", *International Journal of Sustainable Agricultural Management and Informatics*, Volume: 8, Issue: 4, Pages 385–407, DOI 10.1504/IJSAMI.2022.10050415.

3. **Sunil C K**, Jaidhar C D, and Nagamma Patil (2023), "Tomato plant disease classification using multilevel feature fusion with adaptive channel and pixel attention mechanism ", *Expert Systems with Applications, Elsevier*, Volume: 228, Pages 120381, DOI 10.1016/j.eswa.2023.120381.

4. **Sunil C K**, Jaidhar C D, and Nagamma Patil (2023), "Systematic study on Deep Learning-based plant disease detection or classification ", *Artificial Intelligence Review, Springer*, Pages 1–98, DOI 10.1007/s10462-023-10517-0.

Conference Papers

1. **Sunil C K**, Jaidhar C D, and Nagamma Patil (2020), "Empirical Study on Multi Convolutional Layer-based Convolutional Neural Network Classifier for Plant Leaf Disease Detection", In 15^{th} *IEEE International Conference on Industrial and Information Systems (ICIIS 2020),* held at IIT Ropar, India. Pages 460–465, DOI 10.1109/ICIIS51140.2020.9342729.

Curriculum Vitae

Mr. Sunil C K
Full-Time Ph.D. Research Scholar
Department of Information Technology
National Institute of Technology Karnataka
P.O. Srinivasanagar, Surathkal
Mangalore-575 025
Email: sunilck.187it005@nitk.edu.in

Permanent Address

Sunil C K
Chinnahalli
Sakaleshpur, Hassan-573123
Email: sunilchinnahalli@gmail.com
Mobile: +918660346103

Academic Records

1. M.E. in Software Engineering from University Visvesvaraya College of Engineering, Bengaluru, 2015.
2. B.E. in Computer Science and Engineering from KVG College of Engineering, Sullia, Karnataka, 2009.
3. UGC-Net 2018, UGC-NET 2019, and GATE 2023

Research Interests

Big Data
Deep Learning
Machine Learning

Programming Languages

C, CPP, Java, Python.

Extra Curricular Activities

1. Represented NITK in Inter-NIT athletics and won two medals in 2022.
2. Have won six medals in VTU athletics meet during graduation.
3. Represented VTU in All India Inter-University athletics meet.
4. Took part in various open runs and marathons and won in a few events.
5. Agriculture, yoga, running, and playing tennis are my hobbies.