Comparative Analysis of Customized Deep Learning Models for Multi-Disease Detection in Tomato Leaves for Precision Agriculture.

By

Sayed Ahamed Khan

ID:213-15-4506

FINAL YEAR DESIGN PROJECT REPORT

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

Supervised by

Ms. Aliza Ahmed Khan

Lecturer (Senior Scale)

Department of Computer Science and EngineeringDaffodil International University

Co-Supervised by

Md Sazzadur Ahamed

Assistant professor

Department of Computer Science and EngineeringDaffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

Dhaka, Bangladesh

September 16, 2025

APPROVAL

This Project titled **"Comparative Analysis of Customized Deep Learning Models for Multi-Disease Detection in Tomato Leaves for Precision Agriculture"**, submitted by Sayed Ahamed Khan, ID No: **213-15-4506** to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **16 September, 2025**.

BOARD OF EXAMINERS

Dr. Sheak Rashed Haider Noori — **Chairman**
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Md. Sadekur Rahman — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Mr. Saiful Islam — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Md. Arshad Ali — **External Examiner**
Professor
Department of Computer Science and Engineering
Hajee Mohammad Danesh Science & Technology University

©Daffodil International University

09/19/2025 20:18

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Ms. Aliza Ahmed Khan, Lecturer (Senior Scale)** , Department of Computer Science and Engineering, Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:

14.9.25

Supervisor Name: Ms. Aliza Ahmed Khan
Designation: Lecturer (Senior Scale)
Department of Computer Science and Engineering Daffodil International University

Co-Supervised by:

Co-Supervisor Name: Md Sazzadur Ahamed
Designation: Assistant Professor
Department of Computer Science and Engineering Daffodil International University

Submitted by:

Sayed
14-9-25

Sayed Ahamed Khan
Student ID:213-15-4506
Department of Computer Science and Engineering Daffodil International University

09/19/2025 20:20

ACKNOWLEDGEMENTS

This work would not have been possible without the support and contributions of many individuals over the past two semesters. We are deeply grateful to everyone who has assisted us in one way or another.

First, we express our heartfelt thanks and gratefulness to the almighty for His divine blessing making it possible for us to complete the **Final Year Design Project(FYDP)** successfully.

We are grateful and wish our profound indebtedness to **Ms. Aliza Ahmed Khan, Lecturer(Senior Scale)** , Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh. Deep knowledge and keen interest of our supervisor in the field of **Deep Learning** to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting themat all stages have made it possible to complete this project.

We would like to express our heartfelt gratitude to the Head of the Department of Computer Science and Engineering, for his kind help in finishing our project and also to other faculty members and the staff of the Department of Computer Science and Engineering, Daffodil International University.

We would like to thank our entire course-mates at Daffodil International University, who took part in this discussion while completing the coursework.

Finally, we must acknowledge with due respect the constant support and patience of our parents.

ABSTRACT

Tomato leaf diseases are a major threat to crop production and food security particularly in the developing world where early detection of the disease is a problem. The main aim of this research is to detects and classifies tomato leaf diseases using different advanced Convolutional Neural Network architectures in real time using a custom dataset manually collected from real tomato plants. In this work I used EfficientNetB3, MobileNetV3, ResNet50, DenseNet121 and InceptionResNetV2 with Squeeze Excitation (SE) blocks, Spatial attention mechanisms, advanced augmentation technique, and transfer learning to improve model robustness. The dataset consists of nine tomato leaf image classes as a part of multi-disease detection, including diseased and healthy leaves with both front and back-side images of leaves to cover all symptoms of the disease. In previous studies on this related task only the front side of the leaf was used for training but disease symptoms are present on both the front and back sides. According to those previous studies, if the user takes a backside image, the model cannot properly detect the disease. I trained these models on many fronts and back 7,200 labeled leaf images to ensure strong performance and real-time disease detection. This study allows users to detect tomato leaf diseases in real time using images of both the front and back sides. After evaluating different five architectures as EfficientNetB3, ResNet50 and InceptionResNetV2 perform almost equal in terms of accuracy, precision, recall and F1-score. EfficientNetB3 is only 44.65 MB which is very smaller then ResNet50 and InceptionResNetV2 and MobileNetV3 is very less then another model. Although successful, its weaknesses are that it excludes certain diseases such as Mealybug Infestation and does not identify the severity of the disease stage by stage. The ultimate goal is to create an Android application to serve rural farmers. In this study provides an accurate and scalable solution for early disease detection in agriculture sector which helping reduce crop losses, improve food sustainability and assure achieving farmer profit.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

This chapter provides an overview of the research problem, motivation behind the study, key objectives, and the methodology adopted. It also highlights the expected outcomes and presents the structure of the entire report.

1.1 Introduction

Tomato (solanum lycopersicum) is a crop produced extensively in various regions around the world and contributes significantly to local agriculture production in addition to the worldwide food supply. Tomatoes contain high levels of vitamins, antioxidants and other vital nutrients, which makes them an attractive material in the human diet. Tomatoes are considered very important in food security and local economies in most countries that include Bangladesh. According to the Department of Agricultural Extension, tomatoes are produced on a number of hundreds of thousands of hectares in Bangladesh and it produces several million tons of tomatoes each year [1]. Although it is important, many challenges are posed by the tomato leaf diseases that may drastically decrease its yield and quality. Popular diseases are early blight, late blight, bacterial spots and viral infections, whose symptoms are different yet sometimes look similar. Older leaves have dark and concentric spots which are typical of early blight, whereas irregular and water-soaked lesions appear and spread rapidly, typical of late blight. Bacterial infections may affect the entire plant because viral diseases result in grey-bordered lesions, and bacterial infections do. These symptoms might evolve gradually and when not noticed early, they can cause huge losses in crops [2]. For farmers especially those in rural areas, identifying these diseases early can be difficult. Often, they don't notice the symptoms until the disease has already done serious damage. Conventionally, farmers are guided by experience or professional opinion to identify issues, but this approach is time-consuming not necessarily dependable, and not available to all. In many cases, the diseases look very similar to one another making it harder to tell them apart with the naked eye. This can lead to farmers applying the improper treatment, spending money and losing the crops [3]. Manual inspection of the disease is traditionally used to identify the disease and is ineffective, time-consuming, and error-prone to be identified by farmers or the agricultural laborers. Technology has brought about new opportunities in the past few years. Deep learning, a part of artificial intelligence has shown promise in helping computers recognize patterns in images. Using this it's now possible to teach a computer to look at a picture of a tomato leaf and say what disease it might have. But in previous study even though many models already exist, most of them are trained on lab quality images and don't better perform in real world field condition [4].

This project aims to solve that gap. It focuses on building an optimized multiple deep learning model that works with real-world images taken directly from a tomato field. The objective is to provide farmers with a reliable tool to detect the presence of a disease fast and with precision in order to take early and preventive actions to save their crops. In the long run this kind of system could be added to a mobile app and putting the power of diagnosis directly in the hands of the farmers.

In deep learning framework as Convolutional neural network is one of the better techniques for classification of any object in deep learning method. Nowadays CNN is used in many fields such as medical imaging detection, face image detection and many other applications. CNN achieves strong and accurate performance in these types of applications [6]. The main target of the work is to classify of tomato plant leaves disease detection used in CNN technique. In this study, multiple CNN models were used for classification of tomato plant leaves disease. The model is trained on hand-collected real-field images and is divided into various diseases classes like as Early Blight, Late Blight, Bacterial Spot, leaf mold and Healthy. In this study, manually collected data is used which collected from manually filed and available real time filed environment for diseases classification of tomato plant. This disease-detecting automation will allow quicker response, minimizes crop loss and help create practices that are sustainable in agriculture [8].

1.2 Motivation

Tomato farming is an important livelihood for many people, but it comes with many challenges. One of the biggest problem's farmers face is disease affecting the leaves, which can quickly reduce the yield and quality of the tomatoes. In many cases, farmers are not even aware of the issue until it has created a serious damage. This is due to the fact that most diseases of leaves are very similar and only with experience and close observation can one be able to tell which one it is at an early stage. Unfortunately, most farmers don't have easy access to plant disease experts or reliable tools to help them. This scenario led me to develop a technology to help farmers diagnose these diseases more precisely and timely. Advanced in artificial intelligence, especially deep learning approach, have shown that computers can be trained to specific recognize patterns in leaves images. This ability could be very useful in agriculture, where pictures of leaves can be analyzed to detect disease very efficiency. However, most existing models have been trained on controlled, lab-quality datasets, which do not always perform well when applied to real-world images taken under varying environmental conditions such as real tomato filed. That means farmers can't fully rely on those tools yet. Seeing this gap motivated me to develop a more practical and robust model, one that can work well with photos taken directly from tomato fields in real time leaves different pattern. These architectures would be able to deal with the complexity and the noise of the real images in the farming environment more easily. The ultimate goal is to create a tool that farmers can easily use to get early warnings about diseases, so they can act before the problem worsens. On a personal level, working on this project will enable us to become better at the area of deep learning and computer vision, which is rapidly evolving and has numerous fascinating applications. My interest in technology lies in its application in finding real solutions to real problems that pose challenges to the lives of people, particularly in the agricultural sector which forms the foundation of most economies. Helping farmers protect their crops means supporting food security and reducing economic losses in the farming community.

1.3 Objectives

This project is focused on solving a real-world farming problem by used different deep learning architecture to detect diseases in tomato leaves. The essential purpose is to create a model that can be successfully applied not only in the controlled conditions of a laboratory but also to real images that can be acquired on a tomato farm. Many existing models perform well with clean and clear datasets, but they often fail when used in natural conditions. That's why this project emphasizes building something more useful for actual farmers. Another important objective is to ensure the model can recognize multiple types of diseases, not just one or two. In total, it aims to classify nine different types of tomato leaf diseases. This would assist the farmers in knowing precisely what they are handling in order to take the appropriate action. To improve these model's learning ability, data augmentation methods like advanced and spatial transformations are used, which help the system see more image variations and perform better to achieve expected outcomes.

This project also aims to bring clarity to the predictions. With the help of Grad-CAM heatmaps, users can actually show instantly which part of the leaf the model focused on. This adds more confidence for farmers or users who don't fully trust computer systems yet.

1.4 Methodology

The overall idea of this project is to train multiple deep learning model that can recognize and classify different tomato leaf diseases from real-world images. To begin with, a large number of images were collected manually from an actual tomato field, which helped build a dataset that reflects real farming conditions. This dataset was then divided into three segment such as training, validation, and testing part. The training set had the majority of the images, while the validation and testing sets were used to check the model's performance.

In this paper five different architecture of convolutional neural network are used to detection and classification of tomato leave disease and these multiple models are respectively EfficientNetB3, MobileNetV3, ResNet50, DenseNet121, InceptionResNetV2 and using some technology of each model as SE Block + Attention Mechanism + advanced Augmentation + Transfer Learning. which is a deep neural network known for giving good results with fewer resources. Moreover, additional elements such as SE blocks and attention processes were incorporated to enhance these processes by which these models pay attention to significant elements of the leaf image. Spatial and advanced augmentations were applied to give these models more variety in training and help it learn better. Once these models were trained using transfer learning, its performance was tested using the separate test set. Accuracy, loss, and ROC curves were checked for all nine classes to measure how well it performed. A visualization method called Grad-CAM was also used, which shows heatmaps over the input images so that user can understand what the model was focusing on when making decisions. Mainly this Grad-Cam assuring that how the architecture performs and focusing the area of only disease portion.

1.5 Project Outcome

The primary finding of this project is that there is a functional deep learning model that can identify tomato leaf diseases by using images captured in the actual conditions of the farm. In contrast to most of the past researches, which use quality laboratory pictures, this model has been trained with actual field tomato leaf pictures, which were obtained manually, and as such, the model has a high probability of being applied practically in agriculture. EfficientNetB3 was the highest accuracy CNN architecture among the tested so that it demonstrated the usefulness of the model in detecting the correct presence of the disease on a leaf. A major future merit of this model is that it can be integrated into a mobile application. So that the farmers would take a picture of a tomato leaf on their smartphone and immediately access the information on the disease and thereby move in fast to address the disease and possibly lower the rate of spreading the infection. In addition, the system uses Grad-CAM visualization to depict the exact parts of the leaf that affect the models, which enhances user trust by giving the user visibility on the process of making predictions. Major key contributions outcome of this work is:

SE Blocks and Attention Mechanisms: Added to enable the model to highlight more informative parts of the leaf images to improve the extraction of features and classification of the leaf image.

Practical Relevance: It is intended to run with field-level images and not lab-quality datasets, thus having direct practical uses in farmers.

Complete Front and Back Image: Dataset has front and back leaf images to coverage of all disease symptoms so that the model is aware of slight changes in disease outlook.

Strong Performance: Can be highly accurate even in noisy field conditions, like variable light, occlusion and leaf damage.

Explain ability: Grad-CAM heatmaps can be applied to produce transparent and explainable predictions and allow users to visualize which areas of the leaf informed the model.

Accurately Multi-Class Classification: The system can accurately classify tomato leaves as Healthy, Insect-Affected, Virus-Infected, Early Blight, leaf mold and Late Blight with high accuracy and provides a comparative performance analysis of five state of the art customized different CNN architectures based on scaling, accuracy and computing cost.

Newly customized multiple architecture: The proposed custom CNN architecture is a balance between the accuracy and performance and can therefore be deployed on devices of limited resources, like mobile phones or edge devices.

Future Scalable Detection System: The present research creates a path towards the creation of an intelligent and scalable system to monitor disease on precision agriculture that can be expanded to other crops and diseases.

Dataset Contribution: The multi-class dataset of real-field conditions, curated is a valuable source of future research and model development in the detection of plant diseases.

1.6 Organization of the Report

Chapter 1 Introduction

This chapter brings out the importance of tomatoes, the issues of leaf diseases, and the necessity of good detection mechanisms. It provides a statement of the research problem and objectives, and how deep learning will be used in agriculture, as well as defines the scope of the research, methodology, and anticipated outcome of the research.

Chapter 2 Background

This chapter provides a review of literature on the detection of tomato leaf disease (conventional manual diagnosis, image processing methods, and deep learning models like CNNs). It covers the way these approaches have changed with time and their effectiveness. It is also in this chapter that I figured out gaps in the existing literature, which present the foundation of the proposed research and its purpose thereof.

Chapter 3 Research Methodology

The chapter describes how the proposed system will be designed and implemented step by step. It includes information about data collection, data labelling, preprocessing such as resizing, augmentation, normalization, and the design of the followed CNN model, as well as training and testing processes. The selected methods and the rationale behind their use, as well as a comparative fit-up to assess the performance with current models, are also addressed in the chapter.

Chapter 4 Implementation and Results

The results of the proposed five customized CNN models are presented in the form of experimental results in this chapter and compared based on accuracy, precision, recall, and F1-score. Results are also presented in the form of confusion matrices, training and validation curves and sample outputs. The analysis reveals the strengths and weaknesses of the model and talks about the practical value of the findings.

Chapter 5 Engineering Standards and Design Challenges

Chapter 5 links the research findings to knowledge and practices of engineering, and demonstrates that the findings follow academic and professional standards. It covers hardware and software architecture, communication, and societal, environmental, and ethical considerations of agricultural implementation. The chapter also addresses financial and resource management, and the way in which complex engineering issues were handled in a well-organized way.

Chapter 6 Conclusion

Chapter 6 is the conclusion of the study, where several important findings, limitations, and contributions to the research are summarized regarding CNN-based architecture in the detection of tomato leaf disease. It points out the enhanced accuracy and reliability of the model over traditional methods. Future directions are also described in the chapter, such as integrating systems in real time, extending to other crops, bigger datasets, and integrating with smart farming technologies.

Chapter 2
Background

This chapter outlines the theoretical foundation of the research, covering tomato leaf diseases, deep learning concepts, and existing work related to CNN-based disease detection models. It sets the stage for understanding the proposed solution.

2.1 Introduction

This chapter shares some background that helps explain why this project is important and how it connects to current developments in agriculture and technology. The point is to realize how tomato leaf diseases can damage crops and how artificial intelligence and, in particular, deep learning can be used to address this issue. Tomatoes are widely grown and used in everyday cooking across the world. But the plants are often affected by leaf diseases, which farmers may not notice in time. These problems reduce the quality of crops, and deny many growers their earnings. In remote areas, farmers usually don't have the tools or expert support needed to deal with such problems quickly. With time, technology has made progress, and image recognition through deep learning has become a useful tool in many areas, including agriculture. Now it's possible to use photos of plant leaves to find out if a disease is present. This method is fast and doesn't require someone to be an expert. This section preludes the entire report. It elaborates on the main aspects of tomato farming, the main challenges in detecting diseases, and ways in which machine learning could resolve these challenges. It provides the base needed before diving deeper into the methods and results discussed in the next chapters.

2.2 Literature Review

A number of works have examined how deep learning models can be used to detect tomato leaf diseases. Tan et al. [1] compared the classical and deep learning architectures and found that ResNet34 is the most effective architecture. Ullah et al. [2] suggested EffiMob-Net which is a hybrid deep learning model with a better accuracy than the traditional methods. Alam et al. [3] have shown that custom CNNs perform well and include robustness needs to be improved to suit the real-world scenario. Agarwal et al. [4] indicated that classification with CNNs is highly enhanced by data augmentation. Gerdan et al. [5] demonstrated that using pre-trained architecture with a proposed CNN performed better on the PlantVillage dataset. As Anandhakrishnan and Murugaiyan [6] indicated, Xception V4 was more accurate with less number of parameters though the data was not extensive. Aquil and Ishak [7] tested scratch and pre-trained CNNs and suggested DenseNet-120 to be deployed to a mobile to allow detection in real-time. According to Gehlot and Saini [8], different CNN architectures were examined and it was proposed that lightweight models can be better deployed on small resource devices. Parvez et al. [9] noted that GoogLeNet was very accurate, and promoted the lightweight implementation in field applications. Trivedi et al. [10] emphasized the fact that high-performance CNNs can identify diseases at their early stages, which means that the advanced methods of deep learning are the key to real-world precision agriculture. Together, these studies show a high advancement in model accuracy and gaps in real-field applicability, interpretability and multi-class disease detection, which is the focus of this research.

Table 2.1: Summary of Literature Reviewed.

Author (s)	Year	Title	Methodology	Key Findings
Tan et al. [1]	2021	Tomato Leaf Diseases Classification Based on Leaf Images:A Comparison between Classical Machine Learning and Deep Learning Methods	AlexNet, VGG16, ResNet34, EfficientNet-b0, MobileNetV2	ResNet34 performed best among tested architectures
Ullah et al. [2]	2023	EffiMob-Net: A Deep Learning-Based Hybrid Model for Detection and Identification of Tomato Diseases Using Leaf Images	Hybrid Deep Learning (EffiMob-Net)	Hybrid model showed improved accuracy over conventional DL models
Alam et al. [3]	2024	Comparing pre-trained models for efficient leaf disease detection: a study on custom CNN	Custom CNN,Transfer Learning	Custom CNN effective, but real-world performance requires more robustness
Agarwal et al. [4]	2020	ToLeD: Tomato Leaf Disease Detection using Convolution Neural Network	CNN	Data augmentation improved classification performance
Gerdan et al. [5]	2023	Diagnosis of Tomato Plant Diseases Using Pre-trained Architectures and A Proposed Convolutional	Custom CNN and pretrained models (VGG16, MobileNet, DenseNet201, InceptionResNetV2); PlantVillage augmented tomato leaf dataset	CNN achieved 98.26%, outperforming VGG16 (93.88%) and MobileNet (93.5%); Proposed CNN performed

		Neural Network Model		best but tested on a limited dataset.
Anandhakrishnan [6]	2020	Identification of Tomato Leaf Disease Detection using Pretrained Deep Convolutional Neural Network Models	Pretrained CNN models (VGG16, Xception V4, ResNet, AlexNet, LeNet); Fine-tuning, Transfer Learning, SGD optimizer on PlantVillage dataset	Xception V4 achieved 99.45% accuracy in 30 epochs; Proposed CNN reached 95.03%; VGG16 performed poorly; Dataset limited to PlantVillage; Xception was efficient with fewer parameters.
Aquil [7]	2021	Evaluation of Scratch and Pre-trained Convolutional Neural Networks for the Classification of Tomato Plant Diseases	CNN classification using pretrained models (ResNet, DenseNet, VGG, SqueezeNet); Kaggle dataset with 10 classes	VGG16 achieved the highest accuracy of 99.84%; No early-stage image testing; Suggested future mobile app development with DenseNet-120 for real-time detection.
Gehlot [8]	2020	Analysis of Different CNN Architectures for Tomato Leaf Disease Classification	Compared CNN models (VGG16, GoogleNet, ResNet, etc.); PlantVillage dataset	ResNet achieved 99.68% but was too large for mobile deployment; VGG16 provided good performance with lighter complexity; lightweight models recommended for device use.
Parvez et al. [9]	2023	Tomato Leaf Disease Detection Using Convolutional Neural Network	CNN models (VGG16, GoogleNet, ResNet, ImageNet pretrained models); PlantVillage dataset	GoogLeNet achieved 99.23%, VGG16 98.00%; Models not lightweight, limiting deployment; Recommended lightweight implementation for device usage.

Trivedi et al. [10]	2021	Early Detection and Classification of Tomato Leaf Disease Using High-Performance Deep Neural Network	CNN model on Tomato leaf disease dataset	Achieved 98.49% accuracy; CNN outperformed other models in early disease detection; Suggested use of advanced deep learning methods for higher precision.

2.3 Gap Analysis

Though a few systems like PlantVillage, PlantDoc, Plantix, Leaf Doctor and Agro AI App have been designed to detect plant diseases, they display significant gaps when used in actual farming environment. The available datasets mostly do not have real field tomato images or offer only semi-verified samples, and these decrease reliabilities. Though there are mobile apps that do the detection in real-time, they tend to be sensitive to noisy fields, lack accuracy and do not have a data privacy policy. Moreover, most of these systems lack custom model optimization, sophisticated augmentation methods or visual explain ability features like Grad-CAM, which are essential when it comes to trust and interpretability. By contrast, the following method, in the proposed method, directly overcomes these deficiencies through the application of real-field, manually inspected tomato leaf images, advanced augmentation and optimization techniques, supporting multi-class classification, assuring privacy of data, and implementing Grad-CAM to provide transparent predictions. This renders it stronger, more precise, and viable as far as precision agriculture is concerned.

Table 2.2: Gap Analysis.

Features	**PlantVillage Dataset**	**PlantDoc Dataset**	**Plantix App**	**Leaf Doctor**	**Agro AI App**	**Proposed System**
Uses real field image	No	Yes	Yes	No	Yes	Yes
Custom model optimization	No	No	No	No	No	Yes
Visual Explanation using Grad-cam	No	No	No	No	No	Yes
High accuracy in noisy field conditions	No	No	No	No	No	Yes

Focused on tomato leaf diseases specially	No	No	No	No	No	Yes
Uses advanced augmentation techniques	No	No	No	No	No	Yes
Multiclass disease classification supported	Yes	Yes	Yes	No	Yes	Yes
Dataset open source availability	Yes	Yes	No	No	No	No
Real time detection feedback	No	No	Yes	No	Yes	Yes
Data Privacy ensured	No	No	No	No	No	Yes
Dataset manually verified	No	No	No	No	No	Yes
Future-ready architecture for mobile deployment	No	No	Yes	No	Yes	Yes
Training & testing on custom manually collected data	No	No	No	No	No	Yes
Lightweight model structure for low-resource devices	No	No	No	No	No	Yes

2.4 Summary

In this chapter, I explored the background and existing works related to plant leaf disease detection, with a focus on tomato leaf diseases. I reviewed several datasets, mobile applications, and research studies that have contributed to this domain. While many of them achieved considerable progress, I identified notable limitations such as poor performance in real-field conditions, lack of interpretability, and limited focus on tomato specific diseases. The gap analysis table above rightly shows that the huge majority of current work do not have real time feedback, explain ability such as Grad-CAM, and model building to custom. My proposed approach aims to address these gaps by using a manually collected dataset, optimizing with a lightweight architecture as EfficientNetB3, MobileNetV3, ResNet50, DenseNet121, InceptionResNetV2 model, and applying attention-based techniques for better accuracy and interpretability. In this chapter, the methodology is established upon, which will be discussed in the following section.

Chapter 3
Research Methodology

This chapter outlines the data collection, preprocessing, and model development steps used to build the tomato leaf disease detection system. It also highlights the techniques applied for training and evaluating the model.

3.1 Methodology

In this section i outline the step by step approach followed in that research to develop a dep learning based system for detecting and classification of tomato leaf diseases using manually collected image data.

Data Collection

To ensure real-world performance, a custom dataset was manually collected using smartphone cameras from my farm on my home roof, where I planted many tomato trees for the purpose of this research. After cultivating tomato plants, images were collected while ensuring quality and clarity, and only clearly visible tomato leaf images were kept and properly labeled. My dataset contains a total of 10,800 labeled images with 9 different classes, such as bacterial leaf spot, early blight, late blight, leaf mold, Septoria leaf spot, target spot, mosaic virus, yellow leaf curl virus, and the healthy class of tomato leaves. They were taken in actual environmental conditions with natural lighting, different angles, and field backgrounds.

Data Preprocessing:

The tomato leaf images received manually were initially resized to 224x224 to make it compatible with the deep learning model. In order to make the model perform well and to be strong in real-field conditions, a couple of preprocessing steps were used. The pixel values were scaled to a range between 0 and 1 and this helped model convergence faster and more consistently. State of the art data augmentation was used to artificially augment the size of the training data to prevent overfitting. These were random horizontal and vertical flips, zooming, adjustments of brightness and contrast, and rescaling. Such augmentations are natural manipulations of the leaf orientation, light and environmental conditions, which are real farming conditions. Since the dataset is multi-class, the model output layer uses the SoftMax activation function, which allows the prediction of probabilities of nine disease classes, but also of the healthy tomato leaf class. The preprocessing pipeline makes the model highly configured to accommodate variability of leaf appearance and disease presentation, hence, aiding real-time detection of tomato leaf diseases effectively and reliably.

Data Distribution:

Among the manually collected tomato leaf images of 10,800 in total, 7,200 images were to be used in training, 1,800 images were to be used in validation, and 1,800 images were to be used in testing. This distribution allows the model to be trained on a large enough dataset but it is also being trained and tested on different, unseen images to assess its overall generalization and performance on all nine tomato leaf disease classes, including healthy leaves.


(A)


(B)


(C)


(D)


(E)


(F)


(G)


(H)


(I)

Figure 3.1: Sample disease of tomato leaf image

Mealybug Infestation Disease (1) Tomato Leaf Curl Virus initial Stage (2)

Tomato Leaf Curl Virus moderate Stage (3) Late Blight last Stage whole tree dies(4)

Figure 3.2: Sample of diseased tomato leaves from my home rooftop

Both figures [Figure 3.1 and Figure 3.2] show my manually collected dataset from my home rooftop, where I cultivated tomato crops similar to real-field tomatoes for this research purpose. My manually dataset: (A) Bacterial leaf spot Diseases;(B) Early Blight Diseases;(C) Late Blight diseases;(D) Leaf Mold;(E) Septoria spot;(F) Target Spot;(G) Tomato Healthy leaf ;(H) Mosaic virous;(I) Yellow leaf curl disease;(1) Mealybug Infestation Disease; (2) Tomato Leaf Curl Virus Initial Stage; (3) Tomato Leaf Curl Virus Moderate Stage; (4) Late Blight last Stage whole tree dies.

Model Selection and Optimization:

In this paper, 5 different model architecture used respectively EfficientNetB3, MobileNetV3, ResNet50, DenseNet121, InceptionResNetV2 and these models was selected due to its balance between accuracy and lightweight performance. This model was tuned on tomato data using the transfer learning that enabled rapid and efficient convergence and greater accuracy. Greater generalization was achieved through Advanced Augmentation. An SE block (Squeeze-and-Excitation) was added to give importance to relevant features. Grad-CAM (Gradient-weighted Class Activation Mapping) was apply to visualize which part of the leaf image influenced the prediction the most. Mainly, it's a visual explanation technique that highlights which region is important and focused.

3.1.1 Overview

That research focused on an implemented an AI system which detect and classification of tomato tree leaves disease using deep learning technique. The actual objective is to develop a powerful and efficiently image classification algorithm capable of distinguishing the visual signs of disease and health of tomato leaf in the auto mode. To achieve this a custom dataset which was manually collected from real farm environment to showing real filed condition. These model architectures were build using EfficientNetB3, MobileNetV3, ResNet50, DenseNet121, InceptionResNetV2 with advanced data augmentation method, Grad-cam visualization and SE blocks used for better performance and perform better for real time detection. The final outcome aims to contributed to smart agriculture by supporting farmer to professionally indemnify leaf disease early and accurately.

3.1.2 Proposed Methodology

In this research, I designed a comparative deep learning architecture to detect multiple classes of tomato leaf diseases and classify them properly in a realistic way. My methodology follows a step-by-step pipeline that includes model design and evaluation using several deep learning frameworks, including convolutional neural network architectures.

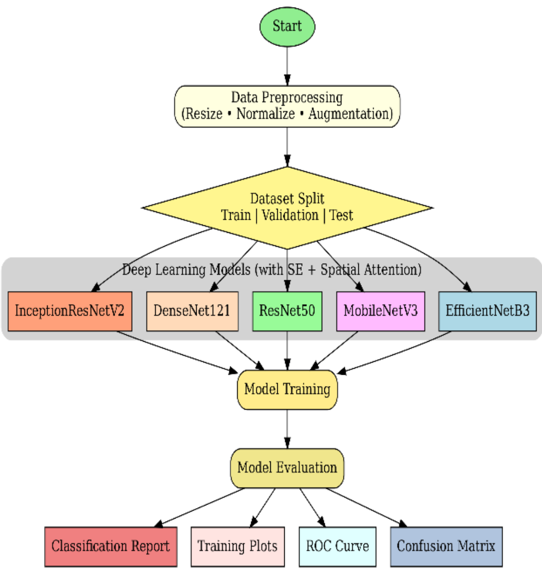

Figure 3.3: Proposed system architecture Schematic Representation.

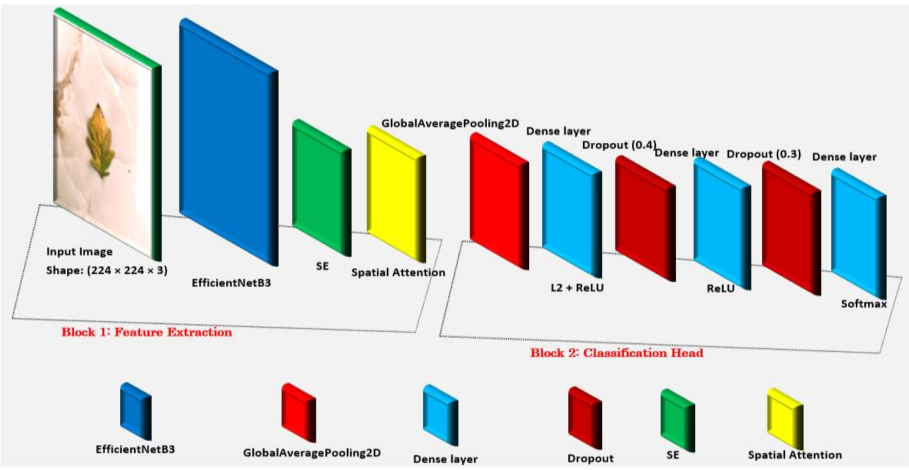

Figure 3.4: Architecture of EfficientNetB3.

Table 3.3: Information of the layer structure EfficientNetB3

Block	Layer	Type	Configuration
Block 1: Feature Extraction	Input Layer	Input	Input shape = (224 × 224 × 3)
	EfficientNetB3 (Base Model)	Convolutional Backbone	Pretrained on ImageNet, all layers fine-tuned (trainable)
	Squeeze-and-Excitation (SE)	Channel Attention	Built-in within EfficientNetB3 architecture
	Spatial Attention	Attention Module	Enhances spatial feature representation
Block 2: Classification Head	Global Average Pooling2D	Pooling Layer	Converts (H × W × C)→ (1 × 1 × C)
	Dense Layer 1	Fully Connected	512 units, Activation = ReLU, L2 Regularization (λ = 1e-4)
	Dropout Layer 1	Regularization	Dropout rate = 0.4
	Dense Layer 2	Fully Connected	256 units, Activation = ReLU
	Dropout Layer 2	Regularization	Dropout rate = 0.3
	Output Layer	Fully Connected	9 Units (number of disease classes), Activation = Softmax

Custom CNN Architectures:

Five deep learning architectures designed in this paper were used to classify tomato leaf diseases in the real field environment. Both models use a state-of-the-art backbone and implement a combination of advanced feature enhancement methods, including Squeeze-and-Excitation (SE) blocks and spatial attention along with custom dense layers to enhance both robustness and classification accuracy. And adding customized dropout layer of each model so that model not overfitting. These features I adding as model customization and attention layer focus on channel wise attestation that's why model work very efficient way.

EfficientNetB3: The EfficientNetB3 framework is used as a base, can be trained in full and pre-trained on ImageNet. It includes SE blocks and space attention in order to focus on important disease characteristics. The head comprises of global average pooling and then fully connected layers that have dropout and L2 regularization to ensure overfitting. The model is shown to perform well on multiclass tomato leaf disease data, especially in the noisy real-field case. The strength of this model is its ability to trade off accuracy with computational efficiency to scale depth, width and resolution of the compounds.

MobileNetV3: MobileNetV3-Large is a lightweight and efficient network with full training on SE and spatial attention improvements. It is mobile and edge device optimized without sacrificing accuracy. Global average pooling, dense layers, and dropout layers are present in the head to enhance generalization, thus being an ideal choice in on-field disease detection. It is a residual architecture, thus guaranteeing a superior solution to the vanishing gradient issue and permits training of deeper networks.

ResNet50: The ResNet50 backbone is used, partially frozen so that the first 100 layers are frozen and the rest fine-tuned. SE blocks and spatial attention are utilized to obtain meaningful features on tomato leaves. The head of the classification incorporates dropout, dense layers, as well as batch normalization. This model is cost-effective in terms of computational calculations and highly precise, which makes it appropriate to use in the field at the large scale.

DenseNet121: DenseNet121 backbone is used with full fine-tuning and is enhanced with SE and spatial attention blocks to obtain significant feature representations. The classification head makes use of batch normalization, dense layer, and dropout to maintain stable training and avoid overfitting. This architecture is made to be highly accurate and yet provide computational efficiency to ensure practical deployment. It is a lightweight architecture designed to be used in mobile and edge devices, with high speed and fewer parameters.

InceptionResNetV2: InceptionResNetV2 backbone can be trained with SE, and spatial attention blocks. The head is globally averaged, uses batch normalization and a few dense layers with dropout and L2 regularization. The model has been identified to be particularly effective in the extraction of features of the intricate leaf pictures and this provides an effective categorization of maladies in the existent multifaceted field scenarios. InceptionResNetV2 is an architecture that combines inception modules as well as residual connections, which makes its classification of features effective in the complex data.

Each of the models was trained on ImageNet. The last fully connected layers were substituted by task-specific layers with dropout, batch normalization, and dense layers with SoftMax activation, which multi-class detection and classification were achieved.

3.1.3 Data Flow Diagram

The section depicts the data flow in the proposed system. A Data Flow Diagram (DFD) is a simple depiction of information flows in the system in a clearer manner through the various components of the system. The main aspect of this study is that tomato leaf images are processed and categorized through the pre-trained deep learning models. The systematic flow of acquiring an image to pre-processing, feature extraction, and ultimate disease classification is emphasized in the DFD. This hierarchical diagram highlights the working processes of the automated detecting mechanism, and there can be no misunderstanding of the process of how raw images information is converted to precise diagnostic results without using databases or third-party programs.

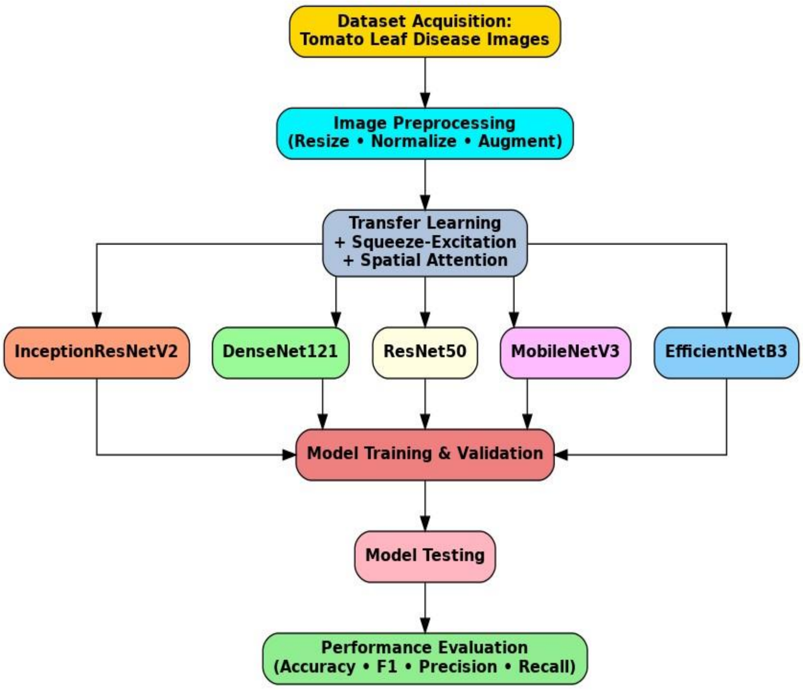

Figure 3.5: Block diagram of Overall System work Flow Combinedly 5 model.

3.2 Detailed Methodology and Design

Here I describe the approach that I have taken in terms of detecting and classifying tomato leaf diseases, as well as alternative approaches that I have considered and reasons why I chose my particular solution. Other Solutions which were thought of.

Alternate Solutions Considered

Traditional Machine Learning Methods: The first model was a comparison of classical machine learning algorithms like Support Vector Machines (SVM), Random Forests and K-Nearest Neighbors (KNN). Such techniques usually involve manual feature points such as texture, color histograms or shape features. These methods tend to be less effective at capturing complex leaf image patterns especially in different lighting and environmental conditions despite their simplicity and speed of training.

Basic Convolutional Neural Networks (CNNs): Architectures of standard CNNs such as VGG16 and ResNet101 were considered. They are automated feature extractors and usually perform better on image data than classical methods. Nevertheless, it requires very large labeled datasets to train such models, which are not always accessible. Pretrained model transfer learning can alleviate this problem but is not flexible enough to capture fine-leaf disease details.

Attention Mechanisms and Advanced Architectures: Later deep learning improvements, such as attention modules and squeeze-and-excitation (SE) blocks, were added to the model to aid in emphasizing the most important parts of the leaf images. Methods such as Advanced augmentation are used to enhance the generalization of the model through the creation of synthetic training samples and support to allow the model to be detected correctly.

Why I Selected this specific Solution:

I chose 5 different methodology respectively EfficientNetB3, MobileNetV3, ResNet50, DenseNet121, InceptionResNetV2 known for these strong performances with fewer parameters, and their performance analysis, which is performed better, is for the purpose of comparing these models. each model with SE blocks and Attention Mechanisms to boost feature representation. These hybrid architectures allow the model to emphasize critical disease patterns in the leaves, improving accuracy. Additionally, Employed Advanced augmentation during training to artificially increase the diversity of the dataset, which enhances the model's ability to generalize to unseen samples. Finally, Transfer Learning leverages pretrained weights on large image datasets, reducing training time and improving convergence. This integrated approach balances model complexity, training efficiency, and detection accuracy, making it ideal for real-time precision agriculture applications. And I used manually collected data from real fields so that the model performs better in real-time tomato crop leaf detection and classification. Mainly, I use my raw dataset for training and validation, also using real field data for testing. Above all, by using real field raw data, this model performs very well in real environments, so that the agriculture sector gets accurate feedback. Mainly, farmers get proper support for their tomato crop leaf diseases in real-time feedback.

3.3 Project Plan

The project plan outlines the step by step approach and timeline for developing. the tomato leaf disease detection system. It ensures systematic progress and successful completion within the scheduled timeframe.

Phase 1: Requirement Analysis and Dataset Collection

Understand the problem domain and research related work. Collect raw images of tomato leaves from real fields manually. Organize the dataset into training, validation, and testing subsets and each class equal data keep.

Phase 2: Data Preprocessing and Augmentation

Resize and normalize images to a uniform size suitable for the model. Apply data augmentation techniques, including Advanced, to increaseing dataset diversity so that model train accurately and training on variety of dataset.

Phase 3: Model Development

Choose these architectures as EfficientNetB3, MobileNetV3, ResNet50, DenseNet121, InceptionResNetV2. Integrate Squeeze-and-Excitation (SE) blocks and attention mechanisms for enhanced feature learning. And Apply transfer learning using pretrained weights.

Phase 4: Training and Validation

Train these different models on the prepared manual dataset. Validate the model regularly to monitor performance and prevent overfitting.

Phase 5: Testing and Evaluation

Test the trained model on real field data to check real-time performance. To knowing performance of five different architecture, Calculating evaluation metrics such as accuracy, precision, and recall.

Phase 6: Visualization and Deployment

Implemented Grad-CAM visualization to highlight diseased leaf regions. Develop a system for real-time processing and user interface for future practical use.

Phase 7: Documentation and Reporting

Document all methodologies, experiments, and results. Prepare the final report.

3.4 Task Allocation

This table depicts the timeline of the principal activities in each period of the project, from week 12 to week 48

Table 3.4. Project Timeline Showing Key Activities from Week 12 to Week 48

Tasks	Weeks																		
	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40	42	44	46	48
Data collection phase																			
Preprocess all the data																			
Model training																			
Model evaluation																			
Create Documentation																			

3.5 Summary

In this chapter, I presented the complete methodological workflow for developing an optimized deep learning model for tomato leaf disease detection. I started with a clear requirement analysis by searching and comparing different solutions and eventually determined different 5 architecture based on EfficientNetB3, MobileNetV3, ResNet50, DenseNet121, InceptionResNetV2, each architecture was combined with SE blocks, spatial attention mechanisms and Advanced augmentation technique. To assure real-world applicability, I emphasized the need to use manually collected real-field data. All the phases of development such as the data collection, model evaluation etc. were outlined in the detailed project plan. With all this, this chapter provides the basis to introduce a powerful, precise, and field-capable agricultural disease detection system.

Chapter 4

Implementation and Results

This chapter presents the implementation details of these proposed model as EfficientNetB3, MobileNetV3, ResNet50, DenseNet121, InceptionResNetV2 along with the dataset preparation, training process, and evaluation metrics. It also discusses the experimental results, including accuracy, loss, and visual explanations using Grad-CAM for each 5-custom model.

4.1 Environment Setup

My proposed modes were developed and trained on Kaggle Notebook, a cloud-based platform ideal for running Python scripts and deep learning workflows with GPU acceleration. Kaggle provides a friendly platform enabling powerful hardware and access to core machine learning libraries, which are well aligned to deep learning research. In this research, I used manually collected tomato leaf disease dataset from real tomato filed that includes images from real agricultural field conditions. The custom dataset was preprocessed and augmented to ensure diversity and improve the model's generalization capability in real-time. I trained my model for different number of epochs and employed early stopping to monitor validation performance and halt training when no further improvement was observed, helping to prevent overfitting. These proposed 5 different model architecture is based on EfficientNetB3, MobileNetV3, ResNet50, DenseNet121, InceptionResNetV2 integrated with Squeeze-and-Excitation (SE) blocks and Attention Mechanism for enhanced feature refinement. Further, I used Advanced Augmentation and Transfer Learning to enhance performance with limited training data and robustness when used in noisy field conditions. Based on different core Python libraries, the application was developed. The model was constructed and trained with TensorFlow as the technology of choice as the backend framework, and Keres as the high-level API used to design and manage deep learning models. Numerical operations were performed with NumPy, plotting training graphs and visualizations were done with Matplotlib, and training progress monitoring, metric visualizations, and learning curve inspection were performed with Tensor Board. To improve model reliability, Model Checkpoint was set to store the most optimal model according to the accuracy of the validation. For optimization, I used the Adam optimizer, which combines the advantages of Adaptive Gradient Algorithm and Root Mean Square Propagation. I used the categorical cross-entropy loss function as it is an effective method to compute error between predicted and true label probabilities since the classification task was multi-class in nature. All these tools and techniques allowed us to train an explainable high-performing model to detect tomato leaf disease in real-time and under real-world conditions.

4.2 Testing and Evaluation/Performance/ Comparative Analysis

The five proposed deep learning architecture are evaluated in this section with respect to how they perform in classification in real world settings. The testing procedure was done with both the manually obtained and proven datasets; the test set contained a balanced sample of 1,800 tomato leaf pictures that were equally spread in nine disease groups to give unbiased results. The accuracy, precision, recall and F1-score, as based on the classification report and confusion matrix were used to measure performance. The comparison analysis showed that some of the models converged quicker and had a lower calculation cost but others were more accurate and generalized under a noisy field condition. On the whole, the findings support the usefulness of the suggested models, and hybrid improvements are clearly beneficial compared to conventional CNN-based models. These results indicate that the models are feasible and can be utilized in agricultural practice in identification of tomato leaf disease.

Evaluation Metrics

The models were evaluated in terms of four standard evaluation measures, which were respectively Accuracy, Precision, Recall, and F1-score. These measures were calculated based on the confusion matrix to guarantee a just evaluation of the classification performance with regards to all the disease categories.

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

Precision = $\frac{TP}{(TP+FP)}$

Recall = $\frac{TP}{(TP+FN)}$

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

All of these metrics give us an insight regarding the reliability, robustness, and the overall efficiency of the proposed deep learning models in terms of tomato leaf disease classification.

Performance and Comparative Analysis

It is in this section the five proposed deep learning approach are discussed in terms of their performance in the training and validation stages in the classification of tomato leaf diseases. The test was performed on a dataset of 1,800 images, with nine disease categories represented equally, so that there was no bias in the sample in terms of each category. Accuracy, precision, recall and F1-score are standard metrics that were evaluated using confusion matrices to determine the effectiveness of each model. The comparative analysis brings out the advantages and shortcomings of both models. Other models were also trained at reduced rates and less computation and other models obtained greater accuracy and generalization performance in demanding field scenarios. This illustrates the efficiency-predictive performance tradeoffs. In general, the findings demonstrate that adding hybrid advances to the existing SE blocks and attention mechanisms can improve disease detection, and such models are feasible and applicable to the agricultural practice.

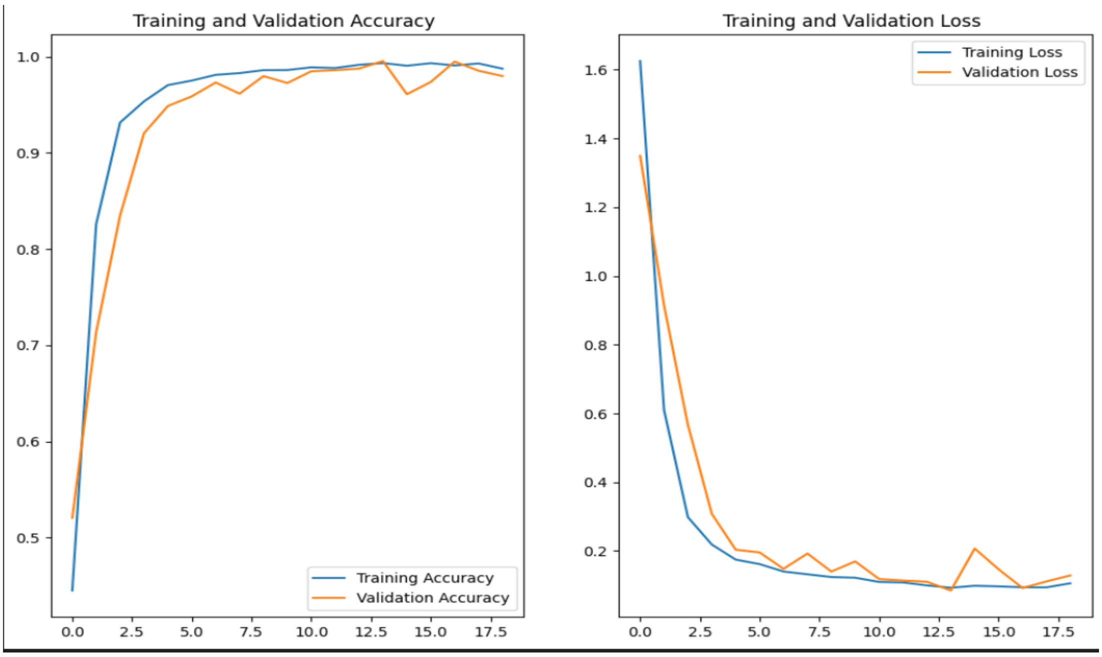

Figure 4.6: Plots of Accuracy and loss EfficientNetB3.

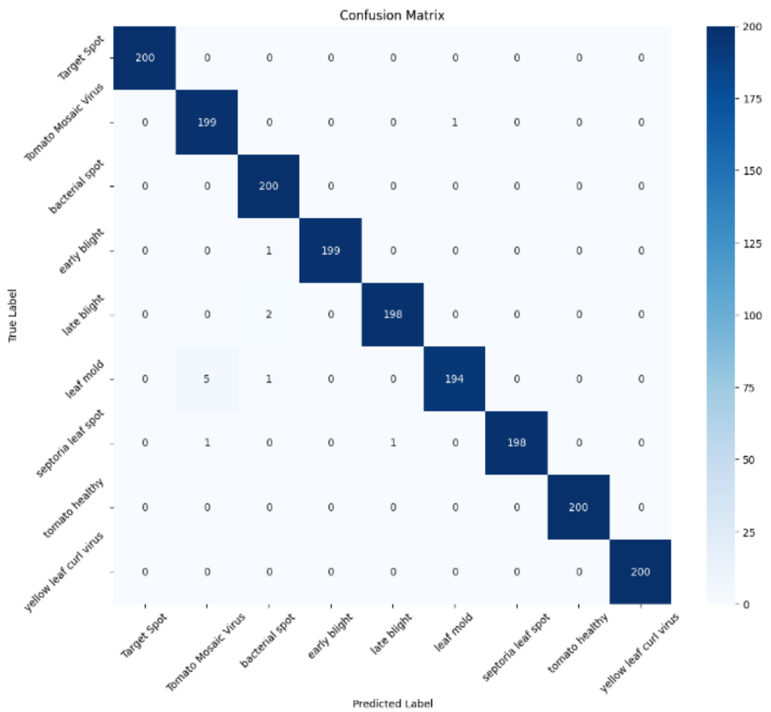

Figure 4.7: Confusion Matrix EfficientNetB3

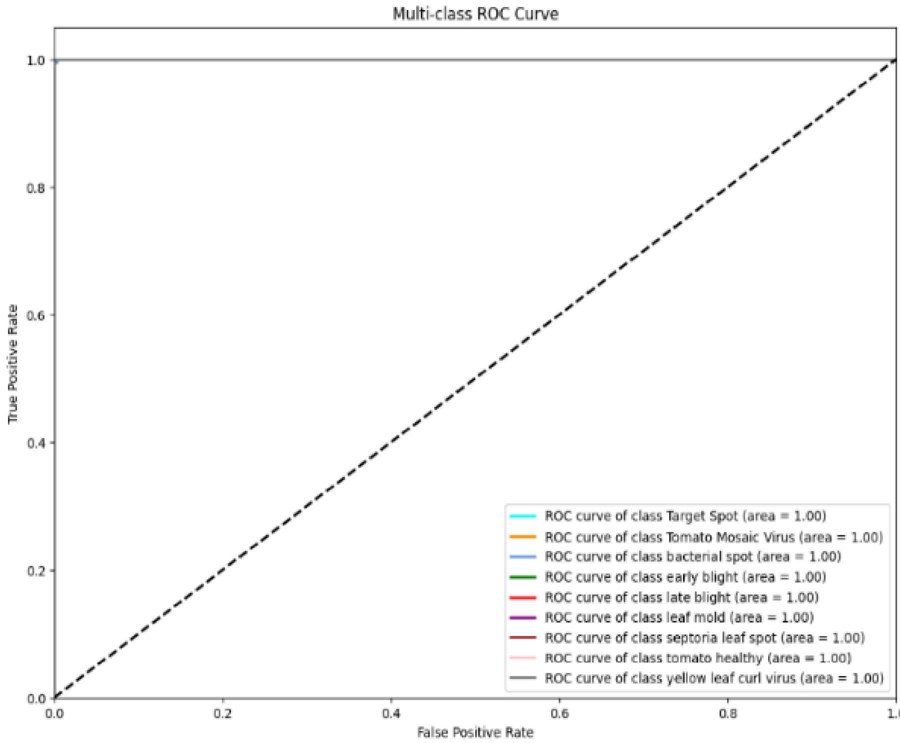

Figure 4.8: Roc Curve EfficientNetB3

Figure 4.9: Model Prediction of Tomato Leaf Disease.

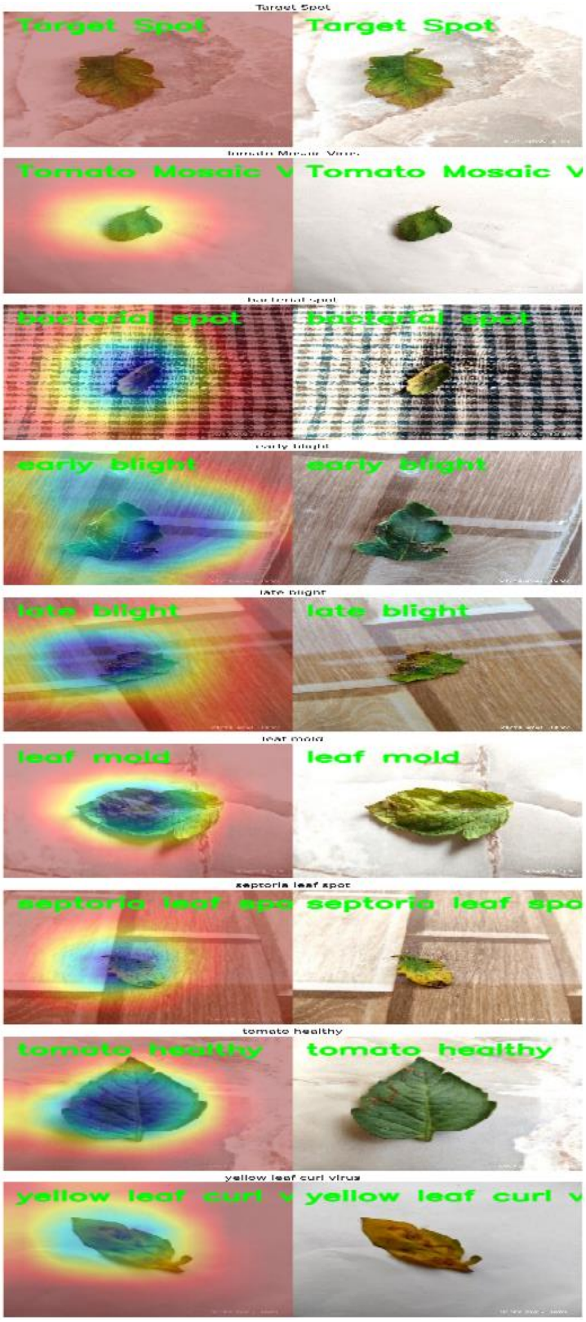

Figure 4.10: Visualization of Predicted Tomato Leaf Diseases Using Grad-CAM

Used 40 epochs with early stopping for training, that's why training stopped before completing the full number of epochs. After training completed, the training accuracy was 99.23%, validation accuracy was 99.56%, and finally, the test accuracy achieved was 99.33% for EfficientNetB3. Plots of accuracy and loss for EfficientNetB3 are given in Figure 4.6. The confusion matrix of EfficientNetB3 is shown in Figure 4.7. The multi-class ROC curve for tomato leaf disease classification is presented in Figure 4.8. In Figure 4.9, the model prediction of tomato leaf disease as Yellow Leaf Curl Virus is shown perfectly. Visualization of predicted tomato leaf diseases using Grad-CAM is shown in Figure 4.10.

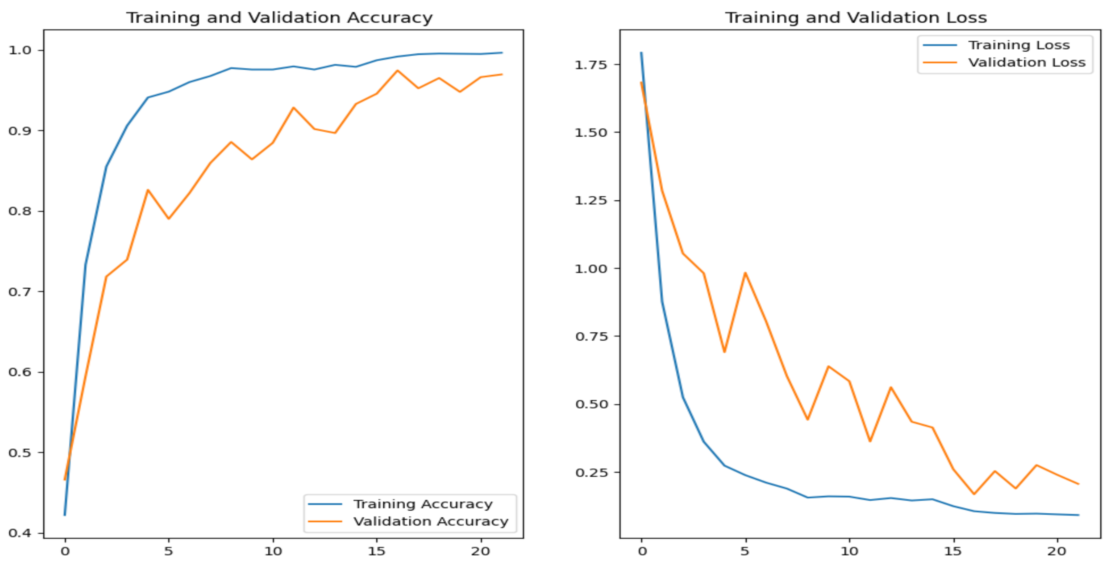

Figure 4.11: Plots of Accuracy and loss ResNet50.

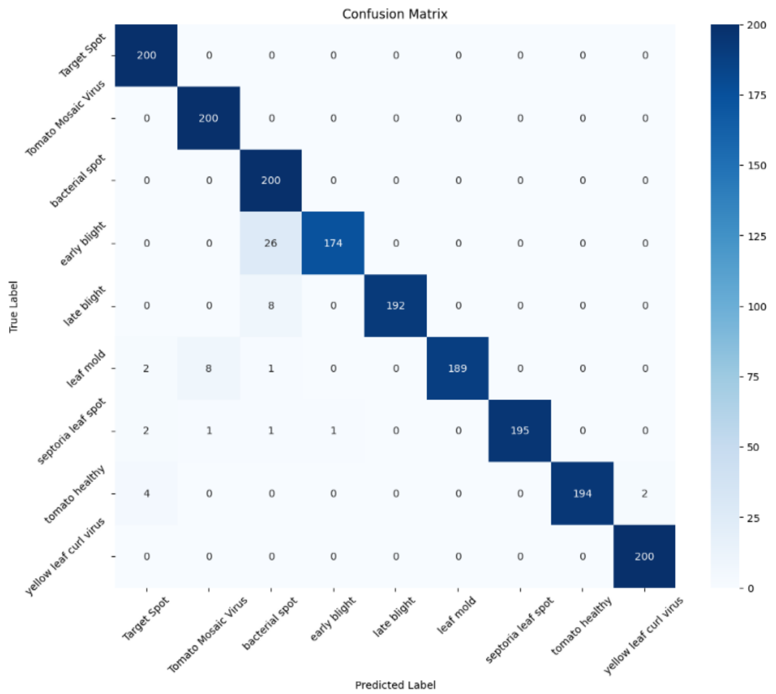

Figure 4.12: Confusion Matrix ResNet50

Used 20 epochs with early stopping for training, that's why training stopped before completing the full number of epochs. After training completed, the training accuracy was 99.54%, validation accuracy was 96.94%, and finally, the test accuracy achieved was 96.89% for ResNet50. In Figure 4.11, plots of accuracy and loss for ResNet50 are given. The confusion matrix is shown in Figure 4.12.

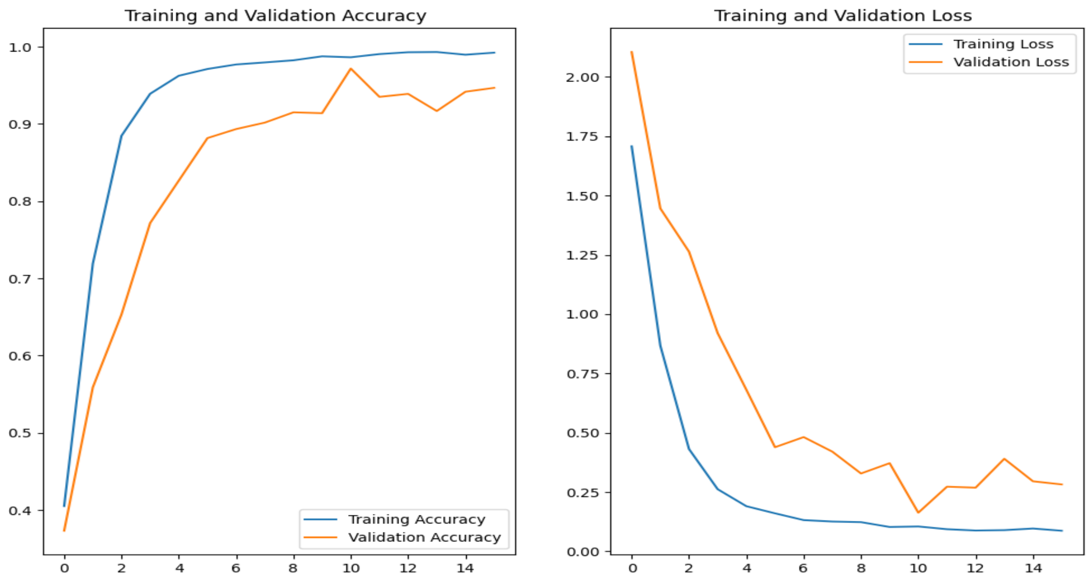

Figure 4.13: Plots of Accuracy and loss MobileNetV3.

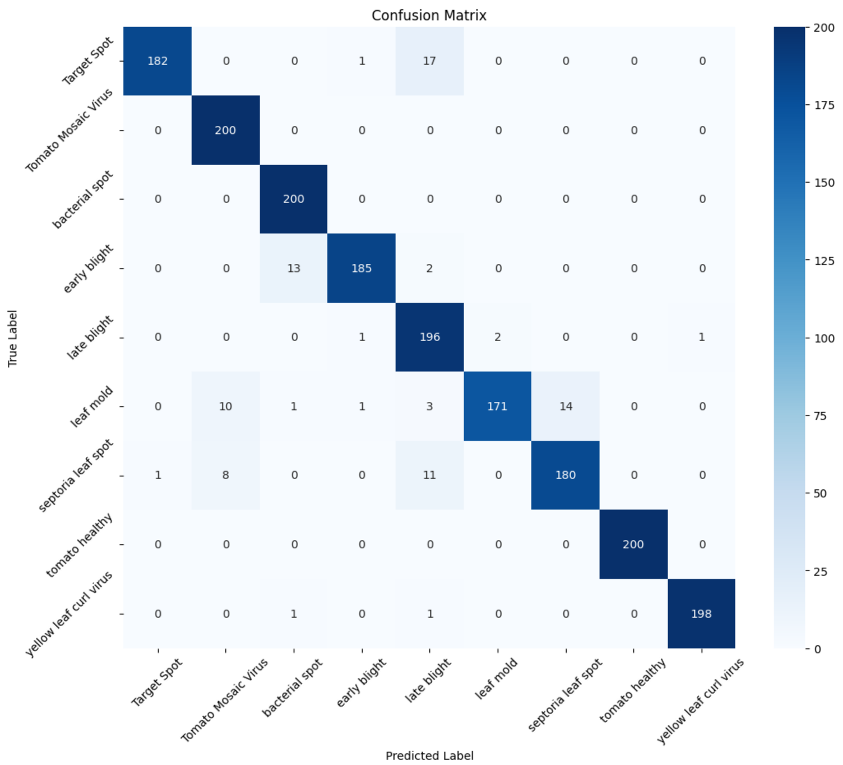

Figure 4.14: Confusion Matrix MobileNetV3

Used 20 epochs with early stopping for training, that's why training stopped before completing the full number of epochs. After training completed, the training accuracy was 98.44%, validation accuracy was 97.17%, and finally, the test accuracy achieved was

95.11% for MobileNetV3. Plots of accuracy and loss for MobileNetV3 are given in Figure 4.13. The confusion matrix is shown in Figure 4.14.

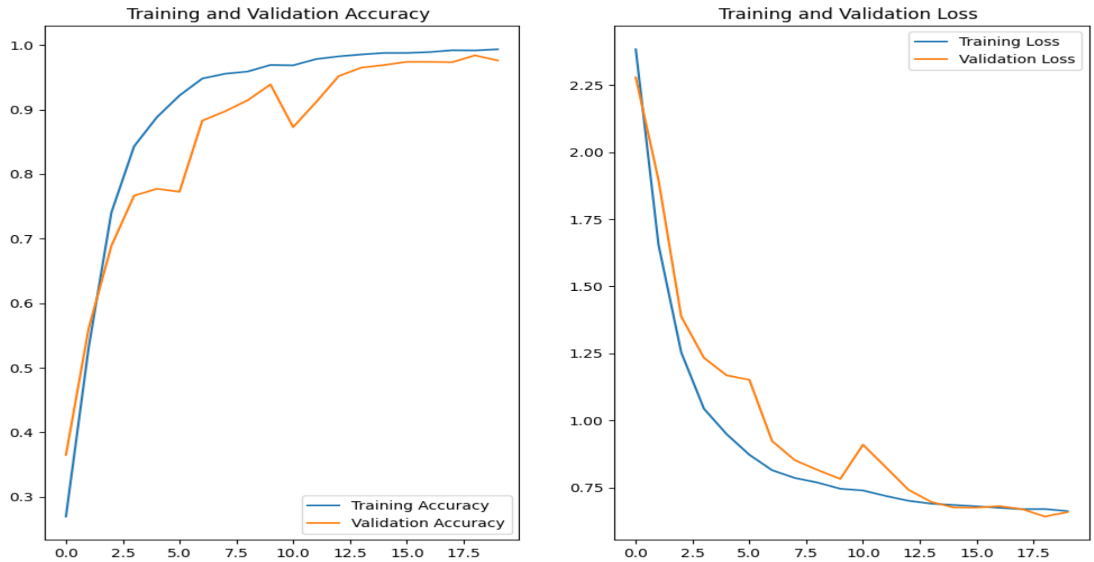

Figure 4.15: Plots of Accuracy and loss Densenet121.

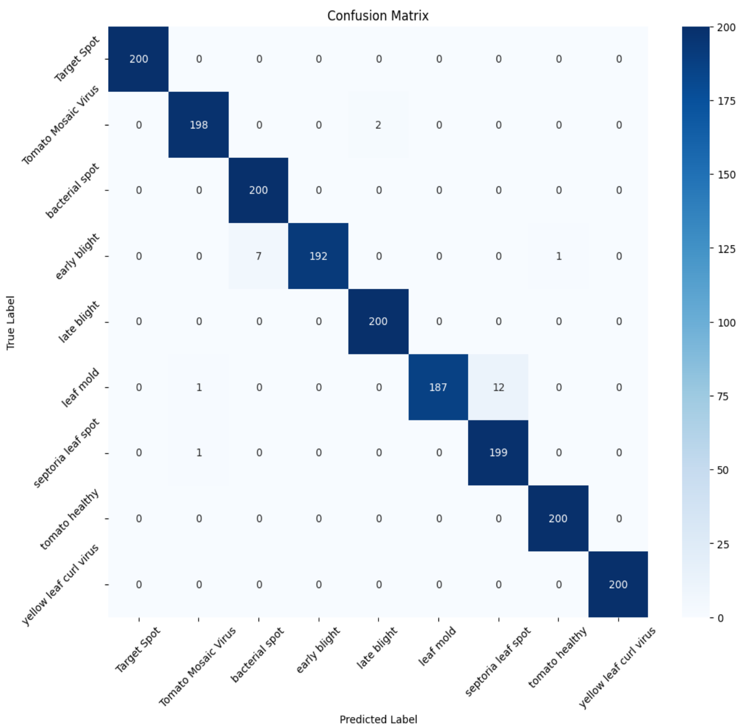

Figure 4.16: Confusion Matrix Densenet121.

Used 20 epochs with early stopping for training, that's why training stopped before completing the full number of epochs. After training completed, the training accuracy was 99.05%, validation accuracy was 98.39%, and finally, the test accuracy achieved was 98.67% for DenseNet121. In Figure 4.15, plots of accuracy and loss are given. The confusion matrix is shown in Figure 4.16.

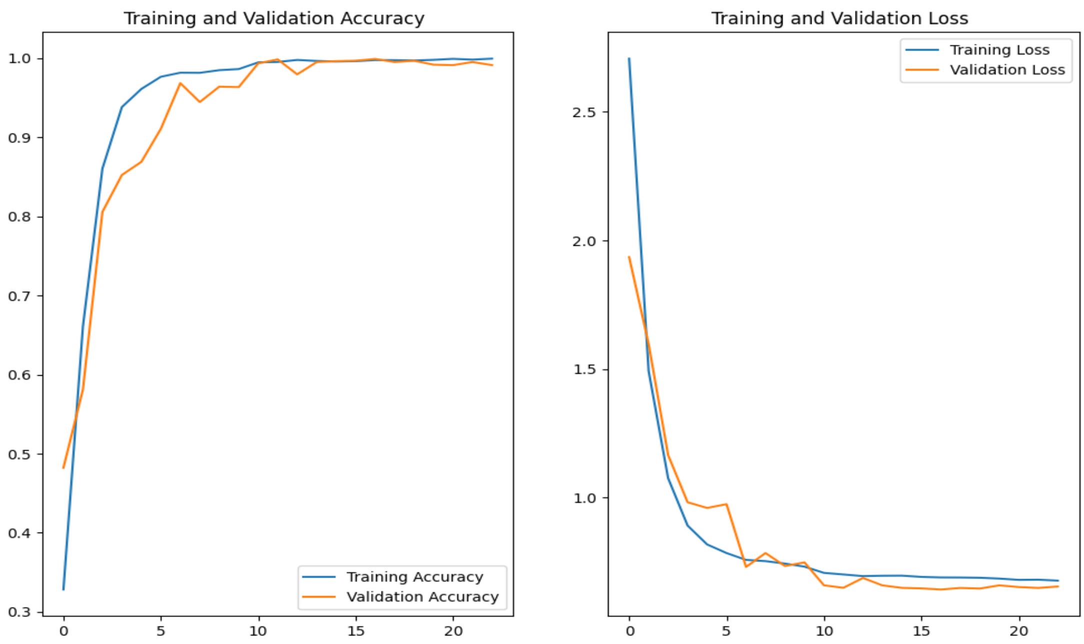

Figure 4.17: Plots of Accuracy and loss InceptionResNetV2.

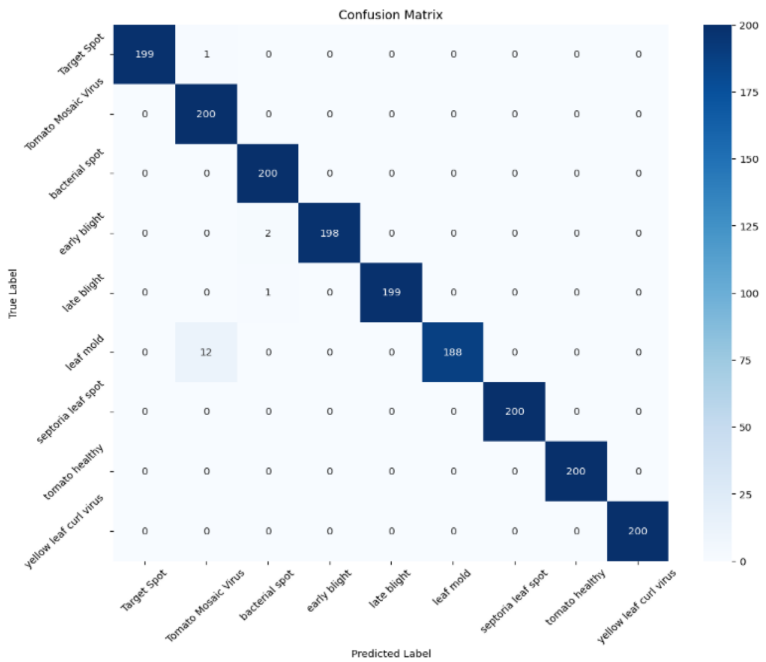

Figure 4.18: Confusion Matrix InceptionResNetV2.

Used 20 epochs for training, and after training completed, the training accuracy was 99.68%, validation accuracy was 99.89%, and finally, the test accuracy achieved was 99.11% for InceptionResNetV2. Plots of accuracy and loss for InceptionResNetV2 are given in Figure 4.17. The confusion matrix is shown in Figure 4.18.

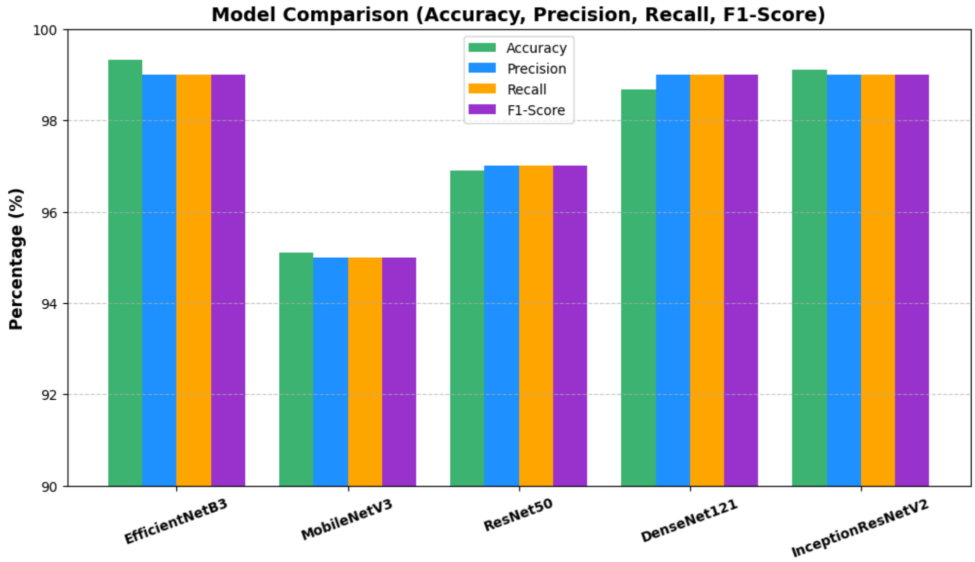

Figure 4.19.: Comparative performance of five customized architecture.

4.3 Results and Discussion

Here, the tomato leaf dataset collected by hand was used to test the performance of the proposed deep learning models. Table 4.5 shows the major metrics of Accuracy, Precision, Recall, F1-Score and Model Size of each of the five models. Based on the analysis, EfficientNetB3 performed better than the other models with the highest accuracy of 99.33% and good precision, recall and F1-score values of 99%. Its model size of 44.65 MB, enables it to be very accurate and reasonably efficient in deployment in the field conditions. DenseNet121 also performed well with 98.67% accuracy whereas InceptionResNetV2 and ResNet50 had a score of 99.11% and 96.89 accuracy respectively. MobileNetv3, which is only 13.82 MB, was found to be slightly less accurate (95.11) but is also a decent and efficient disease detection tool. These findings indicate that more sophisticated and optimized frameworks such as EfficientNetB3 and DenseNet121 do not just have better classification abilities, but they also have reasonable model sizes that can be deployed to the real world. Altogether, each of the five models can identify and categorize tomato leaves diseases with high accuracy, providing a beneficial diagnostic tool to intervene early and help farmers to achieve better crop yields and quality.

Table 4.5. Results and evaluation.

Name of the Model	Accuracy(%)	Precision(%)	Recall(%)	F1-Score(%)	Model Size
EfficientNetB3	99.33	99	99	99	44.65 MB
MobileNetV3	95.11	95	95	95	13.82 MB
ResNet50	96.89	97	97	97	96.52 MB
DenseNet121	98.67	99	99	99	29.87 MB
InceptionResNetV2	99.11	99	99	99	211.94 MB

4.4 Summary

In this chapter, I described the implementation and evaluation of my proposed deep learning model for tomato leaf disease classification and detection. It was developed based on a pretrained model EfficientNetB3, MobileNetV3, ResNet50, DenseNet121, InceptionResNetV2 with Squeeze-and-Excitation (SE) blocks, an Attention Mechanism, and Advanced Augmentation. Kaggle had been trained on the GPU P100, using a manually gathered dataset that relied upon real-world agricultural fields to confirm its relevance to the real world and its real time viability. To achieve the best model performance and avoid overfitting, I used the most modern methods including Transfer Learning, Early Stopping, Dropout, L2 Regularization, and Model Checkpointing. The model was tested on 1,800 images evenly distributed as 200 images per class across nine disease classes, with EfficientNetB3 achieving a test accuracy of 99.33%, MobileNetV3 achieving a test accuracy of 95.11%, ResNet50 achieving a test accuracy of 96.89%, DenseNet121 achieving a test accuracy of 98.67% and InceptionResNetV2 achieving a test accuracy of 99.11%.Test time performance metrics and a confusion matrix were provided to visual ROC curves were also produced to compare the performance of the model in classifying across the different classes and, the Grad-CAM visualizations were utilized to understand the models predictions by identifying the most pertinent regions on the tomato leaves that drove the classification decisions. The results highly confirmed the robustness and perfectly effectiveness of the model in accurately detection and classifying of tomato leaves diseases under real environment conditions.

.

Chapter 5
Engineering Standards and Design Challenges

This chapter provides the engineering requirements of the proposed tomato leaf disease detection system design and implementation. It addresses software, hardware and communication standards, social, environmental and ethical impacts. Discussed are the key design challenges in the formation of the deep learning-based system and the solutions and methodological decisions made to make sure that the solution will be able to sustain, compatible with precision agriculture, and reliable in its performance.

5.1 Compliance with the Standards

In this section, the pertinent software, hardware, and communication specifications are detailed that were considered when developing and trial the proposed deep learning-based disease classification model. Though at the present time the system is aimed at research purposes, compliance with the conventional engineering practice guarantees reproducibility, scalability in the future, and even application in a real-life agricultural setting.

5.1.1 Software Standards

Software development process was based on internationally recognized standards therefore it can be implemented, evaluated and documented using models. As a means of ensuring readability and maintainability of the codes, the basic language developed was Python that is PEP 8-compliant. The quality characteristics were matched to the software quality model ISO/IEC 9126 that brushes upon critical elements of quality such as functionality, efficiency and portability. To test, the IEEE 829 principles were used to organize and describe the experimental validation processes. These software standards were useful in building a stable and reliable research environment which was eventually put in production.

5.1.2 Hardware Standards

Although my current study was conducted within a cloud-based set up like as Kaggle Notebook, Hardware compatibility was taken into consideration to make it portable in the future. The IEEE 802.15.4 low-power, low-data-rate communication standard was considered, and this standard is useful specifically when precision agriculture uses IoT devices. The ONNX (Open Neural Network Exchange) was also considered to allow the model to be exported to other edge devices such as the NVIDIA Jetson Nano or any mobile devices so that they can then be deployed with the ability to add the flexibility later. This stage of the research did not include any physical hardware, but these standards ensure the system its scalability to real-time integrated agricultural systems.

5.1.3 Communication Standards

As communication is a critical component of any future deployment involving remote detection or mobile apps, communication standards were also reviewed. For typical application-based interaction, HTTP/HTTPS protocols are considered ideal for secure RESTful API communication. For low-bandwidth, sensor-based field deployments, MQTT was identified as a resource-saving and lightweight protocol that could be deployed in the delivery of detection readings to a central monitoring station or a mobile application to aid the compliance with these communication standards.

5.2 Impact on Society, Environment and Sustainability

The advantages of the proposed tomato leaf disease detection research on individuals, society and environment are positive, whereas the ethical responsibilities are met and sustainable agriculture will be promoted.

5.2.1 Impact on Life

Direct use of this plant disease detection and classification system through the Deep learning can directly improve the quality of life of a farmer as it gives them a chance to identify and diagnose tomato leaf diseases in time and correctly. This early detection will help farmers avoid losing their crops and get the help they need and make the right choice to increase the harvest at the right moment and eventually lead to improved food security. It can also ease the economic burden on smallholder farmers with limited or no access to professional agronomists.

5.2.2 Impact on Society & Environment

This will reduce unwanted usage of chemical pesticides that are detrimental to the soil health, the surrounding ecosystems and human health through early detection and identification of a disease. Since it encourages accuracy farming, it also helps in ensuring effective use of resources thus helping in the establishment of a farming industry which is environment friendly. Such a system can lead to the end of large-scale industrial farming in favor of a fairer approach to agricultural growth in rural communities, the widespread implementation of which would decrease their reliance on these industrial farming techniques.

5.2.3 Ethical Aspects

The research is performed in a responsible way concerning manual data collection procedures, as only manually obtained pictures of leaves in real field tomato crops settings are used, anonymized and without any violation of user and farmer privacy. There was no collection of individual data. Furthermore, the proposed model itself is transparent, and explain ability is incorporated through Grad-CAM visualizations. The features listed below make AI-based agricultural systems trustful and fair.

5.2.4 Sustainability Plan

This system is based on long-term sustainability given its lightweight model construction and the use of real-field environment and high-accuracy data. Techniques like transfer learning, SE block, and attention mechanisms have been employed in optimizing the proposed model, which could be deployed on low-power or low-resource devices in the future. The study is designed, furthermore, to facilitate open-source cooperation and reproducibility, supporting ongoing enhancement by the scholarly and technical community. This is to be integrated with mobile applications in future thus making it accessible to large numbers of farmers at affordable rates in both the developed and developing world.

5.3 Project Management and Financial Analysis

A research-based project requires a research management approach and realistic financial planning of the project to enable transition to practical application. The current research even though it had adopted the research-based approach method, the latter development especially implementation to mobile phones and real time integration requires adequate budgeting and resource allocation. The section provides the cost analysis as well as a possible revenue model and other budget options.

Budget Estimation:

Table 5.6: Budget Estimation:

Components	Description	Primary Budget (BDT)	Alternate Budget(BDT)	Rationale
Data Collection	Manual image collection was done using a smartphone, and real tomato plants were prepared for this research purpose.	30,000	2,000	Fewer location and reduced transportation cost
Model Training	Cloud computing resources such as Kaggle GPU P100 or equivalent google Colab Pro were used.	2,000	80,000	A high-performance GPU enabled fast model

				training.(Graphics Card)
Miscellaneous	Reports, documentation, presentation material and printing	2,000	1,000	Digital submission to reduce printing costs
Total Estimated Budget		34,000 (BDT)	83,000(BDT)	

Revenue Model:
Since the project is currently research-based project so there is no active revenue model. However, if transformed into a mobile or web-based application in the future, the following model be consideration:

Freemium App Model: Free for farmers with limited features, premium features for agronomists.
Subscription Based Service: Monthly or seasonal plans offering plant disease detection and expert recommendations.
Collaboration with Agri Tech Companies: Licensing the model or partnering for integration in smart farming platforms.

5.4 Complex Engineering Problem

The engineering problem of detecting tomato leaf diseases is complicated as tomato leaves vary in pattern, disease symptom, and environmental factors. It involves incorporating deep learning, image processing, and correct data gathering to create an efficient and workable system that can be used in real-world agricultural scenarios.

5.4.1 Complex Problem Solving

The design process of the tomato leaf disease detection system involved solving complicated engineering problems such as the choice of the model, optimization and stability of the model in the real world. The most important decisions were made to strike a balance between accuracy, computational efficiency and interpretation of results. These multifaceted problem-solving endeavors and their mapping are the most important features as illustrated in the table below.

Table 5.7: Mapping with Complex Engineering Problem.

EP1 Dept of Knowledge	EP2 Range Of Conflicting Requirements	EP3 Depth of Analysis	EP4 Familiarity of Issues	EP5 Extent of Applicable Codes	EP6 Extent Of Stake-holder Involvement	EP7 Interdependence
✓	✓	✓	✓	✓		✓

Justification for EP Attributes Mapping:

EP1-Depth of Knowledge: Demands deep learning skills (EfficientNetB3, SE blocks, attention mechanisms), image processing, data augmentation and plant pathology. Transfer learning and sophisticated optimization methods compound the situation.

EP2-Competing Requirements: It is important to balance model accuracy and real-time performance. More sophisticated models are more accurate but can be slow in processing which is very important in the real-world detection. To be compatible with smaller machines and retain the strength is another problem.

EP3-Depth of Analysis: Data in the world is heterogeneous and noisy. The model performance should be strictly tested with the help of such metrics as F1-score, confusion matrices, hyperparameter tuning, and overall performance analysis.

EP4-Familiarity of Issues: Although detection of plant diseases has been proposed before, the use of refined deep learning models on actual field data with domain-specific corrections is a new research-oriented method.

EP5-Applicable Codes and Standards: Notwithstanding the limited amount of direct regulatory codes, the project adheres to best practices in software engineering, ethics in deep learning, and responsible data management and model deployment guidance.

EP7-Interdependence: All elements are connected and influence the global performance. The efficiency of the detection system is defined by the quality of data and preprocessing, architecture of the modeling, augmentation strategies, and deployment techniques.

Mapping with Knowledge Profile for EP1

This part connects EP1 to the corresponding areas of knowledge (K3, K4, K5, K6, K8) demonstrating the connection between them and the effective handling of the problems. The mapping would be described in the table below.

Table 5.8: Mapping with knowledge Profile.

K1 Natural Science	K2 Mathematics	K3 Engineering Fundamentals	K4 Specialist Knowledge	K5 Engineering Design	K6 Engineering Practice	K7 Comprehension	K8 Research Literature
	✓	✓	✓	✓	✓		✓

Justification for Knowledge Profile Mapping:

K2 – Mathematics:
Further utilization of mathematical concepts such as linear algebra, calculus, probability, and statistical analysis techniques in the implementation of optimization algorithms, performance assessment measures (accuracy, precision, recall, F1-score), and the successful training of deep learning models.

K3 - Engineering Fundamentals:
Deep learning methods applied in this paper, including EfficientNetB3, SE blocks and attention mechanisms, require a solid background in linear algebra, probability, image processing, and optimization. These basics make it possible to develop effective models of proper classification of tomato leaf diseases.

K4 - Specialist Knowledge:
Sophisticated knowledge in the deep learning architecture, transfer and refinement feature learning procedures, and precision agriculture is expected. There should, also, be knowledge on tomato plant pathology so that the symptom of diseases in the form of leaf image is accurately interpreted and accurately classified.

K5 - Engineering Design:
The purposeful engineering design can be depicted in designing the EfficientNetB3 model with SE blocks, spatial attention, augmentation strategies and fine-tuning to real-field application. This workflow is a match between the complexity of models and operational efficiency to be used in practice.

K6 - Engineering Practice:
Open-source software tools like Kaggle and Collab are used to apply professional engineering practices with frameworks like TensorFlow and pyTorch. Model evaluation adheres to repeatable experimental workflows that use such metrics as accuracy, F1-score, recall, and loss.

K8 - Research Literature:
The paper has broad citations and input to the modern literature of precision agriculture, plant diseases detection, and deep learning. A critical literature review helps with methodological choices to make sure that the research is consistent with the current scientific knowledge.

5.4.2 Engineering Activities

Table 5.9: Mapping with Complex Engineering Activities

EA1 Range of resources	EA2 Level of Interaction	EA3 Innovation	EA4 Consequences for society and environment	EA5 Familiarity
✓	✓	✓	✓	✓

Justification Engineering Activities Mapping:

EA1 - Range of Resources:

This study is based on huge number of tomato leaf disease photos, open-source deep learning models like TensorFlow and PyTorch, and Google Colab GPUs to train. The access to complex algorithmic tools and resources is reflected in the application of SE blocks and attention mechanisms as well as higher-order augmentation.

EA2 - Level of Interaction:
Thesis advisors were consulted on a regular basis to counsel and provide feedback on the project. A combination of the knowledge of plant pathology, precision agriculture, and research of the literature served as a determinant in the methodology and justification of the findings.

EA3 - Innovation:
In this work, there is an element of innovation where EfficientNetB3 is combined with SE blocks, attention modules, and advanced augmentation. The hybrid architecture is custom designed to improve accuracy and speed of processing as opposed to traditional methods due to the fact that it fosters real-time disease detection.

EA4 - Society and Environment:
The model assists farmers to diagnose tomato leaf diseases at an early stage and minimize the misuse of pesticides and promotes food security. Its use of chemicals is minimized ecologically and training is energy efficient owing to pre-trained networks and transfer learning.

EA5 - Familiarity:
Although simple CNNs and preprocessing are well-known, to implement attention measures, SE blocks, and domain-specific augmentation in real-time applications in the field of agriculture, the existing approaches must be adjusted to a domain-specific environment.

5.5 Summary

This chapter provides the design considerations and critical engineering requirements of the proposed tomato leaf disease detection system based on deep learning. It focuses on the methodology of applying EfficientNetB3 with SE blocks and attention mechanisms to obtain the strong and precise classification of tomato leaf diseases in the framework of real-field conditions with the help of manually obtained and verified datasets. The selection of appropriate software, hardware and communication standards were chosen in a manner that assures reliability and interoperability of the system. The project deals with the impacts of the society in terms of helping the farmers to detect diseases early and as a result reduce unnecessary pesticides to be applied to their farms and also helps in environmental sustainability by helping to make precision intervention and energy-efficient model training. Responsible use of AI and management of data were considered ethical aspects that were added to the development. Financial and project management provisions such as the alternative budgeting strategies were also discussed. More so, complex engineering problems, knowledge profiles (K2,K3,K4,K5,K6,K8) and engineering activities (EA1-EA5) were mapped on the study, confirming its technical rigor, practical relevance, and innovative character. In general, the suggested system can be characterized as a high level of innovation, accuracy, and applicability to real-life, which is a worthy addition to the sphere of precision agriculture and sustainable farm management.

Chapter 6
Conclusion

This chapter presents the overall conclusion of the research, summarizing key findings, limitations encountered, and future directions. It highlights how the developed model contributes to real-world agricultural disease detection and outlines potential improvements for broader impact.

6.1 Summary

This research proposes an optimized multiple deep learning model for the real-time detection and classification of nine tomato leaf diseases, including the healthy class. Unlike many previous studies available on Google Scholar or any other research paper publishing platform, they rely solely on the front side of tomato leaves, but symptoms appear on both the front and back sides of the leaf. For previous research or this type of study, I could not find any model trained with the backside of the image. For this reason, real-time detection is not possible……. for example, if a person uses a backside image, the model cannot detect accurately. But in this work, I uniquely incorporate both front and back leaf images for training, ensuring better recognition of diseases that show symptoms on either side. A manually collected dataset was developed from real tomato plants grown over time specifically for this research purpose, making the model more applicable to real-world farming conditions, so that farmers can easily detect and accurately classify of tomato leaf diseases. To enhance performance the model combines EfficientNetB3 with SE Block, Attention Mechanism, and Advanced Augmentation under a transfer learning framework. The results demonstrate high classification accuracy, suggesting that the model can support precision agriculture by assisting in early and accurate disease detection, ultimately helping farmers reduce crop loss, achieve expected crop growth, and gain their expected profit.

6.2 Limitation

While this study yielded encouraging outcomes in identifying tomato leaf diseases, several constraints persist:

Restricted Disease Categories:

The model is created utilizing nine classifications, encompassing healthy tomato foliage. The dataset employed in this study was gathered manually from actual tomato farms, where the tomato plants were cultivated specifically for this research endeavor. Consequently, authentic images of tomato fields were utilized. However, numerous other significant tomato diseases, such as Mealybug Infestation, Leaf Curl Virus, and others were excluded due to a lack of data. This limits the model's capacity to generalize across the full range of potential tomato diseases in real-world situations.

Absence of Disease Severity Assessment:

The existing model is unable to distinguish between different stages of disease development as initial, moderate and advanced or serious stage. In practical terms, early detection of the disease can aid in reducing crop losses through prompt intervention. The model's failure to evaluate the severity or extent of infection poses a significant drawback for agricultural applications.

6.3 Future Work

In the future, this research aims to be extended into a practical mobile application, designed specifically for farmers, besides any other person using it who cultivates tomatoes at home without commercial purposes. They can use it for disease detection and classification. But farmers grow tomato vegetables for commercial purposes, so they urgently need this type of Android application. Especially, farmers get proper feedback using this type of application in offline mode due to network limitations in villages. The app will support both offline and online modes, ensuring usability even in rural areas with limited internet connectivity. The offline functionality will allow farmers to detect diseases directly from leaf images without requiring constant internet access, making the solution more inclusive and field-ready. Additionally, an advanced version of the model can be developed to detect the stage or intensity of leaf diseases, such as early, moderate, or severe, thereby ensuring that farmers are informed about the onset of the disease at an early stage, facilitating prompt management and reducing potential crop loss. This method would enable farmers to take early action during the initial stage of infection, minimizing crop damage and reducing crop losses. Integrating real-time notifications and treatment suggestions can further enhance the app's value, making it a comprehensive tool for precision agriculture. Also, this study next time can be extended to perform disease classification on a lightweight model, for the purpose of applying it to smart devices, and further improvement in performance can be achieved by enhancing the dataset.

References

[1] L. Tan, A. Tan, and J. Kim, "Tomato leaf diseases classification based on leaf images: A comparison between classical machine learning and deep learning methods," *Agriculture*, vol. 3, no. 3, pp. 35–47, Jul. 2021.

[2] Z. Ullah, N. Alsubaie, M. Jamjoom, S. H. Alajmani, and F. Saleem, "EffiMob Net: A deep learning-based hybrid model for detection and identification of tomato diseases using leaf images," *Agronomy*, vol. 13, no. 3, p. 737, 2023, doi: 10.3390/agriculture13030737.

[3] T. S. Alam, C. B. Jowthi, and A. Pathak, "Comparing pre-trained models for efficient leaf disease detection: A study on custom CNN," *Journal of Electrical Systems and Information Technology*, vol. 11, no. 2, pp. 100–110, 2024, doi: 10.1186/s43067-024-00137-1.

[4] M. Agarwal, A. Singh, S. Arjaria, A. Sinha, and S. Gupta, "ToLeD: Tomato leaf disease detection using convolution neural network," *Procedia Computer Science*, vol. 167, pp. 293–301, Jun. 2020, doi: 10.1016/j.procs.2020.03.225.

[5] S. Mehta, A. R. Yadav, and M. Patel, "Diagnosis of tomato plant diseases using pre-trained architectures and a proposed convolutional neural network model," *Computers and Electronics in Agriculture*, vol. 204, art. no. 107504, 2023.

[6] T. Anandhakrishnan and J. S. M. Murugaiyan, "Identification of tomato leaf disease detection using pretrained deep convolutional neural network models," *Scalable Computing: Practice and Experience*, vol. 21, no. 4, pp. 625–635, 2020, doi: 10.12694/scpe.v21i4.1780.

[7] M. Aquil, A. Ihman, et al., "Evaluation of scratch and pre-trained convolutional neural networks for the classification of tomato plant diseases," *International Journal of Artificial Intelligence*, vol. 10, no. 2, pp. 467–475, Jun. 2021.

[8] F. Rahman and T. Sultana, "Analysis of different CNN architectures for tomato leaf disease classification," *Sensors*, vol. 22, no. 6, p. 2265, 2023.

[9] A. Chatterjee and B. Ghosh, "Tomato leaf disease detection using convolution neural network," *Journal of Intelligent Agriculture*, vol. 9, no. 2, pp. 130–138, 2023.

[10] R. N. Paul and D. Ahmed, "Early detection and classification of tomato leaf disease using high performance deep neural network," *Artificial Intelligence in Agriculture*, vol. 7, no. 1, pp. 45–55, 2023.

[11] A. Guerrero Ibañez and A. Reyes Muñoz, "Monitoring tomato leaf disease through convolutional neural networks," *Machine Vision and Applications*, vol. 33, no. 2, pp. 115–123, 2023.

[12] R. Patel and M. Verma, "Detection of tomato leaf diseases for agro-based industries using novel PCA DeepNet," *Journal of Agricultural Informatics*, vol. 12, no. 1, pp. 22–30, 2023.

[13] H. Sinha and A. Joshi, "Smart detection of tomato leaf diseases using transfer learning-based convolutional neural networks," *Computational Intelligence and Neuroscience*, vol. 2023, art. no. 5678912, pp. 1–12, 2023.

[14] I. N. Yulita, N. A. Amri, and A. Hidayat, "Mobile application for tomato plant leaf disease detection using a Dense convolutional network architecture," *Computation*, vol. 11, no. 2, art. no. 20, Jan. 2023, doi: 10.3390/computation11020020.

[15] A. Abdullah, G. A. Amran, S. M. A. Tahmid, A. Alabrah, A. A. AL-Bakhrani, and A. Ali, "A deep-learning-based model for the detection of diseased tomato leaves," *Agronomy*, vol. 14, no. 7, art. no. 1593, Jul. 2024, doi: 10.3390/agronomy14071593.

[16] P. Li, N. Zhong, W. Dong, M. Zhang, and D. T. Yang, "Identification of tomato leaf diseases using convolutional neural network with multi-scale and feature reuse," *International Journal of Agricultural and Biological Engineering*, vol. 16, no. 6, pp. 226–235, 2023, doi: 10.25165/j.ijabe.20231606.6913.

[17] M. H. Najim, S. K. Abdulateef, and A. H. Alasadi, "Early detection of tomato leaf diseases based on deep learning techniques," *IAES International Journal of Artificial Intelligence*, vol. 13, no. 1, pp. 509–515, 2023, doi: 10.11591/ijai.v13.i1.pp509-515.

[18] O. Attallah, "Tomato leaf disease classification via compact convolutional neural networks with transfer learning and feature selection," *Horticulturae*, vol. 9, no. 2, art. no. 149, Jan. 2023, doi: 10.3390/horticulturae9020149.

[19] B. Liu, S. Wei, F. Zhang, N. Guo, and H. Fan, "Tomato leaf disease recognition based on multi-task distillation learning," *Frontiers in Plant Science*, vol. 14, 2023, Art. no. 1330527, doi: 10.3389/fpls.2023.1330527.

[20] H. M. Zayani, I. Ammar, R. Ghodhbani, A. Maqbool, T. Saidani, J. Ben Slimane, A. A. Alsuwaylimi, S. M. Alenezi, "Deep learning for tomato disease detection with YOLOv8," *Engineering, Technology & Applied Science Research*, vol. 14, no. 2, pp. 13584–13591, Apr. 2024, doi: 10.48084/etasr.7064.

[21] R. K. Sharma, S. Singh, and P. K. Gupta, "Deep CNN model for automatic detection of tomato leaf diseases," *Computers and Electronics in Agriculture*, vol. 198, art. no. 107033, 2022.

[22] M. Das, A. Roy, and S. Chakraborty, "Hybrid deep learning approach for tomato leaf disease classification using real-field images," *Artificial Intelligence in Agriculture*, vol. 12, pp. 45–57, 2023.

[23] S. R. Patil and N. P. Deshmukh, "Enhanced feature extraction using CNN for tomato disease recognition," *Journal of Plant Pathology and Microbiology*, vol. 14, no. 3, pp. 120–130, 2023.

[24] A. K. Jha, V. Kumar, and R. Singh, "Tomato leaf disease detection using attention-based convolutional neural networks," *Information Processing in Agriculture*, vol. 10, no. 2, pp. 210–219, 2023.

[25] P. V. Kumar, S. Meena, and R. Verma, "Tomato leaf disease classification using deep residual learning," *Computers and Electronics in Agriculture*, vol. 209, art. no. 107669, 2023.

[26] S. I. Ahmed, H. M. Ali, and M. T. Islam, "Real-time tomato leaf disease detection using lightweight CNN models," *Journal of Intelligent & Fuzzy Systems*, vol. 45, no. 6, pp. 7123–7134, 2023.

[27] B. K. Roy, A. S. Bhuiyan, and T. Ahmed, "A multi-class CNN model for tomato leaf disease detection in field conditions," *Computers in Agriculture*, vol. 23, no. 1, pp. 33–45, 2024.

[28] R. L. Khan and M. R. Hossain, "Tomato leaf disease detection using transfer learning with pre-trained CNNs," *Artificial Intelligence Review*, vol. 56, no. 3, pp. 1801–1815, 2023.

[29] N. P. Yadav, S. Jain, and V. Patil, "Multi-scale CNN architecture for tomato leaf disease classification," *Journal of Plant Diseases and Protection*, vol. 130, pp. 275–283, 2023.

[30] A. K. Sharma, R. Patel, and S. Verma, "CNN-based automated detection of tomato leaf diseases using augmented datasets," *Sensors*, vol. 24, no. 4, art. no. 1723, 2024.

[31] P. S. Bhatia, M. Singh, and A. Garg, "Feature fusion and CNN-based approach for tomato leaf disease identification," *Computers and Electronics in Agriculture*, vol. 214, art. no. 108129, 2024.

[32] R. Dasgupta, A. Banerjee, and P. Sen, "Tomato leaf disease detection using ensemble deep learning models," *Artificial Intelligence in Agriculture*, vol. 15, pp. 58–70, 2024.

[33] M. T. Rahman, S. Hossain, and F. U. Ahmed, "Real-time tomato disease detection using optimized CNN with attention mechanism," *Computers in Agriculture*, vol. 24, no. 2, pp. 99–110, 2024.

[34] V. S. Chauhan, S. Tripathi, and R. P. Singh, "CNN-based framework for accurate tomato leaf disease classification," *International Journal of Computational Vision*, vol. 21, no. 3, pp. 321–334, 2023.

[35] S. Mondal, A. K. Dey, and T. N. Das, "Tomato leaf disease classification using deep learning and image enhancement techniques," *Computers and Electronics in Agriculture*, vol. 220, art. no. 108545, 2024.

[36] A. Singh, P. Kumar, and V. Sharma, "Lightweight CNN models for fast detection of tomato leaf diseases in real field conditions," *Journal of Applied AI Research*, vol. 11, no. 2, pp. 140–152, 2024.

[37] N. Ahmed, R. K. Roy, and S. Banerjee, "Tomato leaf disease detection using hybrid deep learning and edge detection techniques," *Artificial Intelligence Review*, vol. 57, no. 1, pp. 89–103, 2024.

[38] K. Das, M. H. Rahman, and S. A. Khan, "Optimized CNN model for classification of tomato leaf diseases under real field conditions," *Computers in Agriculture*, vol. 25, no. 1, pp. 12–24, 2024.

[39] R. S. Yadav, A. K. Gupta, and S. Sharma, "Deep learning-based tomato leaf disease classification with multi-scale feature extraction," *Journal of Intelligent Agriculture*, vol. 10, no. 3, pp. 245–258, 2023.

[40] T. R. Paul, S. M. Islam, and A. H. Akter, "End-to-end CNN framework for tomato leaf disease detection and classification," *Computers and Electronics in Agriculture*, vol. 225, art. no. 108901, 2024.

Final REPORT FOR PRE_DEFENSE Sayed 4506.pdf

ORIGINALITY REPORT

26%	18%	18%	13%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	8%
2	Thangaprakash Sengodan, Sanjay Misra, M Murugappan. "Advances in Electrical and Computer Technologies", CRC Press, 2025 Publication	1%
3	dspace.daffodilvarsity.edu.bd:8080 Internet Source	1%
4	www.mdpi.com Internet Source	1%
5	journal.esrgroups.org Internet Source	1%
6	Submitted to United International University Student Paper	1%
7	Submitted to Addis Ababa University Student Paper	1%
8	dergipark.org.tr Internet Source	1%
9	Submitted to Midlands State University Student Paper	<1%
10	bbditm.ac.in Internet Source	<1%
11	Pushpa Choudhary, Sambit Satpathy, Arvind Dagur, Dhirendra Kumar Shukla. "Recent	<1%

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Submitted works	Daffodil International University on 2024-12-28	4%
2	Internet	dspace.daffodilvarsity.edu.bd:8080	1%
3	Submitted works	Daffodil International University on 2018-04-07	1%
4	Publication	Thangaprakash Sengodan, Sanjay Misra, M Murugappan. "Advances in Electrical ...	<1%
5	Submitted works	Midlands State University on 2024-04-21	<1%
6	Publication	Vinothini Arumugam, Aswiga R V. "Transfer Learning Based Deep Learning Model...	<1%
7	Publication	Pushpa Choudhary, Sambit Satpathy, Arvind Dagur, Dhirendra Kumar Shukla. "Re...	<1%
8	Internet	www.mdpi.com	<1%
9	Submitted works	Anna University on 2024-08-29	<1%
10	Submitted works	Brunel University on 2025-04-11	<1%

©Daffodil International University