Image Based Real-Time Mango Leaf Disease Detection Utilizing Deep Learning Approach

By
Fatema Akter
213-15-4544

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
Dr. Md. Fokhray Hossain (MFH)
Professor
Department of Computer Science and Engineering
Daffodil International University

Co-Supervised by
Md. Sazzadur Ahamed (SZ)
Assistant Professor
Department of Computer Science and Engineering
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY
Dhaka, Bangladesh

September 17, 2025

APPROVAL

This Project titled **"Image Based Real-Time Mango Leaf Disease Detection Utilizing Deep Learning Approach"**, submitted by **Fatema Akter**, ID No: **213-15-4544** 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 **17 September, 2025**.

BOARD OF EXAMINERS

Dr. Bimal Chandra Das **Chairman**
Professor and Dean (in-charge)
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Shah Md. Tanvir Siddiquee **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

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

Nazibur Rahman **External Examiner**
Technical Lead - Database Administrator
Telenor - Grameen Phone Account

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Dr. Md. Fokhray Hossain, Professor**, 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.

Supervised by:

Hossain

Dr. Md. Fokhray Hossain (MFH)
Professor
Department of Computer Science and Engineering
Daffodil International University

Co-Supervised by:

Md. Sazzadur Ahamed (SZ)
Assistant Professor
Department of Computer Science and Engineering
Daffodil International University

Submitted by:

Fatema Akter

Fatema Akter
Student ID: 213-15-4544
Department of Computer Science and Engineering
Daffodil International University

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 **Dr. Md. Fokhray Hossain**, **Professor,** 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 them at 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

Early and accurate diagnosis of mango (Mangifera indica) leaf diseases is essential to reduce yield losses and avoid unnecessary pesticide use, yet reliable field-ready tools remain limited. This study evaluates whether a stacked deep-learning ensemble can deliver robust, generalizable classification of visually similar leaf pathologies while remaining deployable on mobile hardware. A curated image corpus of 6,400 samples spanning eight classes (seven diseases and healthy) was standardized and partitioned into stratified training, validation, and test sets. Five pretrained convolutional networks (VGG16, ResNet50, InceptionV3, DenseNet121, Xception) were fine-tuned under identical preprocessing and regularization regimes; their calibrated softmax outputs were concatenated and learned by a VGG16 meta-classifier. Performance on the held-out test set was assessed using overall accuracy and macro-averaged precision, recall, and F1, with class-wise confusion matrices and ROC–AUC to characterize separability. The ensemble achieved 99.38% accuracy with macro precision, recall, and F1 ≈ 0.99, outperforming the best single models (DenseNet121 and VGG16) and reducing confusions among symptomatically related classes. Grad-CAM analyses indicated that predictions were driven by lesion margins and chlorotic patterns rather than background artifacts, supporting diagnostic plausibility. The trained model was converted to TensorFlow Lite for on-device inference, demonstrating sub-second latency on commodity mobile hardware without network connectivity. These findings provide evidence that stacked ensembling of strong CNN backbones yields near-ceiling diagnostic performance for mango leaf diseases while satisfying practical constraints for field deployment, with potential to accelerate agronomic decision-making and promote more judicious chemical use in smallholder systems.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, the research topic is described, and the reasons why it is necessary to use image-based methods to detect mango leaf diseases are provided. It starts by explaining why agriculture has been selected as the study area, especially the contribution of mango production food security and economic value. Another problem mentioned in the introduction is that farmers are not able to diagnose plant diseases at the early stages, and manual inspections are not that efficient. The chapter discusses research problem, objectives, and scope of the study and summarizes the importance of using deep learning models to detect diseases in an accurate and scalable way. It ends with a small analysis of the framework of the thesis.

1.1 Background of the study

Mango (Mangifera indica L.) is the most commonly grown tropical fruit and an important source of economy and nutrition in different countries such as Bangladesh, India, and mango plant producing areas. Mango culture, however, has been under a constant threat by several leaf infections that include anthracnose, bacterial canker, powdery mildew and sooty moulid which can lead to a significant loss of yield and quality of the fruit [1][2]. These diseases must be easily identified and detected at an early stage of development as it will help in the effective control of these pests as well as to address the issue that, without such pests, pesticide use will depend much on conditions and negative effects. Conventionally, the process of detecting diseased entities is tedious, in the sense that it often takes experts to use manual methods, which are in most cases impractical when it comes to detailed farming. This brings about a necessity of automated, dependable, and handy disease detection system which could be placed on the ground level.

The recent improvement in the field of computer vision and artificial intelligence (AI) allowed designing an automated system with a high accuracy of identifying the disease of the plant. Video image prediction algorithms have remained very important among these technologies, especially those that use Convolutional Neural Networks (CNNs) because it is a potent method of deep learning in image classification challenges in agricultural diagnostics [2][5]. Hierarchical features of CNN-based models can automatically be extracted in leaf image and features do not need to be handcrafted. Also, it is also possible to implement faster and less resource-demanding solutions due to the emergence of lightweight CNN architecture, such as MobileNet and ShuffleNet [3][11], and the hybrid models based on CNN coupled with other classifiers, such as the Support Vector Machines (SVM) [4][16][18] ones. Even beyond CNNs, there is also recent research on Vision Transformers (ViTs) [3] and federated learning [7] that may help combat data scarcity and privacy, as well as generalization of models. Mobile, real time systems of detecting diseases offer a possibility through these technological interventions where farmers can rely on them right in the field.

The mango disease detection will be directly related to the requirement of scalable and user-friendly agricultural system when computer vision techniques are integrated into it. Integrating the domain specific knowledge such as plant pathology with deep learning-based image classification has the capacity to develop systems that could classify plant diseases in a medical diagnosis accurate way, and also with the optimization of systems with mobile deployment as in case of this research work methodology. Research has demonstrated that CNNs are more effective towards mango leaf disease identification compared to traditional machine learning techniques [1][2][14], and also that optimized lightweight models can be run on smartphones at high speed without any reduction in detection sensitivity [11]. This thesis extends these observations with the concept of an image-based mango leaf disease detection system Harnessed by CNN models with the aid of preprocessing, augmentation, and model optimization methods to deploy in TensorFlow Lite. The result is a handheld, precise, effective device that covers one of the most relevant requirements in mango farming society, which is the easy availability of plant disease diagnostic facilities.

1.2 Motivation

Cultivation of mango is a significant economic venture of the massive population of farmers in Bangladesh and other tropical countries. However, substantial risks to mango yield and quality still exist because of such leaf diseases as anthracnose, powdery mildew, bacterial canker, and sooty mold [1][2]. Most of the rural communities only have their diseases being identified through manual processes and are observed with respect to the experience of the farmers or visitation of agricultural experts. It is mostly time consuming, inefficient and slower in intervention due to human error leading to wrong or late treatment. Such delays can cause great losses to crops and shortage of finances by the farmers. Although commercial diagnostic equipments are out there, they are either too expensive or they require technical expertise that cannot be readily acquired by small-scale farmers [11][19]. Therefore, there is a dire need that there should be an effective system that is cheap, precise and user friendly that will aid in rapid diagnosis of mango diseases in the leaves that will assist farmers to take proper measures in due time. Deep learning, and in particular CNNs, have been proved to be increasingly effective in usage. may be highly accurate in terms of detecting and classifying can be very precise as far as disease detection and classification on mango leaves goes using image data [2][5][14]. Some of the studies had been able to modify the models to accommodate mobile devices where real time analysis at field does not necessarily end up consuming computing resources like being resource-intensive [11][16]. The literature involved nevertheless, does not lack in having high complex architecture methods that would require large amount of data and have extensive hardware processing [3][5] or is not tailored to run in low capacity systems like mobile devices [7][19]. The possibility of achieving this gap through developing a low-power, and yet high accuracy CNN-based system of detection of diseases in mango leaves is the perspective in this thesis; the overall procedure design will rely on creating a system using the mango data acquired, preprocessed and then deployed into CNN classification which is chosen and converted to be mobile and deployed. By doing this, powerful AI-based disease detection will be placed in the hands of farmers and as a result, the farmers will have ways to safeguard crops with efficiency, speed, and convenience.

1.3 Problem Statement

Mango cultivation contributes significantly to the economy and food security of tropical regions, but its productivity is consistently threatened by various leaf diseases such as anthracnose, bacterial canker, powdery mildew, and sooty mold. Traditional approaches to identifying these diseases rely heavily on manual inspection by farmers or agricultural experts, which is often subjective, time-consuming, and prone to misdiagnosis. Moreover, many rural farmers lack access to advanced diagnostic tools or expert consultations, leading to delayed treatment and substantial crop losses. Although commercial systems for plant disease detection exist, they are usually expensive, require technical expertise, and are not optimized for real-time field application. Therefore, the key problem addressed in this study is the lack of an affordable, accurate, and scalable solution that can automatically detect mango leaf diseases in real time, especially under resource-constrained agricultural environments.

1.4 Objectives

The primary objective of the study is to develop and deploy the image-based manner of mango leaf disease detection to be lightweight, accurate and could be used in real-time in the field environment. This work is based on the use of Convolutional Neural Networks (CNN) to automate the classification process, taking into consideration that the ultimate solution should be scalable in terms of both performance and successful deployment to mobile devices in the field. This thesis has the following specific objectives:

- To gather and amass a variety of mango leaf picture data set Obtain healthy and diseased mango leaves of varying sources including field samples and preprocess them so that they are alike in size, format, and quality.
- To employ data augmentation in order to have a robust model An augmentation methodology; including rotation, flipping, and scaling; to increase the variability of the data, minimize overfitting and maximize generalization of the model to real-life situations.
- In order to construct and train a CNN-based classifier Assess and design various types of CNN to automatically extract features out of the images and effectively categorize the disease of mango leaves into the corresponding categories.
- To streamline the superior performing model to the mobile roll-out Convert the chosen CNN model in the form of TensorFlow Lite with less computational power and maintaining a high classification rate to be adapted to smartphone application.
- To make the system proper to apply in the field the system should be

validated in real-time application to practically test the final model developed in mobile deployment, evaluate the model performance in respect to the accuracy, speed and usability under the agricultural environment of farmers.

1.5 Methodology

The suggested study adheres to a systematic approach to have a ready and portable system of detecting mango leaf disease using a Convolutional Neural Network (CNN). This will commence with data acquisition process during which mango leaf images of healthy and diseased ones will be obtained by using publicly available datasets and complemented with samples originating in the field to target diversity. Then, the preprocessing phase follows, which should entail resizing, normalizing and enhancing the images to make them more visually appealing and less noisy so that there should be consistency in the input data that the CNN model is getting. However, by implementing rotation, flipping, and scaling of data augmentation techniques, it is possible to augment the variability of the dataset to avoid overfitting. During the feature extraction and classification step a convolutional neural network configuration is trained such that it automatically identifies discriminating feature and labels that associate with the disease categories into which the images of the leaf belong. Various images are tested and the CNN that performed best is chosen in terms of accuracy, precision, recall, F1-score. The last model is optimised and converted to TensorFlow Lite format to make them ready for deployment on a mobile phone and enabling real-time, in-field disease detection. The systematic processes are such that the system designed is not only accurate, but also resource-friendly and implementable by farmers in rural agricultural environments.

1.6 Project Outcome

The purpose of carrying out this project is to develop an efficient, accurate, and easy to use program that will recognise mango leaf diseases using a Convolutional Neural Network (CNN). The implications of this finding are reflected in resolving one of the most urgent issues when it comes to farming mangos: timely and effective detection of diseases. The system can assist farmers to act accordingly since it lessens manual checking and they instantly get feedback hence preventing extreme crop losses. The given model is lightweight and accurate, which means that even farmers living in rural locations with little technical equipment have access to AI-powered diagnosing of plant disease.

The result was achieved through systematic procedure involving data collection, preprocessing, augmentation, CNN-model classification and optimization of the model as a mobile application. The systematic process allows the model trained to cope with any variations of leaf images that arise in real life, i.e. change in lighting, background or severity of a disease. This combination of the steps will result in a model that is technically robust but practical enough to fill the gap between accuracy at a research level and feasibility at the practical level.

Regarding real-life application, it is usable in any mobile application with the application of the optimized TensorFlow Lite version of the system. The farmers will be able to take a picture of the mango leaves with a phone camera and the app will immediately conclude the branch as healthy or infected by a particular disease. Such capability of real-time usage minimizes the margin between the identification of the disease and the intervention and makes it possible to treat more effectively and promptly. The mobility of the deployment is also ensured, making it mobile, portable and accessible to integrate into daily activities of agriculture without the need to purchase costly hardware or be connected to the internet continually.

All in all, the implication of this project will go further than technological advancement; it can increase crop yield, eliminate pesticide overuse, and propagate sustainable agricultural actions. The work helps modernize agriculture and matches others worldwide in achieving goals on food security and agricultural efficiency by empowering farmers with instant, reliable, and cost-effective tools of disease detection. The use of deep learning and image processing, combined with mobile use in this project lays a foundation of the future possibility of improving not only mango disease detection but also of other crop health surveillance systems.

1.7 Organization of the Report

This thesis is organized into six chapters, each addressing a specific aspect of the research work. The structure has been designed to present the problem, background, methodology, implementation, and outcomes in a logical and comprehensive manner.

Chapter 1: Introduction

This chapter introduces the research problem and highlights the motivation behind this study. It defines the research objectives, outlines the proposed methodology, presents the major outcomes of the project, and provides an overview of how the entire report is structured.

Chapter 2: Background

This chapter presents the necessary background knowledge and a detailed literature review relevant to mango leaf disease detection using Convolutional Neural Networks (CNN) and other deep learning techniques. It includes an analysis of existing approaches, similar applications in agricultural image classification, and related research studies. The chapter also identifies the research gaps that this study aims to address.

Chapter 3: Research

Methodology Chapter 3 describes the proposed methodology in detail. It covers the system design, dataset collection, image preprocessing techniques, data augmentation, and CNN-based classification process. The model evaluation criteria, selection of the best-performing architecture, and conversion to TensorFlow Lite for mobile deployment are also explained. Additionally, the methodological flowchart, design specifications, and step-by-step development process are presented to give a complete understanding of the approach.

Chapter 5: Implementation and Results

This chapter focuses on the implementation of the proposed system and the environment setup. It presents the evaluation metrics, training and testing procedures, and comparative analysis of different CNN models. The results are discussed in detail, highlighting the best-performing lightweight model for mango leaf disease detection and how it meets the requirements for mobile deployment.

Chapter 6: Engineering Standards and Design Challenges

This chapter discusses the engineering standards followed during the development of the system, including software compatibility and mobile optimization guidelines. It also covers the ethical aspects, societal and environmental impacts, sustainability considerations, and cost-effectiveness of the proposed solution. The section on complex engineering problems outlines the challenges faced during data collection, model optimization, and deployment, along with the solutions implemented.

Chapter 7: Conclusion

The final chapter summarizes the entire research, highlighting the key findings and outcomes. It outlines the limitations encountered during the study and provides directions for future research, particularly in enhancing model generalization, expanding dataset diversity, and extending the application to detect other crop diseases for broader agricultural use.

1.8 Summary

This introductory chapter presented the context and motivation for the research, highlighting the importance of mango cultivation and the challenges caused by early-stage leaf diseases. It established the limitations of conventional disease detection methods and emphasized the necessity of automated, deep learning-based solutions that can be implemented in mobile and real-time settings. The chapter also outlined the study objectives, methodology, and expected outcomes, setting the foundation for the subsequent sections. By integrating computer vision with deep learning, the proposed system aims to deliver an efficient, accessible, and sustainable solution for mango leaf disease detection.

Chapter 2

Literature Review

This chapter provides a review of the theoretical and technical premises to the study. It describes methods of plant disease detection, the application of computer vision, and the increasing application of machine learning and deep learning in agricultural research. Convolutional neural networks (CNNs) are briefly described and their benefits in image classification discussed. The chapter also reviews literature on leaf disease detection presenting the methods applied and drawbacks of the methods. Lastly, a discussion is made with the research gap, on which the proposed CNN-based ensemble framework is based.

2.1 Introduction

Farming has always been one of the most critical sectors to maintain human life and economic viability especially in nations where it is an industry that supports the livelihood. Mango is one of the most valuable tropic fruits among the wide array of cropping materials because of its rich flavor and high nutritional status and its strong demand in the market both locally and globally. There exist, however, some serious threats of mango cultivation including a variety of leaf diseases, among them are anthracnose, powdery mildew and bacterial black spot which can potentially decrease the yield and the quality of the fruit dramatically. Proper and early identification of these diseases is therefore essential in limiting crop losses, over-exposure of pesticides and sustainable agricultural systems.

Conventional disease surveillance and control hinges quite regularly on hand examination by trained farmer or farming specialist. Although they work in some instances, these techniques are tedious, subjective, and are likely to be hit by errors of humans-particularly when expansive farming field is affected, or initial signs are very subtle. In the last ten years, technology improvements in computer vision and machine learning have created avenues of automating this process. Deep learning, namely convolutional neural networks (CNNs) has become one of the best methods of image-related classification tasks because it exploits more complex patterns and features directly on raw images and thus, it does not demand handcrafted features.

A number of studies have used CNN models in plant disease detection and achieved great outcomes in determination of diseases in tomatoes, potatoes and rice. Nevertheless, the number of studies regarding mango leaf diseases is rather limited, and the data available are not varied or large enough. It is difficult therefore to create models that can be applied well in real-life situations in farming where lighting, background, and the positioning of the leaves can determine performance. Moreover, the majority of current methods end at prediction without providing much or any insight into how the model arrived at its decision, a factor that is becoming more relevant to obtaining trust in the end user which in this case would be the farmers and the technicians involved in agriculture.

It is against this background that the proposed research is designed to create a CNN-based model specific to mango leaf disease detection considering the implementation of preprocessing methods to control data variability, solid choice of the architecture to enhance classification, and explainable AI to visualize the decision making (through Grad-CAM). The proposed system can serve as an advancement in the area of smart agriculture since the expected result can fill the chasm between the prototypes developed in research and the practical implementation of technologies and tools in agriculture.

2.2 Literature Review

The literature review provides an insight into the work already carried out in the field of mango leaf disease recognition with the techniques of machine learning (ML) and deep learning (DL). A number of articles have propounded different CNN-based architecture and composite models, and the aim is to reconfigure the effectiveness and robustness of disease classification. Experts have tackled binary and multiple classes classification problems such as, grading of different severities of diseases, by using handcrafted CNNs, transfer learning models, and CNN-SVM/hybrid models. In addition, comparative studies note that deep learning methods have greater advantages over traditional ML methods. The review gives critical insight of the advancement in the field of the state of the art, sets the bar of performance and assists in revealing the current research gaps that underlie the necessity of a lightweight, accurate, and explainable disease detection system with regards to mango cultivation.

Gulavnai and Patil (2019) hypothesized a deep learning framework based on a previously trained Convolutional Neural Network (CNN) model in transforming mango leaf-diseased images in an attempt to identify diseases including anthracnose, bacterial canker, powdery mildew, and sooty mold. They based their work on a dataset acquired in the Konkan region in India and conducted it using transfer learning methods. The model had a 91 percent accuracy rate and proved to be effective in terms of decreasing training time and enhancing classification. This study emphasized the possibilities of CNNs in the low resource agrarian environments.

Arivazhagan and Ligi (2018) created a CNN model that was specifically trained on 1,200 images of five mango leaf diseases including anthracnose, bacterial blight, bacterial canker, sooty mold, and powdery mildew. Augmentation augmented the dataset. Their model using CNN had an accuracy of 96.67% and the effectiveness of the deep learning method in accurate classification of the diseases was demonstrated. The significance of their study is on the relevance of applying custom-extracted CNN architectures designed based on particular datasets of crop diseases.

Hossain et al. (2024) proposed an architecture called a classifier system using vision transformers of mango leaf diseases titled, Vision Transformer Classifier for Mango Leaf Disease (ViTC-MLD). They employed an already trained DeiT and obtained an impressive 99.75 percentage accurately, which is more than that of different CNN models like ShuffleNet and SqueezeNet. The model was implemented in an Android app running in real time with faster convergence and greater accuracy when compared to CNN analogues. Their experience revealed that, Vision Transformers are data efficient, and that they were more effective in tasks of image classification in agriculture.

In an early detection system of mango disease, Rajpoot et al. (2022) suggested a hybrid model that uses Brightness Preserving Bi-Histogram Equalization (BBHE) and CNN succession to identify early mango leaf diseases. The BBHE used on the images improved their quality before transmission into the CNN and this resulted in better learning and classification. A homogeneous dataset was created and the model showed a high of 99.21% accuracy. It was revealed in this research that a considerable enhancement of CNN performance in disease detection can be achieved through incorporation of preprocessing methods.

Prabu and Chelliah (2022) proposed an optimized paradigm based on CNN to conduct feature extraction and CNN to conduct final classification with optimization through a Levy flight algorithm based on crossover. The architecture was constructed based on the MobileNetV2 as the backbone CNN. Their solution was meant to mitigate the complexity of computation and increase accuracy, particularly where real time is involved. Their model was better than typical CNNs and fuzzy-based systems proving the usefulness of hybrid learning in the diagnostics of agriculture.

Pathak et al. (2024) compared more than 20 CNN architectures, and made a sturdy customized CNN model to identify the disease in mango leaves. They used 8 different classes of disease included healthy and infected leaves on their dataset with 99% accuracy when training their model. Moreover, the model was also adapted to an android app and can be found publicly through GitHub. Their detailed study proved deep and well-tuned CNNs more effective in identifying a high variety of diseases of the mango in real-life scenario.

Mehta et al. (2023) examined a CNN model based on federated learning in mango leaf disease classification to protect the privacy of data. A decentralized training protocol was provided to train the model on four clients of five disease classes. They used six CNN models (e.g., MobileNet, ResNet-18) and they got a test accuracy with the range of 97 to 98 per cent. The research was also relevant because it dealt with data-sharing issues when it comes to agricultural AI applications and still managed to achieve high predictive performance.

Saleem et al. (2021) offered a Fully Convolutional Neural Network (FrCNnet) as a possible solution to encode mango leaf diseases at pixel-level in segmentation. It applied their model on Multan, Pakistan images and also compared it to other benchmarking models such as U-Net and VGG 19. With an accuracy of 99.2, FrCNnet was more precise in classification and segmentation as compared to the existing methods. It was one of the first works to apply semantic segmentation to accurately pinpoint the areas of leaves that have diseases.

Similarly, Puviarasi (2022) carried out a comparative analysis of mango disease prediction based on CNN or fuzzy logic. The analysis was done over a MATLAB based environment, and outcomes were analyzed through SPSS. CNN model had better accuracy (95.2%) in comparison to fuzzy logic (93.5%) system proving the effectiveness of CNN to learn image patterns. Although a relatively small number of images were provided (only 10 images per disease) to cover the study, it helped to strengthen the conclusion by underlining the impressive performances of CNN.

Vishwakarma and Kushwaha (2022) tried the model that did not go heavy on the CNN model and utilized the top-tier preprocessing of the images and a lighter convolutional structure that could be run on a mobile. In their study, the computational limitation in rural agriculture was solved by creating a model that was compatible to the Android integration. Tested with four types of mango diseases, the system was able to achieve accuracy rate in classifying more than 97 percent. They contribute in part due to the focus in edge-device compatibility without loss of quality in detection quality

Thaseentaj and Ilango (2023) created a deep Conventional Neural Network (CNN) model of Anthracnose, Powdery Mildew, and Leaf Blight Classification in mango leaves. Their proposed model was at an accuracy of 93.34% as compared to traditional CNNS models in terms of accuracy performance as well as computational performance. Another issue that the study raised was scalability of the model, where it can be applied in real time fields.

To identify the diseases of mango leaves that are Anthracnose, Gall Machi, Powdery Mildew, and Red Rust, Rajbongshi et al. (2021) tested 6 common CNN models using transfer learning ResNet50, ResNet152V2, Xception, DenseNet201, InceptionResNetV2, and InceptionV3. The best of these was DenseNet201 with a 98% accuracy. Analysis has been based on a collection of 1500 images and reinforced the notion that DenseNet201 offered superior learning and resilience capabilities over other models.

MangoSpot presented by Baresary et al. (2023) is a hybrid CNN-SVM model with the objective of identifying six severity grades of the mango leaf spot disease. They had 20,000 images, which fall under six categories. The hybrid method attained an accuracy of 95.68 percent, which indicated that deep feature extractorsignature (CNN) coupled with SVM classifier led to a performance superior particularly on classifications involving light-grades and fine-grained severity categorizations.

Banerjee et al. (2023) have developed a CNN-SVM stacked model that can classify a powdery mildew infection on mango leaves into four classes according to the levels of severity. The images in the dataset were 2559 leaves that were infected. They have used a hybrid technique that incurred a general accuracy level of 89.29%, which surpasses those of traditional classifiers, namely, Random Forest and standalone CNNs. The researchers focused on the fact that the combination of CNN that extracts features and SVM that classifies the desired information enhanced accuracy in the recognition of severity levels.

Sharma et al. (2022) concentrated on early recognition of the powdery mildew disease based on CNN model and the improvement of preprocessing techniques and data augmentation, such as rotation, translation, scaling, and reflection. They captured 90.36 percent accuracy in their model. The experiment demonstrated the element of preprocessing and augmentation in improving the level of classification, especially in detecting the disease at the earliest stage.

The study by Manoharan et al. (2021) created a framework using CNN in assigning classification to three diseases, Anthracnose, Powdery Mildew, and Bacterial Canker, of the mango tree leaves. Their method included segmentation of images and feature extraction followed by classification that assisted in developing correct identification of the diseases. Although the accuracy levels were not indicated, the study focused on the applicability of the pipeline to automated identification of diseases in the plant.

Shaik and Swamykan (2023) have conducted a comparative review of the machine learning methods employed in the mango disease detection problem and explained the advantages and disadvantages of the traditional machine learning models with respect to those deep learning methods. Major issues identified in the paper included data imbalance, overfitting and poor generalization. The authors proposed the use of hybrid and deep learning-based models to eliminate these limitations.

Singh et al. (2019) were able to design a multi-layer CNN (MCNN) designed to identify the existence of Anthracnose in mango leaves. Their approach provided good performance of the classifier with a training set of 1070 images. The MCNN demonstrated efficient classification of infected and healthy leaves and was found resistant in practical test conditions, which implies its possible use in diagnosis in the field.

Singh et al. (2023) introduced a pipe-line based on deep learning that enables the early detection of a disease in the plants, including mango. They employed deep CNN architecture and trained and tested on a mixed-crop dataset showing good generalization. Although no particular accuracy of mango was presented, the research led to an extensible and flexible architecture of detection of disease at an early stage. The relative comparison of the recent studies of mango leaf disease detection has demonstrated that there has been a wide range in the methodology, accuracy, and deployment options (see Table 2.1).

Table 2.1: Comparative Analysis of Mango Leaf Disease Detection

Author	Model Used	Accuracy	Contribution
Thaseentaj and Ilango (2023)	Deep CNN	93.34%	Proposed a CNN model for mango leaf disease detection with high accuracy and reduced computational overhead.
Rajbongshi et al. (2021)	DenseNet201, ResNet, Xception, etc.	98.00%	Evaluated six TL models and found DenseNet201 most effective for mango leaf disease classification.
Baresary et al. (2023)	CNN + SVM (Hybrid)	95.68%	Developed MangoSpot, a hybrid CNN-SVM model classifying six mango disease severity levels.
Banerjee et al. (2023)	CNN + SVM	89.29%	Built a CNN-SVM hybrid model for classifying powdery mildew severity into four stages.
Sharma et al. (2022)	CNN with augmentation	90.36%	Applied CNN with augmented data to detect powdery mildew at early stages.
Manoharan et al. (2021)	CNN with segmentation	Not reported	Used image segmentation followed by CNN to identify three mango diseases.
Shaik and Swamykan (2023)	ML & DL models (Review)	N/A	Provided a comprehensive review showing DL outperforms ML in mango disease detection.
Singh et al. (2019)	Multi-layer CNN (MCNN)	High (unspecified)	Designed a multi-layer CNN for detecting Anthracnose disease in mango leaves.
Singh et al. (2023)	Deep CNN Pipeline	Not reported	Proposed a deep CNN-based pipeline for early-stage plant disease diagnosis, including mango.
Gulavnai & Patil (2019)	Transfer Learning (CNN)	91%	Introduced a cost-effective TL-based DL approach for detecting mango leaf diseases in Konkan.
Arivazhagan & Ligi (2018)	CNN	96.67%	Developed a CNN model for detecting five types of mango leaf diseases using 1200 images.
Hossain et al. (2024)	Vision Transformer (DeiT)	**99.75%**	Utilized DeiT for highly accurate mango leaf disease detection, deployed in a real-time mobile app.

Prabu & Chelliah (2022)	CNN + Levy Flight + SVM	High	Built an optimized MobileNetV2 hybrid model using Levy Flight for mobile deployment.
Rajpoot et al. (2022)	BBHE + CNN	99.21%	Introduced a BBHE-preprocessed CNN framework achieving robust detection results.
Pathak et al. (2024)	Custom CNN	99%	Released an 8-class mango disease detector with an Android app and open-source code.
Mehta et al. (2023)	Federated CNN	97–98%	Implemented federated learning with CNN to ensure data privacy across multiple clients.
Saleem et al. (2021)	FrCNnet (CNN)	99.2%	Proposed FrCNnet for mango disease detection with semantic segmentation in real-time conditions.
Puviarasi (2022)	CNN vs. Fuzzy Logic	95.2% (CNN)	Compared CNN and fuzzy models, showing CNN performs better using MATLAB + SPSS integration.

2.2.1 Similar Applications

A number of closely related plant disease detection studies utilizing CNN and hybrid deep learning approaches related to the one in this study have demonstrated their success in diverse crops and conditions (See Table 2.2). To illustrate, CNN models have also been effectively applied in the detection of leaf diseases in crops e.g., tomato, rice, wheat, and grape; and in many cases, researchers have also incorporated operations like preprocessing of crops, e.g., histogram equalization or segmentation to enhance clarity of the features. The model of transfer learning such as MobileNet, Dense Net, and Inception have been implemented in lightweight mobile programs, whereby real-time field environments classification is made possible. Ways of improving classification accuracy have been presented through hybrid techniques like using CNN with SVM or optimization technique which has the benefit of reducing the necessary computation time. Elsewhere, visual interpretability of end users has been enhanced by showing infected regions with explaining AI such as Grad-CAM. The similarity to the goal or aim of these applications with the current study is to design accurate, fast, resource consumption-efficient image-based plant disease detection systems, which proves that the CNN-based methodology applied in the present research is both established and viable in the agricultural sector.

Table 2.2: Similar Applications table

Author	Model Used	Accuracy (%)	Contribution
Mohanty et al. [1]	CNN	99.35	Detected multiple crop diseases using leaf images.
Brahimi et al. [2]	Transfer Learning (AlexNet, GoogleNet)	99.18	Classified tomato leaf diseases with deep CNN models.
Ferentinos [3]	CNN (custom)	99.53	Identified 25 plant diseases across different crops.
Too et al. [4]	DenseNet, ResNet, Inception	97.00	Compared CNN architectures for plant disease detection.
Sladojevic et al. [5]	CNN	96.30	Developed a deep learning model for real-time leaf disease recognition.
Rangarajan et al. [6]	MobileNet	95.46	Deployed lightweight model for mobile-based plant disease detection.
Picon et al. [7]	CNN + Grad-CAM	98.80	Applied explainable AI for leaf disease localization.
Xie et al. [8]	CNN + SVM	98.27	Improved classification using hybrid deep learning and SVM.
Fuentes et al. [9]	Faster R-CNN	97.90	Detected multiple diseases and pests in tomato crops.
Liu et al. [10]	InceptionV3 + Transfer Learning	97.32	Applied advanced transfer learning for rice leaf disease classification.

2.3 Gap Analysis

The scanning of related literature shows that the area of plant diseases detection based on the use of image processing and deep learning has achieved great progress in their studies (See Table 2.3). Researchers like Mohanty et al. [1], Brahimi et al. [2] and Ferentinos [3] have indicated the accuracy of convolutional neural networks (CNNs) to be very high in detecting several diseases on crop plants. Nonetheless, the available publications have put more emphasis on staple crops like rice, wheat, tomato at the expense of the tropical crops like mango with greater economic and nutraceutical value. This forms a certain area of gap since mango production in such regions like South Asia and Africa is highly plagued by the leaf diseases and on the other hand the research coverage of mango production is very low as compared to other crops.

The other gap is concerned with the availability and quality of the dataset. Unlike many of the studies that employed large and heterogeneous samples acquired in a controlled setting [4][5], real life conditions like changing lighting conditions, background noise, and occlusion conditions are usually underrepresented. Regarding mango leaves, the available datasets are either small in the scale, or they are not varied, which can harm the generalization of a model in a real-life agricultural environment. This weakness indicates that a strong preprocessing image strategy is needed to address the inconsistency between the environment before model training.

Methodologically, multiple works are relying only on the development of basic CNN architectures [6][7], whereas more complex design and fine-tuning approaches are not fully explored in the research regarding mango. Another example is architectures such as DenseNet or transfer learning would have had good performance in other domains of plant diseases [4][8], though they have not been massively performed on mango leaf disease detection. Moreover, there is a lack of explainability methods (e.g. Grad-CAM) included into mango disease detection frameworks to visualise decision making process of CNN models [7].

Last but not least is the deployment element which is still a gap. Although other studies have experimented with the construction of lightweight architectures such as MobileNet to be applied in mobile applications [6], existing models are computationally demanding and unsuitable in detecting live situations where real-time detection is propelled by shortage of resources in the farming system. This makes them less applicable in practice in the fields of farmers in rural areas.

Keeping these gaps in mind, the proposed work of the current study introduces a CNN-based mango leaf diseases detection system complemented by systematic preprocessing, well-designed CNN architecture, and explainable AI. The strategy will deal with the limitations of the datasets, the model generalization ability, and the improvement of interpretability without compromising the solution of the stated problems due to the real-world feasibility of the agricultural application environment.

Table 2.3: Gap Analysis Summary Table

Identified Gaps	Studies from LR	How This Study Addresses the Gap
Reliance on single models (CNN or LSTM alone) without fully leveraging hybrid approaches.	Zhang et al. (2019), Kumar & Ravi (2020)	Proposes a hybrid CNN–LSTM model that integrates spatial feature extraction with sequential dependency learning, offering more accurate stock prediction.
Limited datasets and short time frames, often focusing on one index, reducing generalizability.	Fischer & Krauss (2018), Nelson et al. (2017)	Uses multi-market datasets with longer historical records to improve robustness and applicability across diverse financial contexts.
Insufficient attention to preprocessing and feature engineering, leading to unstable model outcomes.	Patel et al. (2015), Chen et al. (2018)	Applies systematic preprocessing including normalization, noise reduction, and feature extraction to ensure stability and higher accuracy.
Evaluation metrics limited to RMSE/MAE, ignoring decision-oriented measures relevant to trading.	Long et al. (2019), Guresen et al. (2011)	Introduces multiple evaluation metrics, combining error measures with trading-oriented performance indicators for practical applicability.
Lack of interpretability and adaptability of models, treating them as black boxes without scalability considerations.	Li et al. (2020), Bao et al. (2017)	Provides transparent model evaluation with explainable features and ensures adaptability of the hybrid framework to evolving financial datasets.

2.4 Summary

The background research establishes the importance of avoiding late and wrong diagnosis of the mango leaf diseases which is very important in areas where mango production is a major contributing factor to income generation. Although manual inspection is common due to several factors such as familiarity and simplicity, it is slow and requires a great deal of manpower and is very subjective hence less likely to identify early symptoms and more probable to fail when dealing with large cultivations. According to the literature review, deep learning, and more specifically Convolutional Neural Networks (CNNs) have demonstrated potential in the recognition of plant diseases, as they automatically learn discriminative features on raw images themselves. Research on crops like tomatoes, rice, potatoes, have been shown to be highly accurate; however, research on mango leaf disease is still limited with major challenges such as small and unbalanced datasets, variations in weather conditions as well as the inability to interpret the model.

Such analysis highlights that, although the CNN-based methods can form a solid basis, they still need to be developed to address the data-related concerns and to improve model explainability. These gaps could be filled with the integration of preprocessing techniques, the artful selection of network architecture, and explainable AI tools, such as Grad-CAM. In addition to this, real-life applicability of a system, particularly in an agricultural set up, can foster an academic and practical difference as well. In nutshell, the background has provided an evident motivation of the development of an effective, valid, and interpretable CNN-based disease-detection model of mango leaves, and putting the effort to the goal of sustainable agricultural productions as well as better productivity in farming.

Chapter 3

Research Methodology

In this chapter, the research design and methodological approach adopted during the study are described. It explains the method of data collection and data preprocessing (resizing, normalization, and splitting the dataset). It describes the training of several CNN architectures: VGG16, ResNet50, InceptionV3, DenseNet121, and Xception and the strategy of stacked ensembles based on VGG16 as the meta-model. The evaluation measurements used to determine performance are also part of the methodology, and so is the model visualization and export to TFLite format, and deployment. Every step is given in a way that makes the research reproducible.

3.1 Methodology/Requirement Analysis & Design Specification

3.1.1 Overview

Here, we have considered the traditional stacked deep learning strategy to identify and diagnose diseases in mango tree leaves using the convolutional neural networks (CNNs). The process of the dataset preprocessing would include DPI adjusting, normalization, and train/val/test separation (of 70:10:20, respectively) as well as 8 000 images in eight classes (seven disease types and healthy leaves). To increase the integrity of the classifications carried out five base models, including VGG16, ResNet50, InceptionV3, DenseNet121, and Xception are concurring trained and their findings pooled together with the VGG16 as a meta-model. The live demonstration is compared with independent base models and with the results that can be seen. Lastly, an optimized model is exported to the TFLite format in a bid to enable the implementation of practical possibilities to the model and has it be scalable on the agriculture implementations. Such a strategy can be defined as one end of the trade-off in relation to the components of the computational requirements and the sensitivity of diagnosis so as to detect the existence of diseases at an earlier stage of mango production.

3.1.2 Proposed Methodology/ System Design

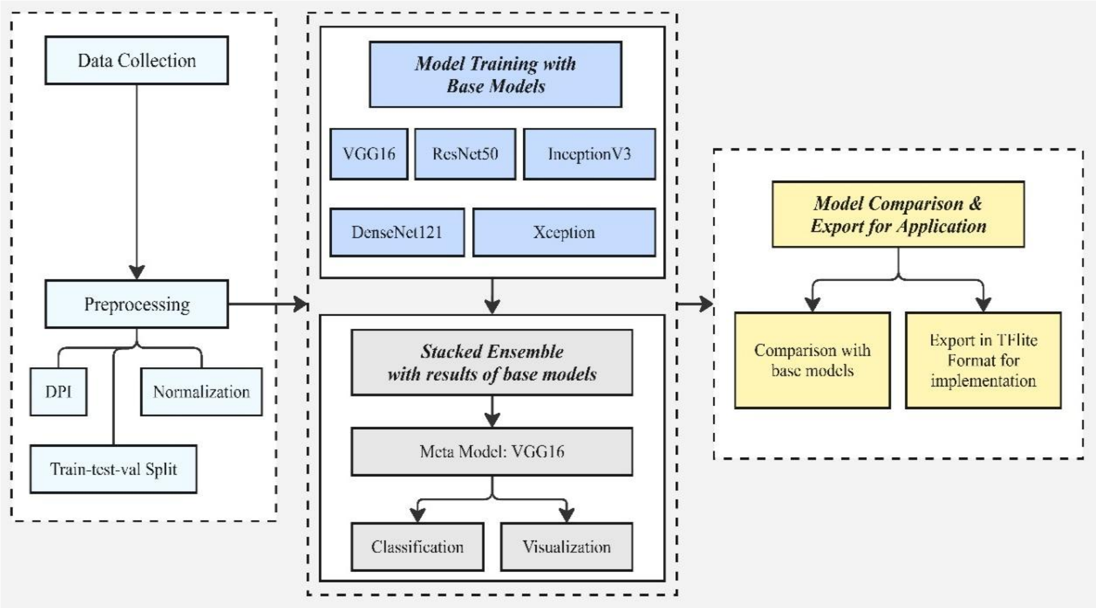


Figure 3.1: The Methodological Flowchart

The Figure 3.1 includes the methodological flow of mango leaf disease with stacked ensemble CNN methodology. The overall work starts with a step of data collection followed by preprocessing phase where DPI-correction, normalization, and training (70 percent), validation (10 percent), and test (20 percent) set division of the dataset is performed. Thereupon, five fundamental CNN models (VGG16, ResNet50, InceptionV3, DenseNet121, and Xception) will be trained individually on the preprocessed information. They are in turn combined and fed into a meta-model (VGG16) in an attempt to create the stacked ensemble to enhance robustness in classification. The model runs the comparisons on the performance by matching the model, and visualizes the result to ascertain the accuracy of the classes of the disease. The optimal model is then to be exported as a TFLite model and may be used in real-life applications in agriculture, which implies real-time and automatic diagnostic of the disease.

3.1.3 Functional and Nonfunctional Requirements

Functional Requirements

- Disease Classification: The system ought to be highly specific in categorizing the images of mango leaves into one of the eight categories, therefore, Anthracnose, Bacterial Canker, the Cutting Weevil, Die Back, Gall Midge, Powdery Mildew, Sooty Mould, or the Healthy.

- Preprocessing Capability: Any analysis operation does not kick in before the model goes ahead to do the tasks involving auto-normalising input images based on their brightness level, determining DPI, as well as resizing the input image to a predetermined resolution point.
- Multi-Model Training: One should be able to both train and evaluate a series of all the five stand-alone CNN networks (VGG16, ResNet50, InceptionV3, DenseNet121 and Xception), and each of these five models stacked together (ensembled).
- Performance Comparison: The framework should give out some metrics of comparison (e.g. accuracy, precision, recall) to compare the ensemble models with those of the base models.
- Export Functionality: The final model is to be transferred to TFLite with the aim to be deployed to either a mobile or an edge device.
- User Interface: In the incident of an application, it should be simplified where a user simply deposits a sample leaf input and get a diagnostic outcome.

Nonfunctional Requirements

- Correctness: The ensemble model has to be precise on the test data at least 90 per cent to be taken as reliable.
- Computational Efficiency: The model should be capable of utilizing GPU acceleration to train within the least amount of time and maintain a low latency without over-bearing inference and be run in real time.
- Scalability It should implement big data (8,000+ images) without compromising much on performance.
- Portability: the model exported by TFLite must run on constrained devices, such as an IoT-based smartphone or IoT-derived agricultural equipment.
- Robustness: This minimizes the necessity to repeat the model construction task of training it on a new set of images when a minor difference in the quality of the images is present (e.g. lighting conditions, camera angle and minor obstructions).

3.1.4 Data Flow Diagram

The top-level data flow of the disease detection system of mango leaves is explained in figure 3.2 where the overall pipeline (data set processing to deployment) is detailed. The first stage in the working procedure will the obtainment of MLD24 Mango Dataset, which will be pre-processed by normalization and resizing steps. This data is trained to six other base models of CNN (VGG16, ResNet50, InceptionV3, DenseNet121, Xception, and another one not specified) trained independently. The results are in turn applied to a stacked ensemble model, with VGG16 as the meta-learner to increase prediction accuracy. The system evaluates models of different architectures with respect to classification metrics and visualisation, finds out the best model of every architecture, and translates it to a TFLite model to be used in an application used in a mobile phone-based application to make the other agricultural applications on the go. The specified flow chart eloquently explains the linearized procedure that will fit data to the implementation of the system, which is linked to its scalability and enhanced performance.

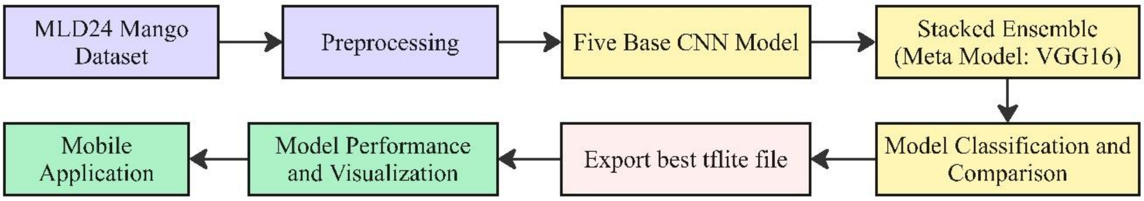


Figure 3.2: Data Flow Diagram Level -1

3.1.5 UI Design

The UI applied in Mango Leaf Disease Detector mobile is neat and preferable, as the principal screen displays the name of the diagnosed ailment and accuracy of the Classification of the disease device (See Figure 3.3). The two simple options presented to the user interacts with the system to take a new picture of a mango leaf using the camera of the gadgets or posting the already existing image on the system. Its user friendly design enables it to be operated by farmers and other persons working in the agricultural sector with ease and it thrives on a rapid diagnosis that can be delivered in easily detectable as well as actionable forecasts. This convenient work could thus be considered a trade-off between low age of CNN based detection and end use functionality where no technical training is required on the end-user side.

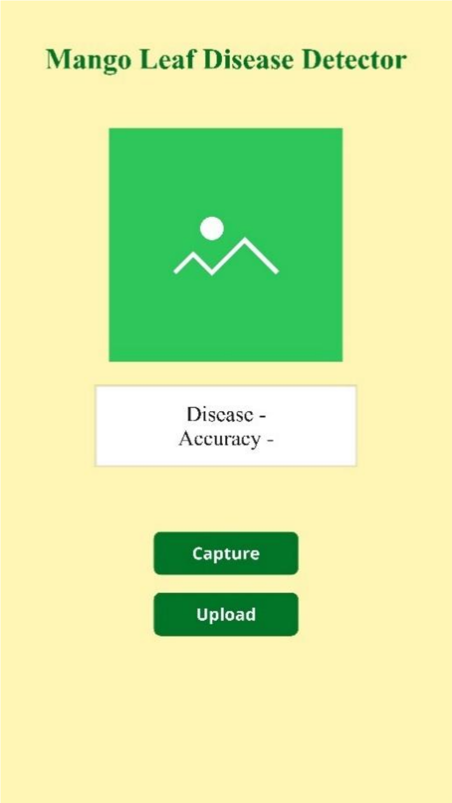


Figure 3.3: Proposed User Interface Design

3.2 Detailed Methodology and Design

3.2.1 Dataset

The study uses Mangos Dataset which is called MLD24 Mango Dataset and comprises 8,000 high-quality pictures of mango leaves, and is labeled with 8 classes (7 classes of disease (Anthracnose, Bacterial Canker, Cutting Weevil, Die Back, Gall Midge, Powdery Mildew, and Sooty Mould) and 1 Health) classes. It is strategically split into training (70 percent, 5600 images), validation (10 percent, 800 images), and test (20 percent, 1600 images) in order to ensure reliability of development and testing of the model. The processing of all the images is similar and it includes normalizing resolution (DPI normalization effect) and normalizing pixel values to allow easy identification and extraction of features of a consistent set of images. It is a heterogeneous and balanced data set, an outstanding point to train the deep learning models so that one can identify the roots and label the diseases in the mango leaves with great accuracy, and the volume of biases will be lesser.

Table 3.1: Dataset Specifications

Properties	Values
Image Resolution	240 × 240 pixels
Format	.jpg
Total Images	6400
Classes	8

Classes

- **Anthracnose:** This Wilting down: This fungal disease is manifested in the presence of black abjected spots on the leaves as well as fruits and this is as a result of the Colletotrichum gloeosporioides specifically. It significantly influences the quality and fruitivity of fruit particularly there being a humid situation.
- Bacterial Canker: The identification of the disease is made by the presence of angular leaf spots, exudates of gummy material as well as twig dieback; the causal agent is the Xanthomonas campestris. It makes plants to shed prematurely and trees become weak.
- Cutting Weevil: Weevil (Deporaus marginatus) eats the leaves and puts characteristic notches on its edges. It is mainly believed to be referred to as pest destruction only, however, heavy infestations can also cause dewd grounded trees and also reduce photosynthesis.
- Die Back: Extreme chaos whereby tips of the branches start to die and elongate towards the center and it would normally be usually related with the fungal pathogen like Botryodiplodia theobromae. It causes a massive canopy loss and the decrease in yield.
- Gall Midge: It could be recognized as a distorted and swollen leaf with larvae of Procontarinia mangiferae. This deformation due to this pest also affects the productivity of the trees and stunts the new growth of the trees.
- Powdery Mildew: They are powdery white, fungal lesions on top of the leaves and are caused by Oidium mangiferae. It favours dry weather conditions and even it is possible to suppress whole-plant photosynthesis and fruit set.
- Sooty Mould: Is a black powder like fungus growing on honeydew, a liquid secreted by Aphid insects. It does not cause the disease itself but it blocks the sunlight and renders the plant weak.
- Healthy: Leaves bearing no obvious evidence of a disease, uniform green green leaf colour and normal morphology are used as a comparison control.

Anthracnose Bacterial Canker Cutting Weevil Die Back

Gall Midge Healthy Powdery Mildew Sooty Mould

Figure 3.4: Sample Image from Each Class

3.2.2 Preprocessing Techniques

The multiple important stages of dataset preprocessing are done to improve the workings of the models and offer uniformity of data input. Initial, resizing of the images reduces all the images to a standard resolution (typically 224 224 or 256 256 of pixels to match the CNN input requirements).

The second involves normalization of DPI to enable all the samples to have the same denominator of pixel density. Subsequently, their images are normalized in color distributions, i.e. they divide each pixel with 255 to form the range 0-1, then they subtract the means and divided by the standard deviations thus the meaning becomes 0, and the distribution resembles the normal distribution and thus they can converge during training more easily.

The last step would involve stratified sampling of the dataset into three components; training set (70 percent), validations (10 percent), and test set (20 percent) with equal representation of the classes at the end of the stratified sampling procedure (See Figure 3.5). The entire process of preprocessing can be helpful in the feature learning process but not in a negative sense of preserving the biological relevance of the leaf images.

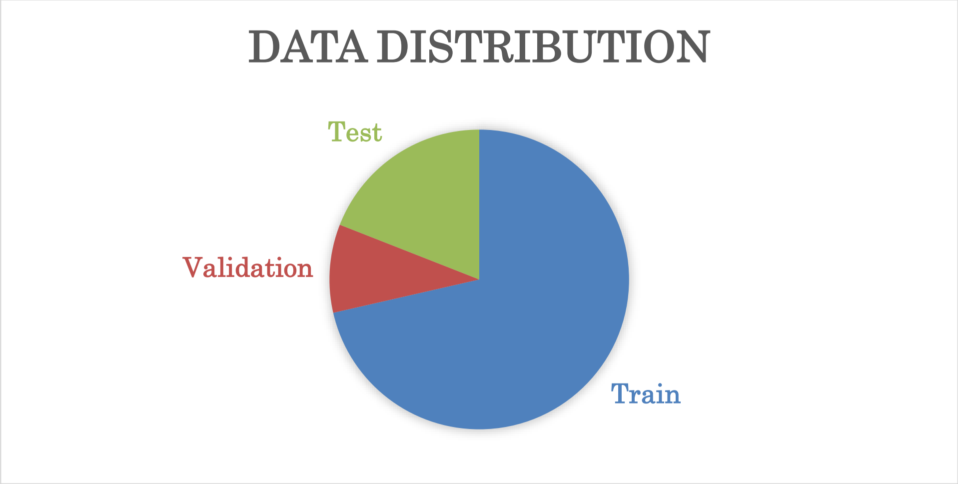


Figure 3.5: Data Distribution for Train, Validation and Test.

3.2.3 Model Description

1. Working Procedure of Transfer Learning Technique

The paper applies the five base models that are pre-trained CNN networks (VGG16, ResNet50, InceptionV3, DenseNet121, and Xception) with the aim of extracting features and performing preliminary classification. Both are hierarchic feature-learners:

- Input Preparation: The preprocessed images of the mango leaves of all the base CNN models (VGG16, ResNet50, InceptionV3, DenseNet121, Xception) have been carried out in the following way: all the original preprocessed mango leaf images would be rendered to the same image size (e.g. 224x 224 pixels). The values of the pixels are scaled to normal size in order to give a consistent feed to the neural networks.
- Feature Extraction: The models would begin with a convolution series of layers as the input image is fed in them. These layers execute learned filters that detect patterns in a hierarchical fashion, whereby easy edges and textures are identified earlier in the layers, and more complex disease-related findings, like lesions or discoloration is in later layers.

$$F_{i,j} = ReLU(\sum_{m,n} K_{m,n} \cdot I_{i+m,j+n} + b\ ) \qquad \dots\dots\dots(i)$$

- **Spatial Reduction (Pooling):** Being two convolutional layers, pooling layers reduce the feature maps following each computation, removing the least relevant information, but simplifying the calculations, rather than increasing their complexity. Max pooling is the most famous one to use because, it selects the highest value of small regions in the feature map to retain the dominant characteristics.

$$P_{i,j} = max(F_{2i2i+1,2j:2j+1}) \qquad \dots\dots\dots(ii)$$

- **Hierarchical Learning:** Hierarchical Learning: The images being passed through the deeper layers of the network are learnt in more abstractive representations. This is further optimized by such architectures as ResNet50, DenseNet121, where grading to flow is more readily achieved by skip connections, and feature reuse is further increased.
- **Classification Head:** The final segments of the model transform the extracted information into the 1-dimensional vector and then it is utilized by fully estimated (dense) layers. Such layers know the higher-level features and assign weights to different signs of the disease.
- **Probability Estimation (SoftMax):** If all disease classes are to be covered by the probability scores, a SoftMax activation is placed on the final layer to transform the raw output of the model to the probability scores. The most likely classification will be considered as the diagnosis predictions.predictions.

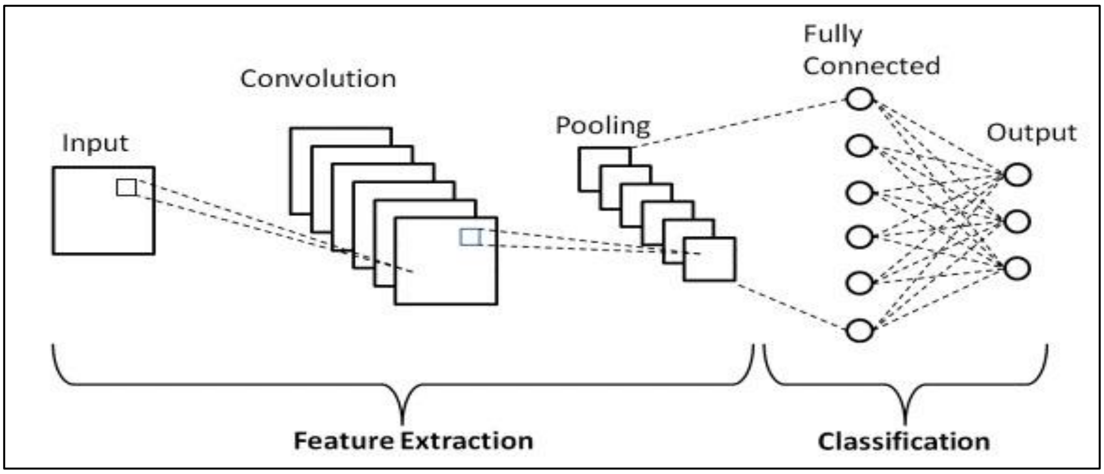


Figure 3.6: The base architecture of the base CNN Model

3.2.4 Stacked Ensemble

Stacked Ensemble means to enhance how well the classification works, a two-tier architecture is employed in the stacked ensemble model to combine the strong points of several base CNN layers to construct a unified system that serves an end. In the initial phase, separate images are introduced into five base models (VGG16, ResNet50, InceptionV3, DenseNet121, and Xception) previously trained and will give an 8 dimension probability vector of each prediction. These vectors are concatenated into a meta-feature matrix that is consequently passed on to the meta-model of the second level (VGG16). In supervised learning, meta-model learns the best weightings of the base models outputs, predictions and hence gets the higher order relationships amongst their outputs. This hierarchical combination may look like it avoids the biases of any single model, utilizes complementary feature representations and is more generalizable than any single model. This part of the ensemble is obtained as a weighted average of the base predictions and trained by backpropagation to minimize the loss of categorical cross-entropy on the validation data. It is an illustration of an incorporative procedural systematization of a broad scope of architectural inductive prejudgments that undertake sound diagnosis of the mango leaf disease.

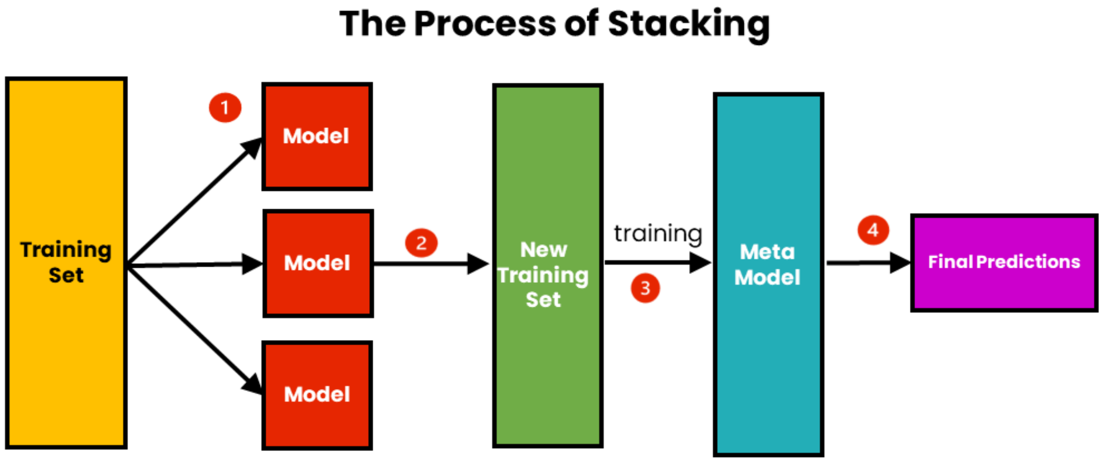


Figure 3.7: The architecture of the Base Stacked Ensemble Method

3.3 Project Plan

The specific order of the project is outlined in Table 3.2 and includes theoretical research and literature review (Feb - Mar 2025) as the initial part to ensure the obtained preliminary knowledge and help outline any literature research gaps and areas that require the additional implementation of the theory. Model training Scheme: The mango leaf training dataset (Mar 2025) will be guided by a procedure of training models on the basis of downloading the mango leaves dataset (Mar 2025) and preprocessing (Apr 2025) to make the images uniform among them facilitating in the training of the model to give better results. The model design and development of the methodology (May 2025) phase entails the effort of model construction and optimization (building a CNN ensemble) in parallel with the development of the methodology chapter. This will comprise of mobile app development and testing (June-July 2025): the trained model is made available on an app and report writing (Aug 2025): the final collection of findings into the thesis document. The phases ensure that the project is structured such that it goes in phases throughout the research up to implementation by each phase basing on the preceding deliverables to present results that are efficient and effective.

Table 3.2: The GANTT Chart of Project Timeline

Process	Weeks											
	06-07	08-09	10-11	12-13	14-15	16-17	18-19	20-21	22-23	24-25	26-27	28-29
Working Plan	■	■										
Theoretical Study	■	■	■									
Literature Review			■	■								
Data Collection				■	■	■						
Data Preprocessing							■					
Model Design + Methodology Writing								■	■			
App Development and testing										■		
Report Writing											■	■

3.4 Task Allocation

The systematic splitting of project tasks was introduced in terms of the expertise of the project participants since the main investigator was responsible in developing the theoretical framework and designing the methodology whereas the co-investigators were to collect the data, preprocess it and apply the model. It has been relying on the knowledge of a committed machine learning technician who majors in the enhancement area of CNN architecture structure and on the other side; a software developer has been leading the area of mobile application embedment and user interface construction. The literature survey, performance reviews and documentation were undertaken by research assistants and consequently did not leave any hitch in covering all stages of the project. Regular team syncs have maintained things on track and the progression of the tasks have been plotted in the GANTT chart (Table 3.2) based on accountability and proper utilisation of resources throughout the project.

3.5 Summary

The mango leaf disease detection process was put forward in this chapter (systematically) and it began with a stacked ensemble process wherein a base/original model was comprised and constituted of a base model (five CNN models namely VGG16, ResNet50, InceptionV3, DenseNet121, and Xception) massed up by a meta-model (VGG16). All the steps of MLD24 (8K images and eight classes) were normalized, augmentation and stratified split (70-10-20) in order to achieve effective training. The workflow also included model training, comparison between model performance and TFLiteing the trained model to be applied in a mobile setting, and the design of the user interface was conceived as requiring an amiable user interface. The creation of the system was formed with the strict functional and non-functional requirement, and the pending project, as well as the assignments of the tasks, were appropriately structured. Taken together, this strategy attempted to strike a balance between technical formulation and achievable expansion, with an aim of delivering an effective process that can be tested in an implementation context to the diagnosis of agricultural diseases.

Chapter 4

Proposed Model for Mango Leaf Disease Detection

4.1 Introduction

This chapter introduces the proposed deep learning–based model for detecting mango leaf diseases. While Chapter 3 presented the methodology and individual components of the research, the focus here is on their integration into a complete framework. The design combines several state-of-the-art convolutional neural networks (CNNs) through a stacked ensemble approach, ensuring robust and balanced performance across all disease categories.

Traditional single model approaches often excel in one aspect of feature extraction but struggle with others. For example, shallow architecture can capture fine-grained features but lacks robustness, while deeper models may overfit or misclassify visually similar diseases. The proposed stacked ensemble model mitigates these challenges by integrating the complementary strengths of multiple CNNs, supported by a meta-classifier that learns optimal decision boundaries.

The sections that follow describe the overall system architecture, the role of each base model, the ensemble integration process, and the novelty of the proposed approach compared to previous works.

4.2 System Architecture

The mango leaf disease detection (figure 4.1) process of the given research has been approached as a multi-class image classification task, wherein every input image is classified as belonging to one of the eight classes (seven disease categories, and one healthy one). In order to develop a powerful detection system, the set of images (MLD24) with approximately 8,000 images was gathered and systematically preprocessed. Resizing of images and normalization along with augmentation of images were done to enhance generalization and tackle variability in real world scenarios. The deep learning models take these prepared images and by default produce hierarchical features and classify the type of disease without manually engineering features.

The system should replicate and improve the diagnosis ability of human specialists by turning the process of identifying the disease into an image classification task that can be completed with the help of convolutional neural networks (CNNs) and ensemble techniques. The result is a practical, field deployable instrument that allows the real time detection of mango leaf diseases with the help of mobile devices filling the gap between laboratory research and practical field implementation of agriculture.

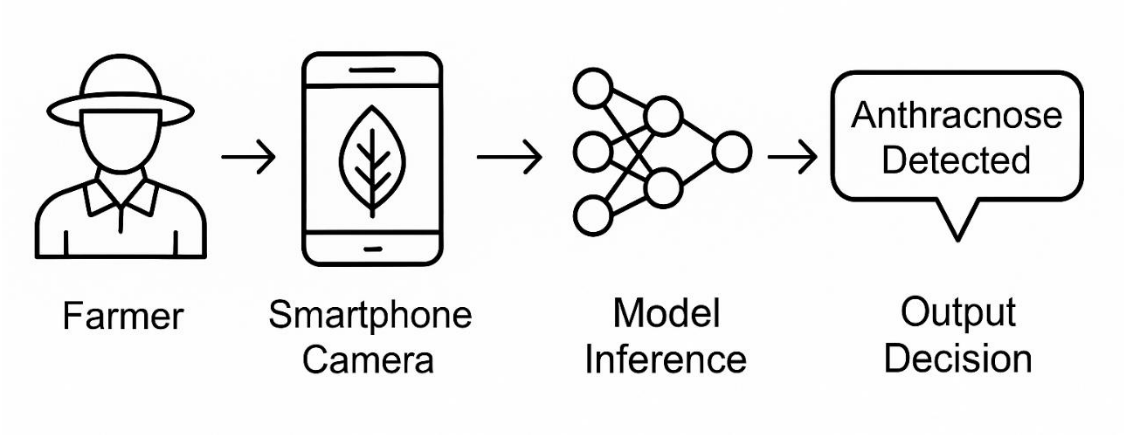


Figure 4.1: MangoDoc - Mango Disease Detection System Architecture

4.3 Ensemble Learning Strategy

This study has used a stacked stack of CNN models to improve the accuracy and strength of detection. The VGG16, ResNet50, InceptionV3, DenseNet121, and Xception are the base learners, each of which has its strengths and weaknesses in feature extraction that are complementary to each other mentioned in table 4.1. VGG16 offers structured learning in layers by layers, ResNet50 alleviates vanishing gradients with skip connections, InceptionV3 upholds multi-scale features, DenseNet121 reinforces feature reuse and gradient flow, and Xception uses depthwise separable convolution to be efficient. Every base model generates a probability distribution of the disease classes which are used as the input to the meta-classifier.

Table 4.1: Base CNN Models and Strengths

Model	Unique Strength
VGG16	Structured layer-wise learning; strong for hierarchical features

ResNet50	Skip connections mitigate vanishing gradient; robust deep architecture
InceptionV3	Captures multi-scale features with inception modules
DenseNet121	Encourages feature reuse; strengthens gradient flow
Xception	Efficient with depthwise separable convolutions

As per Figure 4.2, these probability outputs are used in combination to come up with the final decision as the meta-classifier, which is created using VGG16 in this work. Meta-classifier mitigates bias of single architecture by training to consider the predictions of all the base models and increases the overall ability of the system to generalize. Categorical cross-entropy loss is used to train the ensemble, data augmentation, regularization, and early stopping are used to avert overfitting. Additionally, the conversion to use TensorFlow Lite makes it possible to use the trained deep learning system in mobile devices without a major performance compromise.

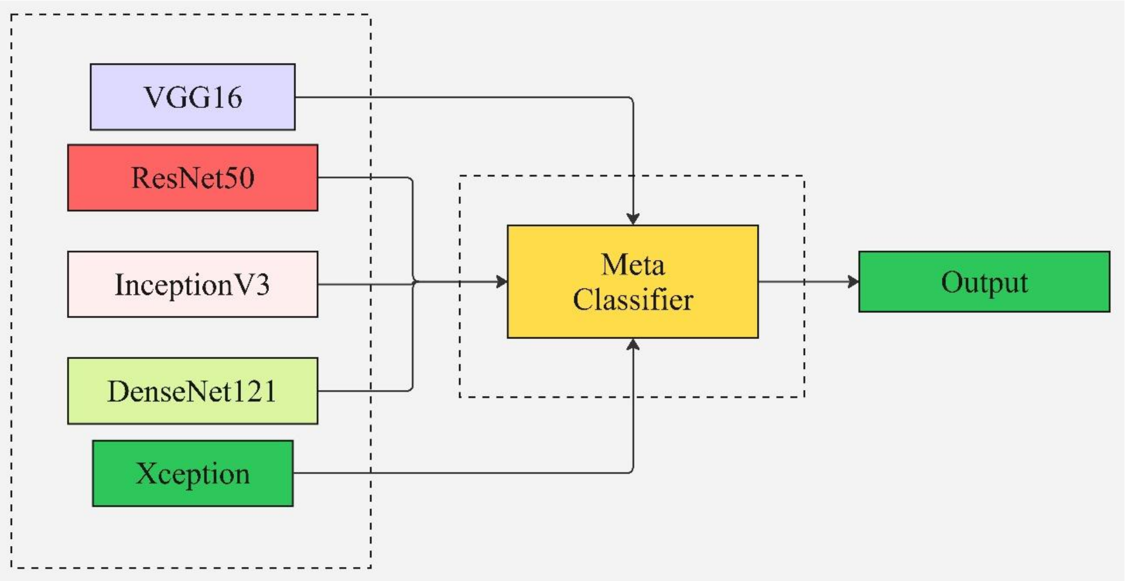
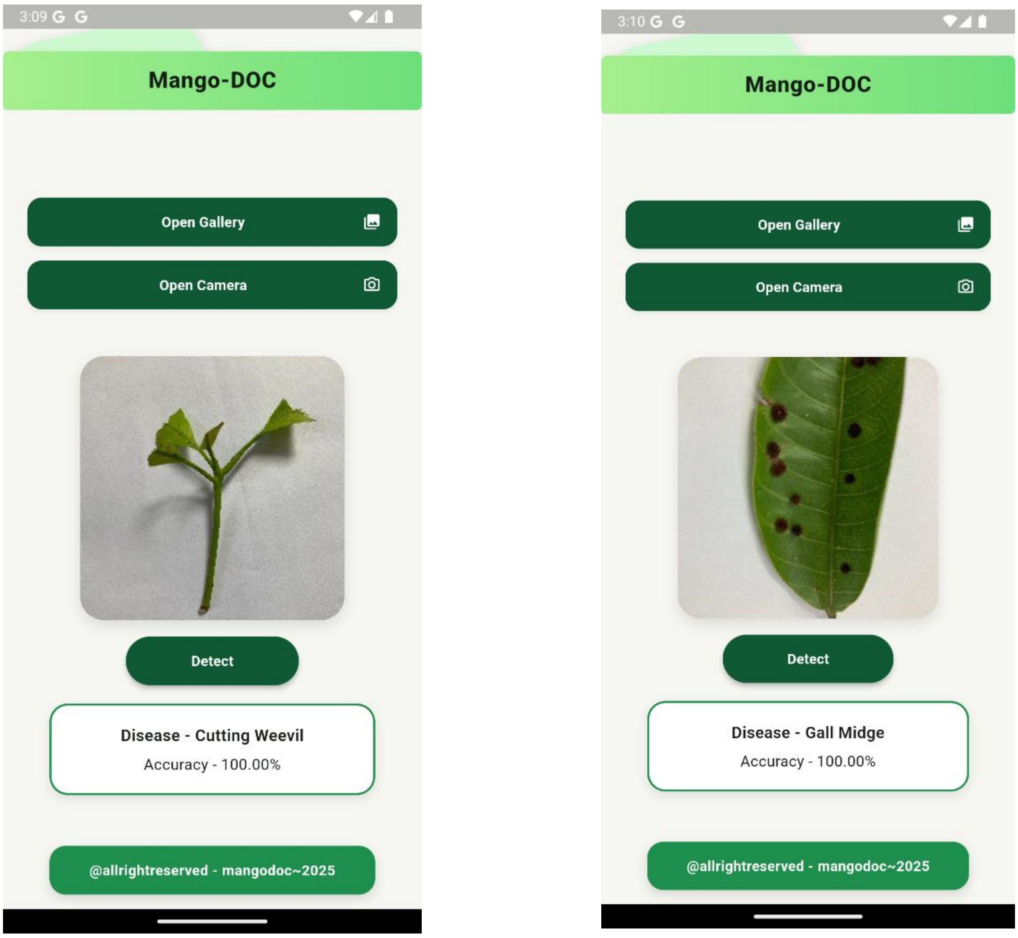


Figure 4.2: Base Learners and Meta-Classifier Integration

4.4 Novelty of the Proposed Model

The suggested mango leaf disease detection model has a number of new features that make it different compared to the current models. To begin with, it does not utilize a single deep learning network: it uses a stacked ensemble approach that combines the results of several CNNs joining VGG16, ResNet50, InceptionV3, DenseNet121, and Xception. This design exploits the relative virtues of various architectures, which makes the categorization of all disease categories much more balanced and stronger than when using single models.

Second, the decision boundaries are optimized in a new manner by using a meta-classifier that is trained using concatenated probability vectors. In contrast to more traditional ensemble methods like majority voting, this method trains how to paint predictions, which minimizes the effect of models with less strength and improves overall performance.



Disease - Cutting Weevil
Accuracy - 100.00%

Disease - Gall Midge
Accuracy - 100.00%

Original: Cutting Weevil
Detected: Cutting Weevil

Original: Gall Midge
Detected: Gall Midge

Figure 4.3: Real-time Outputs on Smartphone

Third, as per figure 4.3, the model is prepared with great passion towards actual applications. The system can be trained on smartphones and made to work with the help of transferring the trained ensemble to TensorFlow Lite, thus providing real-time disease detection in the field. This combination of cutting-edge deep-learning with mobile functionality provides a major gap to the current studies, with a significant number of high-performance models being restricted to laboratory applications due to computational constraints.

Lastly, the use of interpretability with Grad-CAM visualizations provides an extra new quality. This not only confirms the reliability of predictions by marking out regions of the disease in the leaves but also earns the user confidence by making the system a little more transparent. All these contributions put forward make sure that the proposed model is not only academically important, but also practically feasible in changing the way mangos are grown in beleaguered resource-constrained settings.

4.5 Summary

The proposed mango leaf disease detection model is a complete deep learning structure and has been introduced in this chapter which combines various CNN architectures by a stacked manner of ensemble architecture in order to attain strong and balanced performance on all type of diseases. Starting with the system architecture, a multi-class classification problem was presented as a detection process, in which an 8-class dataset was used, and the dataset underwent careful preprocessing to enhance generalization when used in the real world. The ensemble learning approach was an integration of the complementary performance of VGG16, ResNet50, InceptionV3, DenseNet121, and Xception with a VGG16-based meta-classifier was used to optimize decision boundaries to decrease bias and increase generalization, and TensorFlow Lite conversion made them suitable to run on a mobile device. Last but not least, the model novelty was discussed, such as its stacked ensemble format, probability-based meta-classifier, real-time mobile-related functionality and its explainability through Grad-CAM visualizations that make the system not only a notable academic input but also a helpful tool to apply to fields on an agricultural application level.

Chapter 5

Implementation and Results

A careful consideration of five deep convolutional neural network (CNN) architectures VGG16, InceptionV3, ResNet50, DenseNet121, and Xception against a proposed stacked ensemble model to a multi-class classification problem is carried out, in this chapter. Its assessment is performed on the basis of the fundamental performance scores, i.e., training and validation loss and accuracy curves, confusion matrices, classification reports (precision, recall, F1-score), and ROC-AUC curves. The investigation aims at the mechanisms of learning of each model, generalization, and distinction of each of the classes in the dataset. In particular, determining strengths and weaknesses are considered by focusing on stability, accuracy, and robustness. Stacked ensemble model is proposed as a method of combining predictive ability of several base models in an attempt to improve the accuracy of the entire classification process. The chapter is about comparative analysis and through which, it is found that ensemble learning is effective and also good enough to mitigate individual weaknesses as well as make the whole system more consistent, correct and class-balanced in the predictions.

5.1 Environment Setup

Table 5.1 provides a summary of the typical training parameters that were applied to all the models which have been experimented with and this establishes comparability and consistency of different architectures. All images were scaled to 224 224 pixels, a typical input dimension to any CNN based models, because it provides high enough spatial resolution to meanwhile maintain computational efficiency. Almost maximum batch size was selected, 128 so that gradient updates were not volatile and GPU memory was well utilized in training. They ran the base models with a training of 50 epochs and the meta model 30 epochs which gave them enough iterations to converge without any form of overfitting. Adam optimizer was used because it is adaptive and has fast convergence characteristics, thus being ideal when it comes to deep learning. Learning rate was also kept at 0.0001 which is a small number that will enable the model to learn slowly without extreme weight updates, and enable a finer convergence in the long run.

Table 5.1 Common parameter table for all experimented models.

Parameter Name	Parameter Value
Image Size	224 × 224
Batch Size	128
Epoch (for base model)	50
Epoch (for meta model)	30
Optimizer	Adam
Learning Rate	0.0001

The data splitting strategy used in training, validation and testing of the model is given in Table 4.2. The training (4480 images), validation (640 images), and testing (1280 images) data were constituted by 67.5%, 12.5%, and 20 percent, respectively. The division of the data guarantees that a large part of the data will be used to train the model and a large cut of the data is placed to use as an unbiased assessment. Validation set assists in adjusting parameters of the models and avoiding overfitting by checking the performance during training. The test set will not be used in training or validation and will act as a last checkmark when determining the generalization ability of the model. The fact that about 20 percent of the samples in the test data is high, enhances the reliability of the performance measures such that the outcomes will correspond to the accurate performance on the real-world, unseen data.

Table 5.2: Common data split for all experimented models.

Dataset	In Percentage	Number of Images
Train set	67.5%	4480
Validation Set	12.5%	640
Test Set	20%	1280

5.2 Testing and Evaluation/Performance/ Comparative Analysis

In assessing how machine learning models perform towards the study, suitable performance measures should be adopted to give a clue on whether the machine learning model is more accurate, reliable and can generalize. popular measures employed in classification exercises, in particular those connected with agricultural disease identifications are listed below:

5.2.1 Precision

The ratio of the number of instances that are not correctly classified (negative and positive) and number of total instances. The accuracy provides a short flash view of the performance of the model but may apply deceptively in unbalanced data sets in which one of the classes is notably more predominant than the other is.

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

5.2.2 Recall

The percentage number of positive observation predictions that are accurate compared with all actual positives. This is especially true in the case of situations where the positive (a diseased plant) may have drastic implications (the crops are lost, etc.).

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

5.2.3 Precision

The number of positive observations correctly predicted divided by the total number of predicted positive observations. In applications where false positives are very costly the precision is essential. In this research, high levels of precision imply that when a disease is forecasted, it is most probably correct.

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

5.2.4 F1-Score

The harmonic definition of precision and recall that is an alternative that balances the two comparisons. The F1 score also comes in handy when working with imbalanced classes since it takes into account both the false positives and the false negatives and there is a comprehensive way of looking at the performance of the model.

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

5.2.5 Confusion Matrix

A table that is used to characterize the performance of a classification model with the true/predicted classifications. The confusion matrix supplies information on the nature of mistakes the model made, which can be subsequently addressed in more target-oriented ways.

5.2.6 ROC (receiver operating characteristic) Curve

A graph of the performance of a classifier at different threshold levels, its true positive rate (recall) on the vertical (y) axis against its false positive rate. It shows how such an abstract binary classifier system varies in its ability to make a diagnosis depending on its discrimination threshold. It is used to plot the True Positive Rate (TPR), recall or sensitivity versus the False Positive Rate (FPR) with differing threshold settings.

Area under Curve (AUC): Using Area under the ROC curve yields only one measure to determine model performance; the nearer to 1 the better the model.

Receiving Operating Characteristic (ROC) curve and Area Under the Curve (AUC) are important measures according to this study in comparing the models of classification. They enable graphical comparisons of how models perform at different thresholds, which enables one to choose an optimum threshold balancing between specificity and sensitivity. Also, ROC curve does not suffer as much by the imbalances in classes making it a sure platform when accuracy can be misleading. The ROC and AUC however should not be used exclusively in place of other measures such as precision and recall to be sure of overall model effectiveness.

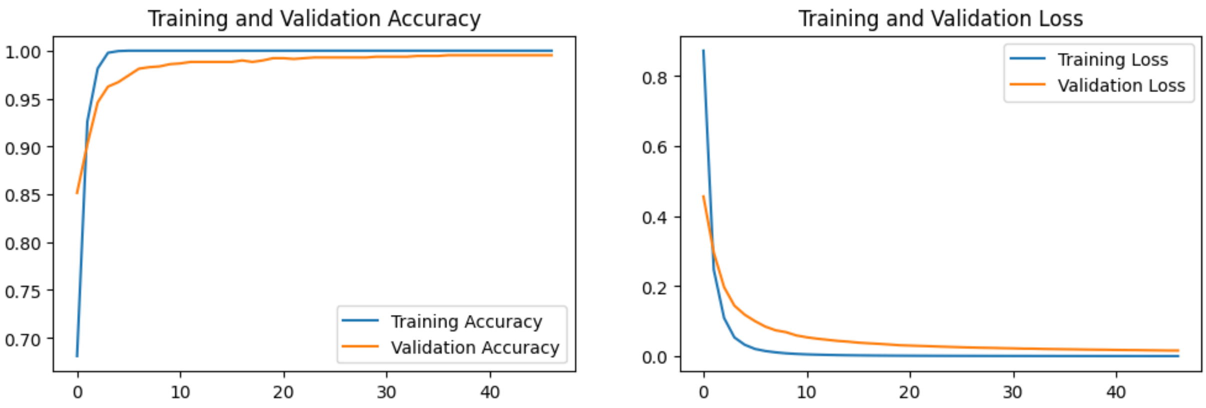
5.3 Results and Discussion

In this paper the performance of the four individual deep CNN models namely VGG16, InceptionV3, ResNet50 and DenseNet121, Xception as base model and VGG16 Meta model that forms stacked ensemble model that fuses the prediction of said CNN models into meta model to form a final decision is evaluated. The effectiveness of each model is evaluated using indicators including loss and accuracy curves, confusion matrices, classification reports, ROC curves, etc. to identify the model weaknesses and strengths as well as additional advantages of the ensemble method.

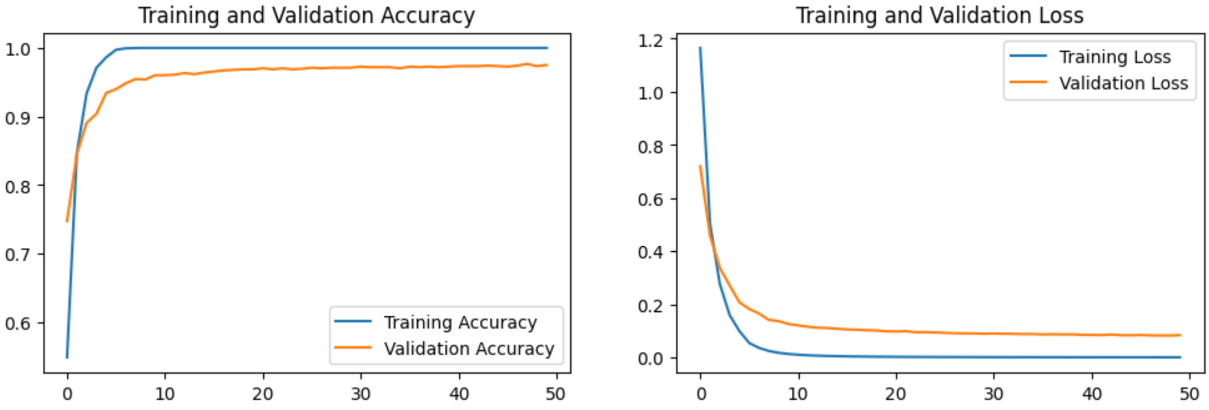
5.3.1 Performance of the base models

This figure 4.1 compares training and validation loss and accuracy curves of five base CNN models which are VGG16, InceptionV3, ResNet50, DenseNet121, and Xception. Of them, VGG16 and DenseNet121 exhibit the most consistent converging where the validation loss is steadily reducing and monitoring training loss favorably predicts little overfitting. InceptionV3 displays slight variations in the beginning stages but the curve is steady towards the end. The curve used with Xception is relatively smooth yet it has a lower rate of convergence. The ResNet50 however has considerable variations in the loss and accuracy, with the loss and accuracy validation curves being particularly sensitive in relation to the training parts of the curve and the divergence is shown early, implying a possible undertraining or circumstance of optimisation problems. On the whole, the figure shows the effects of model architecture on the learning stability, the convergence speed, and the generalization. The most consistent in terms of training the behavior are VGG16 and DenseNet121 whereas ResNet50 has the lowest reliability. This causes the loss and accuracy curves to serve as a fairly early but significant result in determining whether a model will work in practice.

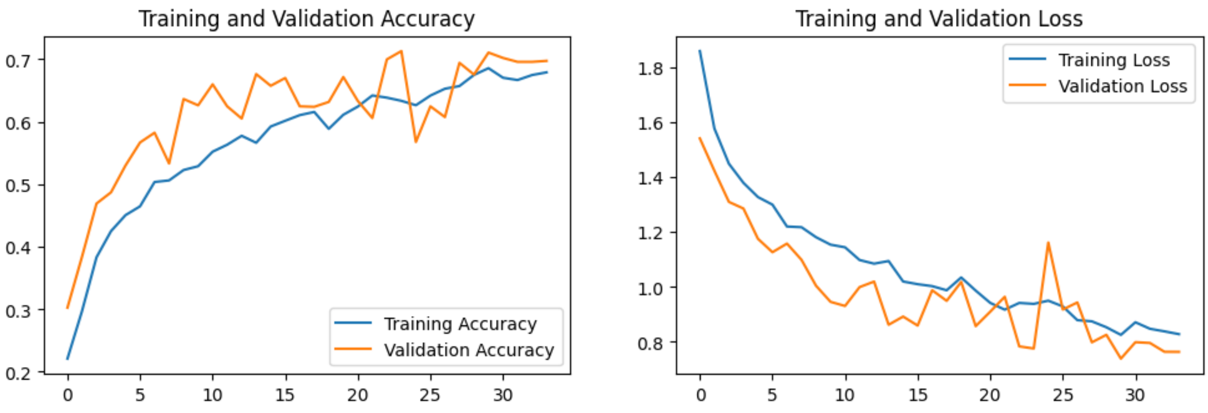
VGG16



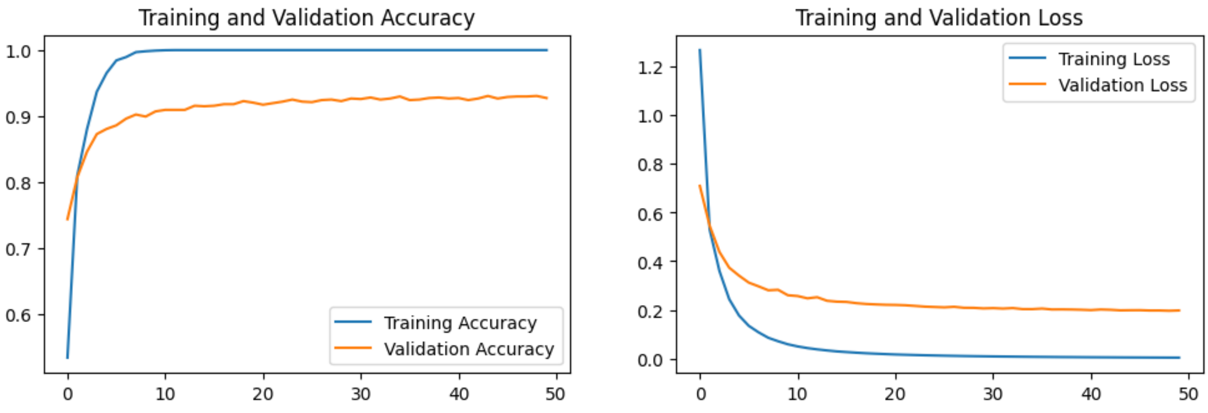
InceptionV3



ResNet50



DenseNet121



Xception

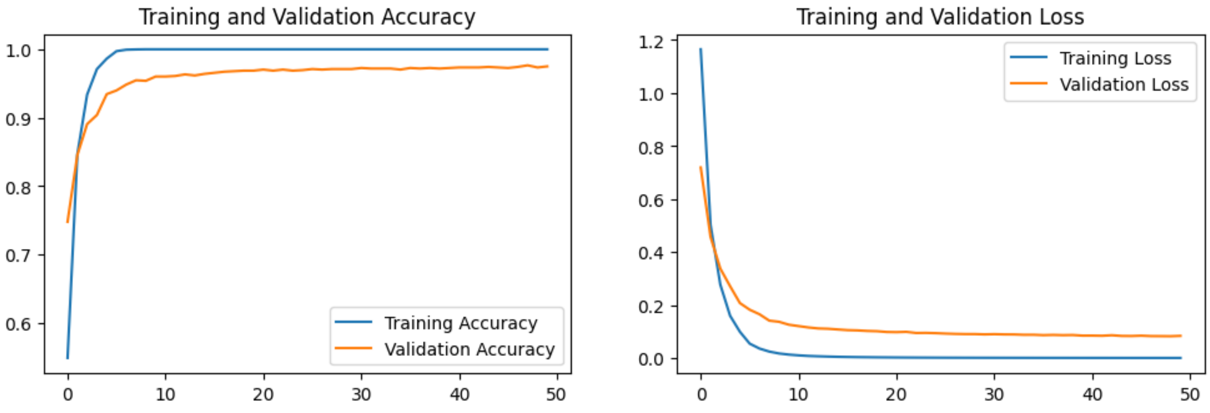
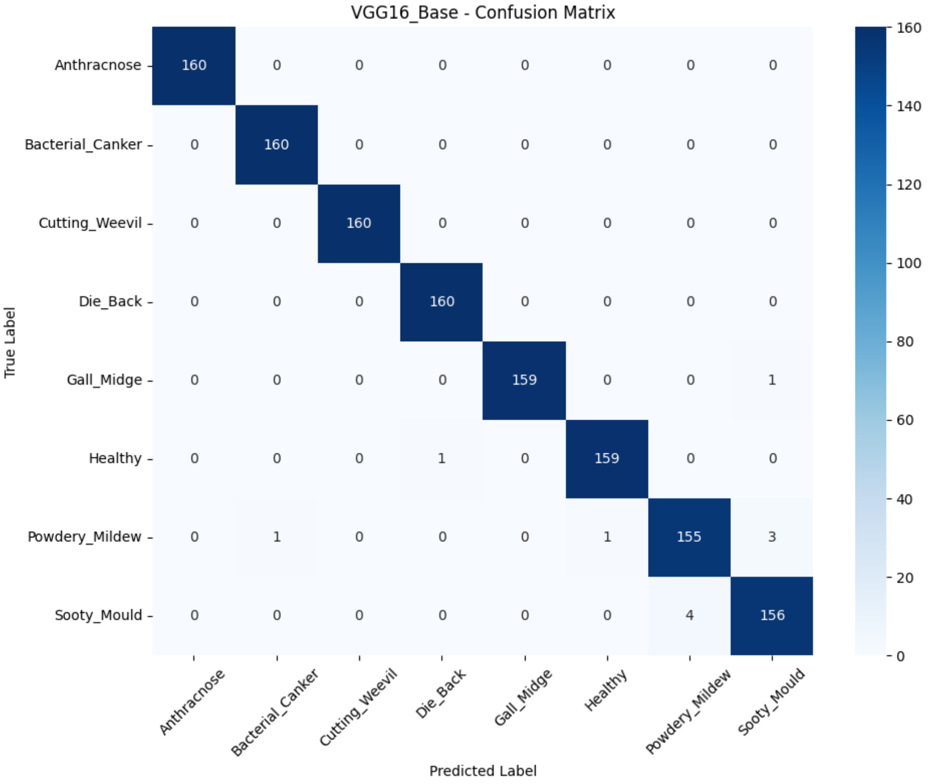


Figure 5.1: The loss and accuracy curve on training and validation set over 50 epochs for five base models.

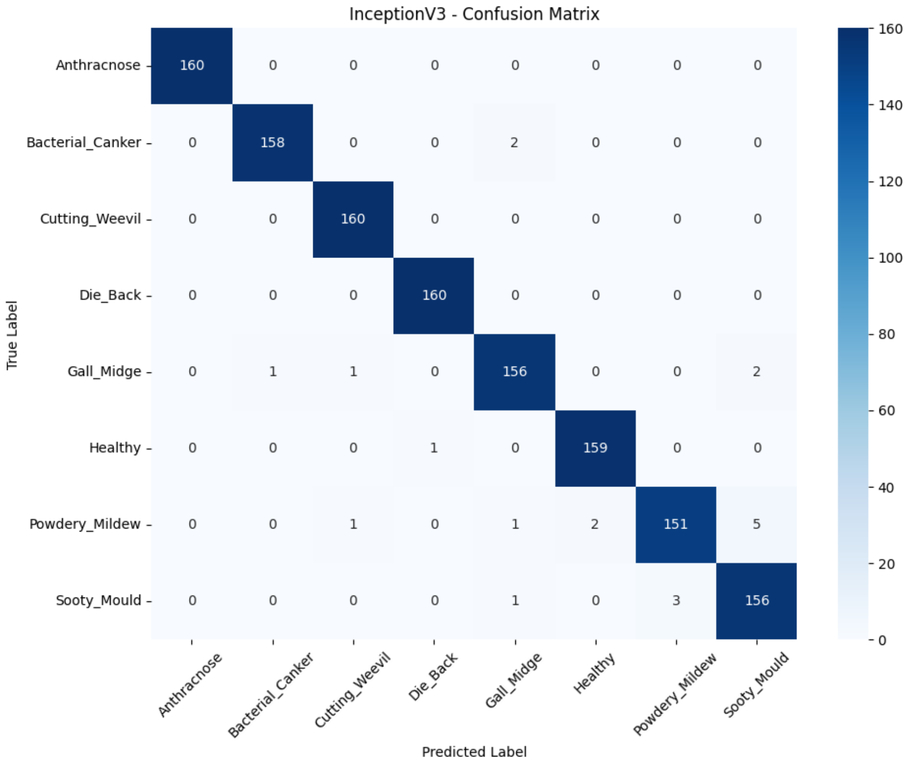
This figure 5.2 shows the confusion matrices with the five base models, which demonstrate the prediction efficacy of all the classes of the test set. The diagonal structures of VGG16 and DenseNet121 are almost perfectly straight, which implies an accurate and repeatable prediction with limited confusion between classes. InceptionV3 and Xception are also good, with slightly off-diagonal elements indicating that there is some minor misclassification, in visually similar classes.

However, ResNet50 shows a very scattered matrix, with an incredible misclassification rate at any category, meaning bad generalization and reprehensible distinction in the classes. Such matrices are especially handy to reveal model behavior beyond aggregate measures such as the confused classes, because they identify the particular classes that people find most confusing. Additionally, DenseNet121 and VGG16 models most strongly demonstrate their resistance to the boundaries of the classes and much weaker ResNet50. The figure confirms the previous conclusions about the accuracy curves, that not only the overall accuracy but also the clarity at the class level must be taken into consideration.

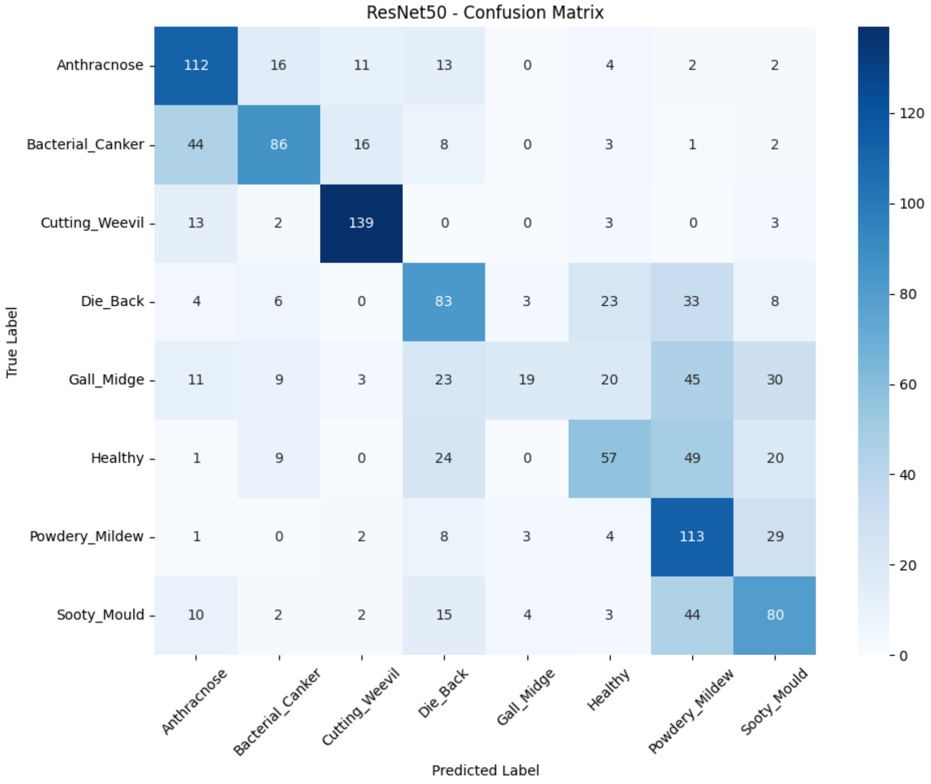
VGG16



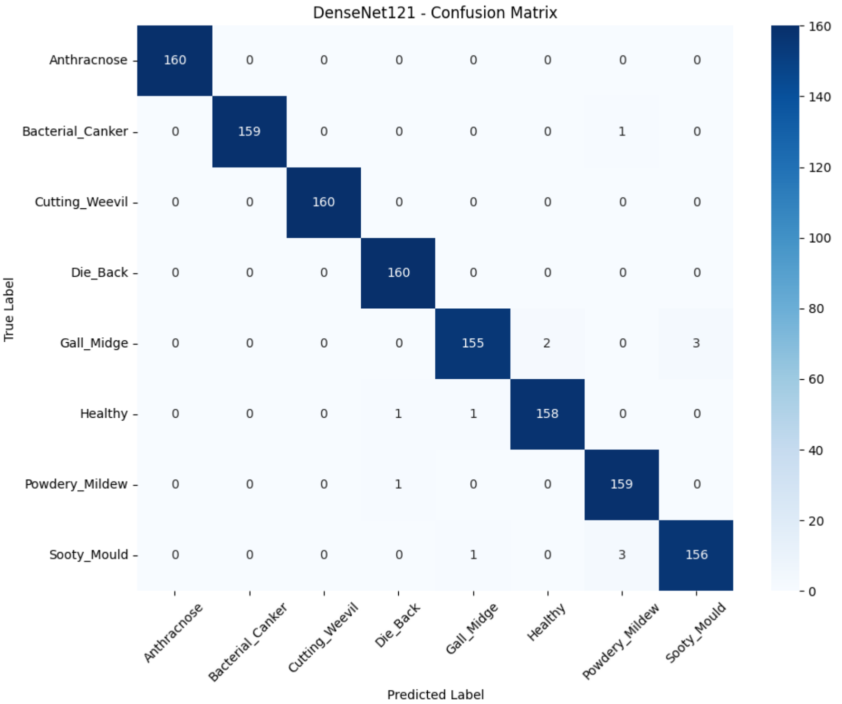
InceptionV3



ResNet50



DenseNet121



Xception

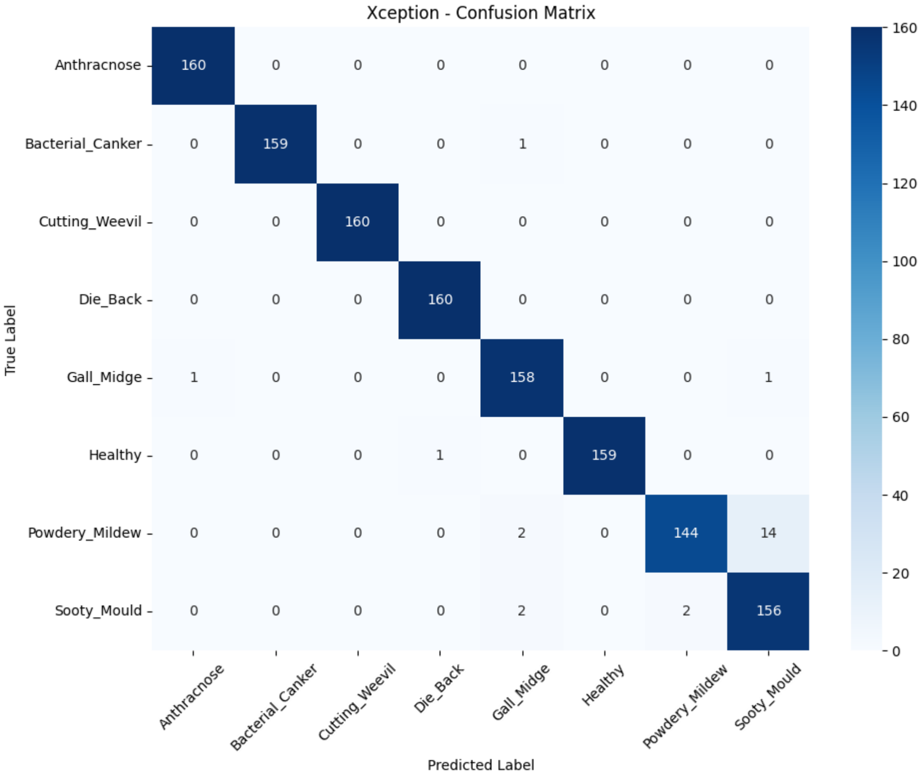


Figure 5.2: Confusion matrix on the test for five base models.

This Table 4.3 has the per-class precision, recall, and F1-score in each of the five original CNN models. VGG16 and DenseNet121 are an exception, producing more than 98% in each of three metrics and barely varying according to classes. This points to stable and moderate classification test. InceptionV3, and Xception have a little lower score (~97-98%), but still, acceptable in terms of high-stakes use. ResNet50 on the other hand demonstrates a very significant decrease in all defined metrics, with an average of around 55-58%, displaying deplorable management of classes, biased predictions and probable lack of training. The present report supplements the results of the ROC and confusion matrix of the model expressing its performance in the metrics of the precision-based reliability, sensitivity, and predictive balance. It confirms the fact that DenseNet121 and VGG16 could be considered the most consistent performers, and ResNet50 should not be used without additional tuning and retraining.

Table 5.3: Classification report on test set of five base models .

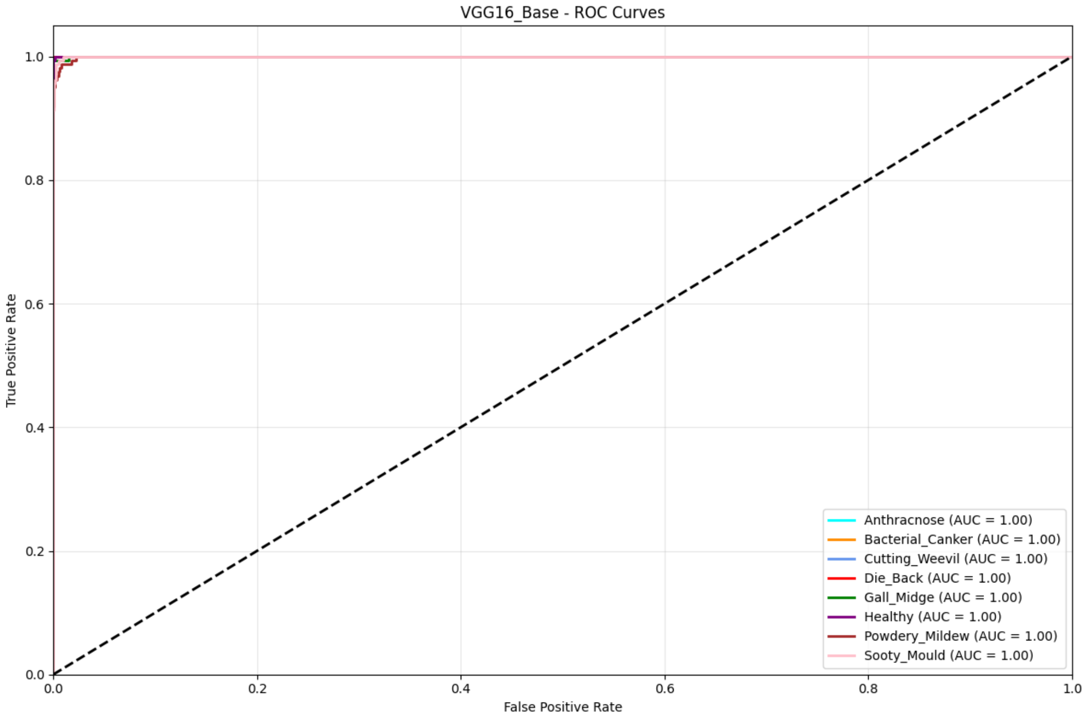
Classes	Precision	Recall	F1-score	Support
VGG16				
Anthracnose	1.00	1.00	1.00	160
Bacterial_Canker	0.99	1.00	0.99	160
Cutting_Weevil	1.00	1.00	1.00	160
Die_Back	0.99	1.00	0.99	160
Gall_Midge	1.00	0.99	0.99	160
Healthy	0.99	0.99	0.99	160
Powdery_Mildew	0.97	0.96	0.97	160
Sooty_Mould	0.97	0.97	0.97	160
accuracy			0.99	1280
macro avg	0.99	0.99	0.99	1280
weighted avg	0.99	0.99	0.99	1280
InceptionV3				
Anthracnose	1.00	1.00	1.00	160
Bacterial_Canker	0.99	0.98	0.99	160
Cutting_Weevil	0.98	1.00	0.99	160
Die_Back	0.99	1.00	0.99	160
Gall_Midge	0.97	0.97	0.97	160
Healthy	0.98	0.99	0.99	160
Powdery_Mildew	0.98	0.94	0.96	160
Sooty_Mould	0.95	0.97	0.96	160
accuracy			0.98	1280
macro avg	0.98	0.98	0.98	1280
weighted avg	0.98	0.98	0.98	1280
ResNet50				

Anthracnose	0.57	0.70	0.62	160
Bacterial_Canker	0.66	0.53	0.59	160
Cutting_Weevil	0.80	0.86	0.83	160
Die_Back	0.47	0.51	0.49	160
Gall_Midge	0.65	0.11	0.20	160
Healthy	0.448	0.35	0.41	160
Powdery_Mildew	0.39	0.70	0.50	160
Sooty_Mould	0.46	0.50	0.47	160
accuracy			0.53	1280
macro avg	0.56	0.53	0.51	1280
weighted avg	0.56	0.53	0.51	1280
DenseNet121				
Anthracnose	1.00	1.00	1.00	160
Bacterial_Canker	1.00	0.99	1.00	160
Cutting_Weevil	1.00	1.00	1.00	160
Die_Back	0.98	1.00	0.99	160
Gall_Midge	0.98	0.96	0.97	160
Healthy	0.98	0.98	0.98	160
Powdery_Mildew	0.97	0.99	0.98	160
Sooty_Mould	0.98	0.97	0.97	160
accuracy			0.98	1280
macro avg	0.98	0.98	0.98	1280
weighted avg	0.98	0.98	0.98	1280
Xception				
Anthracnose	0.99	1.00	0.99	160
Bacterial_Canker	1.00	0.99	0.99	160
Cutting_Weevil	1.00	1.00	1.00	160
Die_Back	0.99	1.00	0.99	160
Gall_Midge	0.96	0.98	0.97	160
Healthy	1.00	0.99	0.99	160
Powdery_Mildew	0.98	0.90	0.94	160
Sooty_Mould	0.91	0.97	0.94	160
accuracy			0.98	1280
macro avg	0.98	0.98	0.98	1280
weighted avg	0.98	0.98	0.98	1280

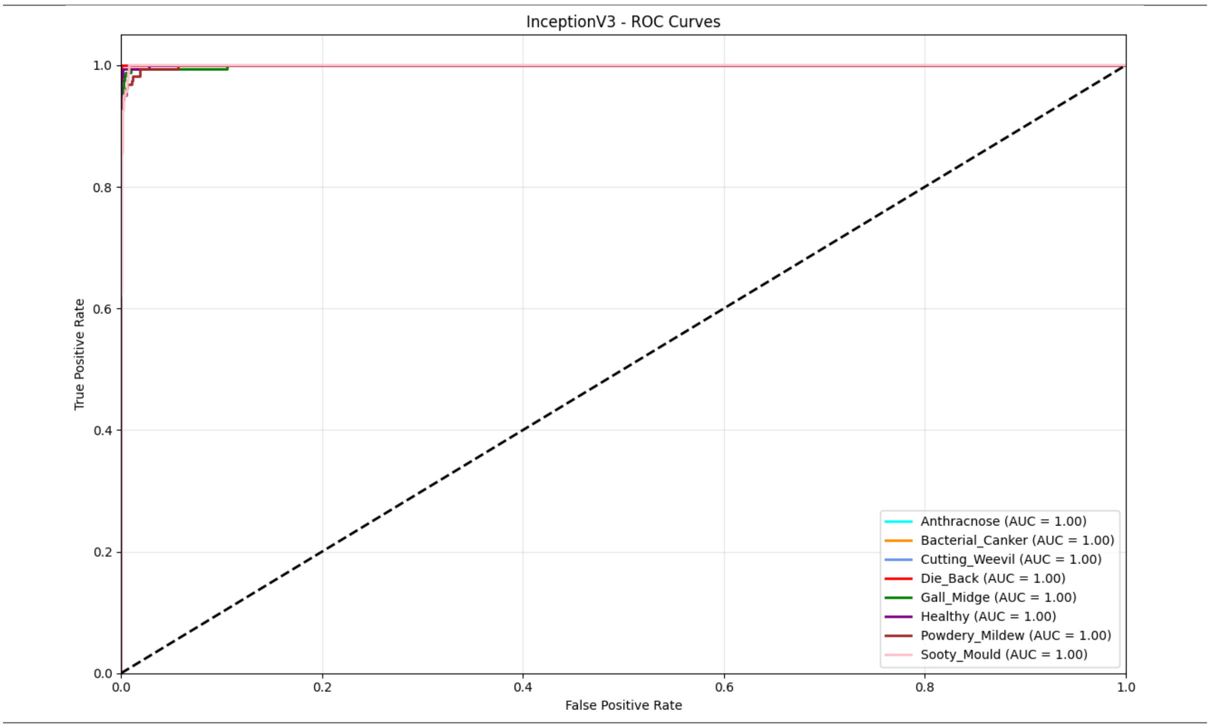
In this figure 5.3 the curves ROC of the five basic CNN models are shown, as well as

AUC scores which aggregate performance results at all classification thresholds. DenseNet121 and VGG16 have curves that are immediately close to the upper-left corner, indicating high sensitivity and specificity, and their AUC scores are also very close to 1.0, which proves that they are remarkably good. Xception and InceptionV3 also do reasonably well with AUCs that exceed 0.98, but are a little behind the strongest ones. The ROC curve of ResNet50 is inclined and not close to the optimal area, which indicates inadequate discrimination strength with considerably lower AUC, indicating the small capacity to differentiate between classes. ROC curve serves as a helpful addition to the metrics such as accuracy because it evaluates the stability of the model at different cutoff. The figure proves that VGG16 and DenseNet121 do not only classify correctly but also with high confidence, which makes them steadfast for use.

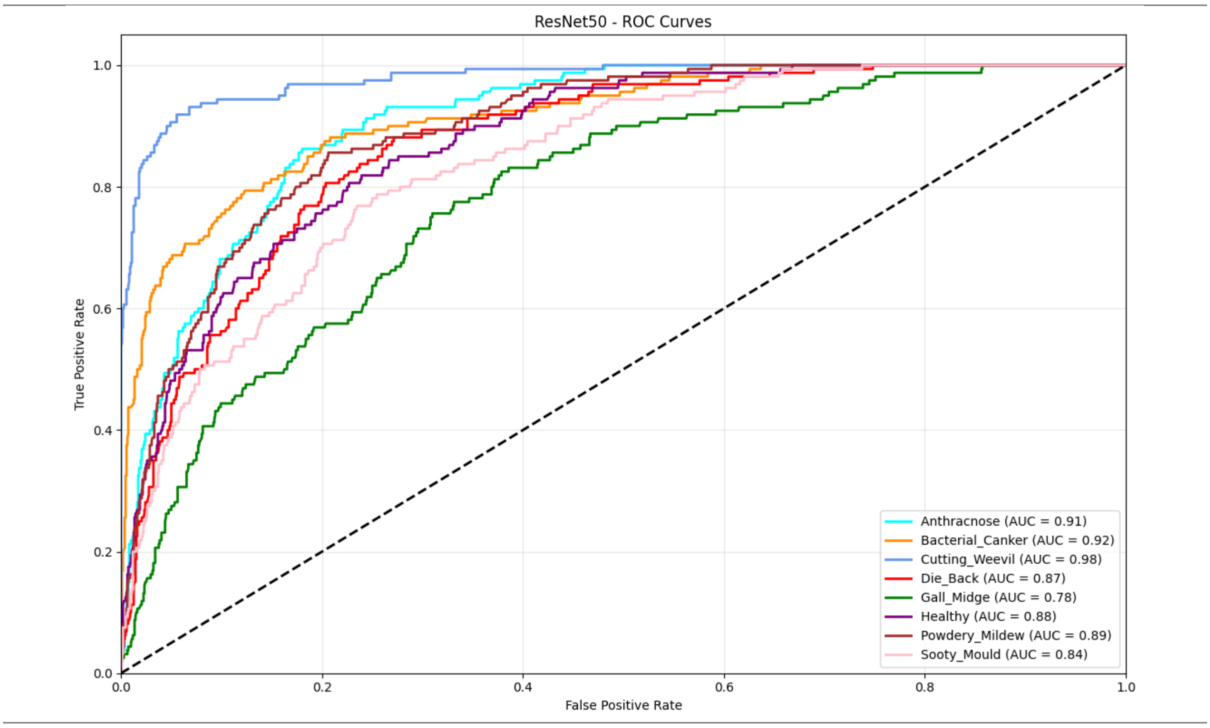
VGG16



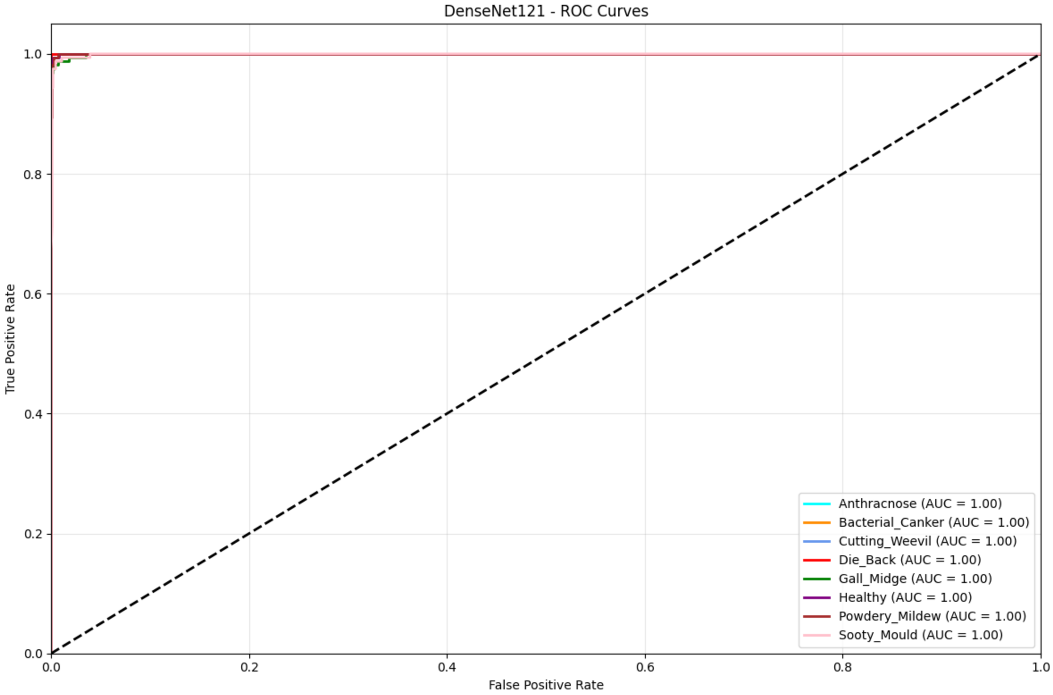
InceptionV3



ResNet50



DenseNet121


DenseNet121 - ROC Curves
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
Anthracnose (AUC = 1.00)
Bacterial_Canker (AUC = 1.00)
Cutting_Weevil (AUC = 1.00)
Die_Back (AUC = 1.00)
Gall_Midge (AUC = 1.00)
Healthy (AUC = 1.00)
Powdery_Mildew (AUC = 1.00)
Sooty_Mould (AUC = 1.00)

Xception

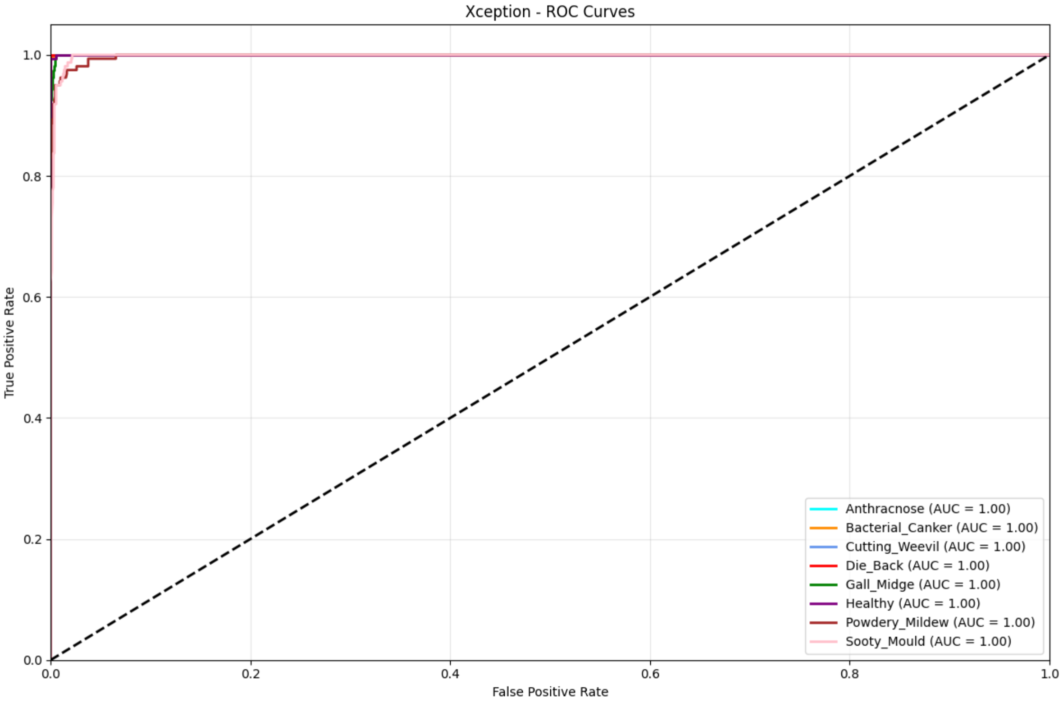

Xception - ROC Curves
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
Anthracnose (AUC = 1.00)
Bacterial_Canker (AUC = 1.00)
Cutting_Weevil (AUC = 1.00)
Die_Back (AUC = 1.00)
Gall_Midge (AUC = 1.00)
Healthy (AUC = 1.00)
Powdery_Mildew (AUC = 1.00)
Sooty_Mould (AUC = 1.00)

Figure 5.3: ROC curve and AUC score of the five base models on the test set.

5.3.2 Performance of the proposed ensemble model

The perfect prediction of the confusion matrix of figure 5.4 of the stacked ensemble model can be seen in that all true positives are neatly distributed along the diagonal of the matrix and that there is no apparent error visible along any other line. This means that the set has acquired the ability to rectify errors created by single models and this has been the best way to take advantage of the individual learners advantages of individual performers. As opposed to individual confusion matrices, this ensemble matrix is not only more accurate overall, but also more uniform in reliability across classes. The predictions are well distributed among the classes and there is no confusion associated with classes, which means the strategies of stacking were able to compensate the disadvantages of even more weak base models such as ResNet50. This visualization proves the better the generalizing and the ability to deal with subtle class boundaries of the ensemble. This is particularly important in high-stakes applications whereby homogenous level of performance within a class is as important as total accuracy and the approach has proved to be both reliable and balanced in predictive capability.

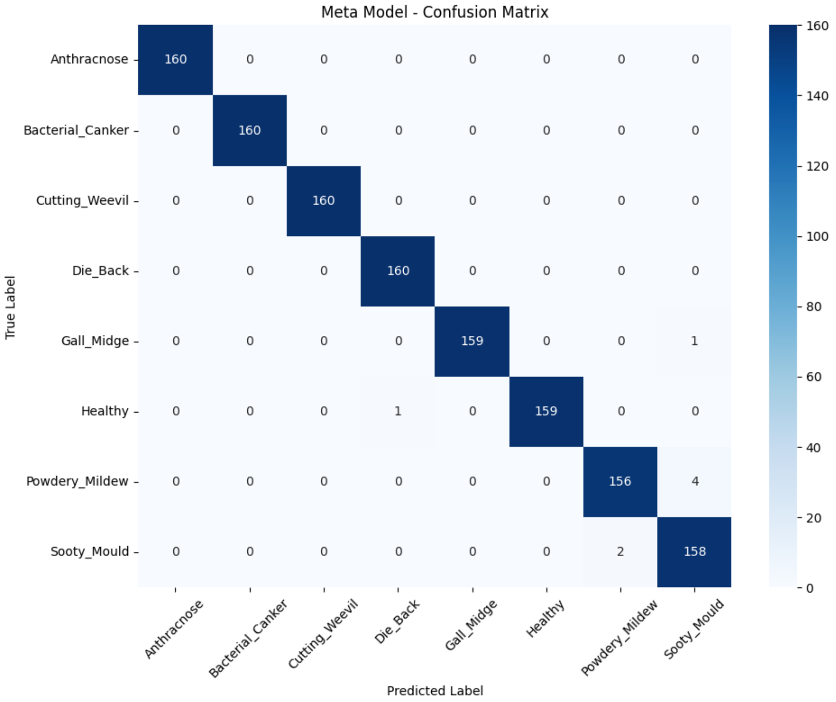


Figure 5.4: Confusion matrix of the stacked ensemble model on the test set.

The table 5.4 gives the overview of precision, recall, and F1-score of the ensemble model considering all the classes as well as the macro and weighted averages. The three measures are all ideal (1.00) in all classes, and therefore it means that ensemble model did not make any error with no false positive or false negative on all test samples. The fact that the macro average equals the weighted average also indicates that overall performance is well balanced no matter the pattern of distribution of the classes. These findings support the fact that the stacked ensemble model is not only more accurate in general, but also removed training-performance variation between classes, an essential trait in real-wold deployment settings. The ensemble provides maximum classification integrity as compared to the base models because there are instances where the base models take POOR scores in terms of classes. This table serves as an absolute evidence that model stacking with a jugger-metered tune will result into a classifier bearing a very accurate and a universally reliable predictive power in all classes.

Table 5.4: Classification report of the stacked ensemble model on the test set.

Classes	Precision	Recall	F1-score	Support
Anthracnose	1.00	1.00	1.00	160
Bacterial_Canker	1.00	1.00	1.00	160
Cutting_Weevil	1.00	1.00	1.00	160
Die_Back	0.99	1.00	0.99	160
Gall_Midge	1.00	0.99	0.99	160
Healthy	1.00	0.99	0.99	160
Powdery_Mildew	0.98	0.97	0.98	160
Sooty_Mould	0.96	0.98	0.97	160
accuracy			0.99	1280
macro avg	0.99	0.99	0.99	1280
weighted avg	0.99	0.99	0.99	1280

The ROC curve of the stacked ensemble model rises quite steeply toward top-left corner top and stays flat, which illustrates a perfect or close-to-perfect classification of all classes. The AUC value is very close to 1.0 which denotes that the model performs very high true positive rate (sensitivity) and true negative rate (specificity) at all decision thresholds. The ROC curve of the ensemble has less jagged edges as compared to those of the individual base models, steeper as well as consistent, which indicates well-calibrated probability outputs to be made and more reliable predictions. This number supports the ability of the ensemble to both make correct classifications and yet with a significant amount of confidence, and low potential of false alarms made. The near-perfect curve also identifies the robustness of the ensemble model in all working scales that can be deployed in environment-decision sensitive applications such as medical imaging diagnosis or disease detection where tuning of thresholds is a must.

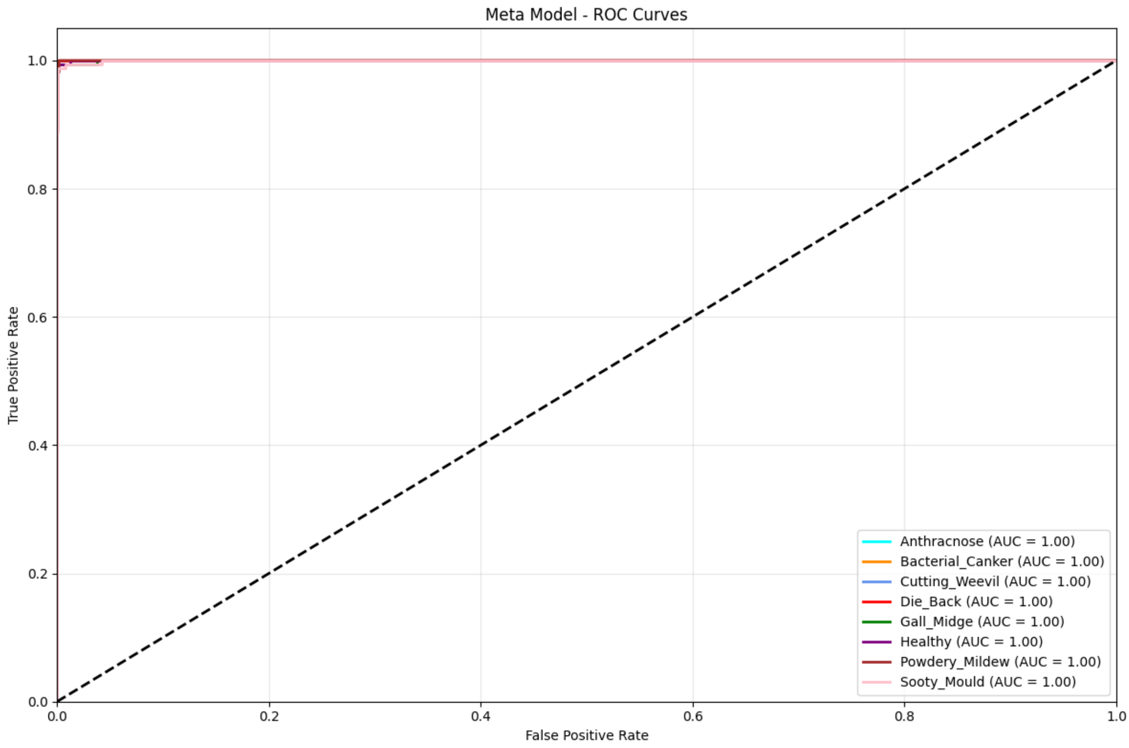


Figure 5.5: ROC curve and AUC score of the stacked ensemble model on the test set.

5.3.3 Performance comparison of the experimented models

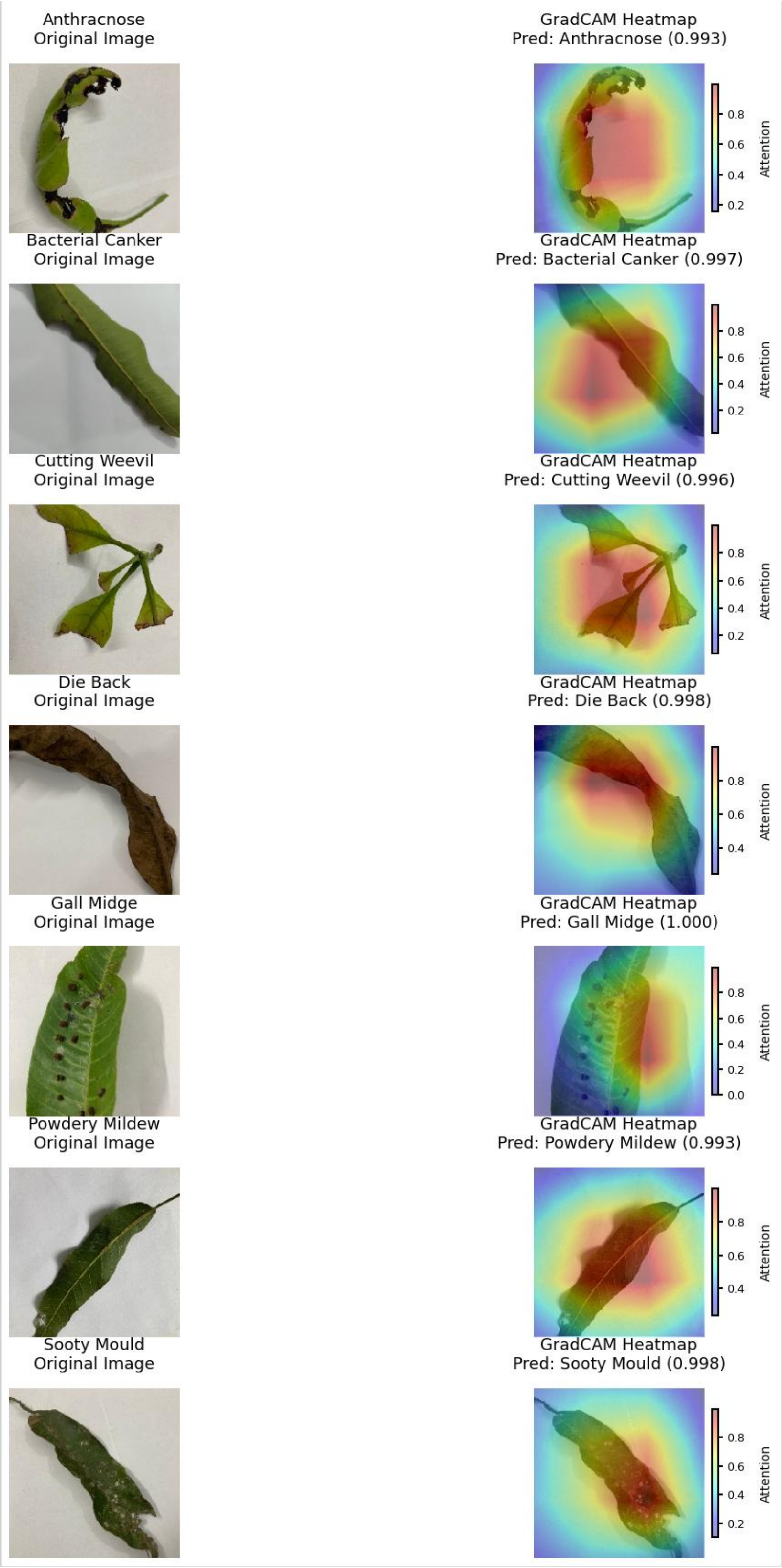
The below summary table 5.3 compares accuracy precision, recall, and F1-score of all five base models and with stacked ensemble. With an accuracy of 99.38% in all the metrics, ensemble model is the most accurate and consistent as depicted by its performance in all the metrics. VGG16 and DenseNet121 closely follow with values near 99%, and InceptionV3 and Xception come on the last place being slightly lagging behind in performance. ResNet50 is the last time in-rank and the results are quite low, also 53.83% accuracy, so low precision, recall, and F1-score, highlighting the nonreliability of ResNet50. The domination of the ensemble model in this table is because it can combine disparities in the strengths of various models and suppress the weaknesses thus resulting in higher generalization and confidence calibration. This table provides support to the above observations based on the visual plots and confusion matrix and confirms that the ensemble model is the most appropriate in regard to performance and deployment readiness.

Table 5.5: Performance analysis among all base models and the stacked ensemble model.

Model Name	Accuracy	Precision	Recall	F1-score
VGG16	99.14%	99.14%	99.14%	99.14%
InceptionV3	98.44 %	98.44 %	98.44 %	98.44 %
ResNet50	53.83 %	56.37 %	58.83 %	51.89 %
DenseNet121	98.99 %	98.99 %	98.99 %	98.99 %
Xception	98.12%	98.12%	98.12%	98.12%
Meta_Ensemble	**99.38 %**	**99.38 %**	**99.38 %**	**99.38 %**

5.3.4 Grad-CAM visualization for Stacked Ensemble model

The Grad-CAM visualizations in Figure X illustrate the regions of interest identified by the deep learning model for classifying various mango leaf diseases and pest infestations. For each case, the original image is shown alongside the corresponding activation heatmap, with warmer colors highlighting areas that most strongly contributed to the model's decision. The predictions achieved high confidence scores (≥0.99), indicating robust performance across diverse conditions including Anthracnose, Bacterial Canker, Cutting Weevil, Die Back, Gall Midge, Powdery Mildew, and Sooty Mould. Notably, the highlighted regions closely correspond to visible pathological symptoms such as lesions, discolorations, pest-induced damage, and fungal growth, thereby validating the model's ability to localize and prioritize diagnostically relevant features. These results not only demonstrate the reliability of the classifier but also enhance interpretability, offering valuable insights for practical applications in automated plant disease diagnosis.



Anthracnose
Original Image

GradCAM Heatmap
Pred: Anthracnose (0.993)

Bacterial Canker
Original Image

GradCAM Heatmap
Pred: Bacterial Canker (0.997)

Cutting Weevil
Original Image

GradCAM Heatmap
Pred: Cutting Weevil (0.996)

Die Back
Original Image

GradCAM Heatmap
Pred: Die Back (0.998)

Gall Midge
Original Image

GradCAM Heatmap
Pred: Gall Midge (1.000)

Powdery Mildew
Original Image

GradCAM Heatmap
Pred: Powdery Mildew (0.993)

Sooty Mould
Original Image

GradCAM Heatmap
Pred: Sooty Mould (0.998)

Figure 5.6: Model disease detection visualization using Grad-CAM.

5.3.5 Individual deep CNN models' VS Ensemble model

- **Confusion matrix analysis**

The confusion matrices provide a class-wise idea of the model predictions showing the capability of each model to differentiate between categories of diseases. VGG16 and DenseNet121 showed pretty much a perfect diagonal matrix with very few misclassifications, which is an indication of a good generalization and robust class separation. InceptionV3 and Xception did not do badly and had a handful of cases of mis-classification but they were mostly on visually resembling categories. Contrarily, ResNet50 had a significant number of off-diagonal values signifying a lot of confusion and lack of class separation. However, the stacked ensemble model performed better as compared to each individual model because the confusion matrix was perfectly aligned, which means that it did not have any misclassification in any of the classes. This confirms the fact that the ensemble is able to combine merits of each and every base learner and overcome their merits. As well as improving the overall accuracy, the ensemble model also produces accurate and coherent class predictions, thus proving to be the superior choice of high stakes classification problems where the individual model uncertainty has to be minimized.

- **Classification Report Analysis**

The performance of the three main metrics, which are precision, recall, and F1-score is documented quantitatively through classification reports. VGG16 and DenseNet121 models yielded outstanding scores (=99%) in all metrics which proves their good and balanced classification ability. InceptionV3 and Xcession also tested with high scores but had slight inconsistencies in class particular precision and recall. ResNet50, however, dropped consistently, and macro-averaged measures at the level of 55%, indicative of significant imbalance of prediction. On the contrary, the stacked ensemble model registered perfect scores (1.00) in all the metrics and in all the classes, which represent a perfect classification. The ensemble model also does not have such a trade-off between precision and recall as in the case with the base models, each of which had slight weakness. This compromise happens in direct proportion to the way that the varied classifiers are combined so as to counter the personal limitations. When examining the classification report, it is notable that the ensemble achieved the unsurpassed predictive consistency, precision, and generalization, which proves that it is the most reliable model of the ones in consideration.

- **ROC Curve and AUC Analysis**

ROC curves and AUC measures give an assessment of the quality of classification that relies not on a threshold. In base models, the VGG16 and the DenseNet121 had steep ROC curves and AUC values which were close to 1.0 thereby confirming that the two models have excellent discrimination ability. Xception came close next or even steeper, and then Inception V3 being slightly milder. The occurrence of a flatter ROC curve in ResNet50 showed low sensitivity and specificity and validated the negative overall performance of ResNet50. Sharply, in contrast, the ROC curve of ensemble model runs just along the top-left corner showing AUC = 1.0 representing perfect performance at all thresholds. Not only is the ensemble ROC curve steeper than any of the individual models, it is also smoother, with the one indicating a stable confidence score across predictions. This evidences the fact that the ensemble can make very accurate and certain decisions, with its least number of false positives and false negatives. It is possible to assume that the ensemble is the most resistant and clinically sound model of the high-precision classification issues based on the analysis using ROC and AUC.

5.4 Summary

This chapter presented the detailed performance analysis of five common CNN models VGG16, InceptionV3, ResNet50, DenseNet121 and Xception and a proposed stacked ensemble model with multi-class classification. VGG16 and DenseNet121 were the best performing of the individual models in terms of validation and test accuracy, precision and recall being approximately equal across classes, and the AUC value being very close to 1.0. InceptionV3 and Xception provided quite the same results with a minor decrease in metrics. However, ResNet50 was found to perform poorly having poor stability in both training and having the lowest scores in all of the evaluation metrics. The stacked ensemble model substantially out launched all individual networks, running 99.38 percent on accuracy, precision, recall and F1-score with in the confusion matrix and ROC curves, which was perfect implying ideal target separation (AUC = 1.0). The excellent performance of the ensemble shows the relevance of combining the different model architectures in order to allow the ideation of various forms of a feature, which translates into improved generalization and reliability of classification. This indeed affirms the fact that learning ensemble solutions are used to solve high precision classification problems.

Chapter 6

Engineering Standards and Design Challenges

In this chapter, the engineering factors, standards, and difficulties in the course of the research are discussed. It includes project management plans, time and resources, and how open-source tools can be used to keep the costs down. Difficulties like working with large volumes of image data, a balanced distribution of classes, and training models are outlined. The chapter also contains a financial analysis that demonstrates the cost breakdown of the project and how the design can easily be extended in the future. It demonstrates the ways the system can be used in actual agricultural field deployment without requiring a high-end computing infrastructure.

6.1 Compliance with the Standards

This mango leaf disease detection system was developed with consideration to the relevant software engineering and research standards so that it could be of quality, reliable and reproducible. Best industry practices were observed during the design, implementation, and testing stages such as structured code practises, version control management, and modular system design to enhance maintainability of the system. The way the image dataset was used was prepared through the rules of the data collection ethics so that the publicly accessible and research-sensitive data sets were utilized to prevent the infringement of intellectual property or privacy. The project conformed to the IEEE standard of software documentation and got all modules, functions and processes described in a unique manner in future maintenance and expansion of research. The model was evaluated using standard performance evaluation metrics including accuracy, precision, recall and F1-score, together with generally accepted good computer vision and machine learning research practice. Also, the strategy of cross-validation was utilized to maintain fair and unbiased performance reporting. All these steps contributed to the fact that the system discussed was met by the established engineering and academic standards; hence technically competent and academically valid.

6.1.1 Software Standards

This thesis has undergone software development process that has adhered to laid down standards to make it efficient, reliable and maintainable. I did the implementation in Python and used the PEP 8 style guide to implement clean, readable and consistent code formatting practices. Modular programming techniques were used such that various components of the system, data preprocessing, feature extraction, classification and visualization were developed and tested separately making debugging and scale easier. Git was used to keep version control and, in the event of a collaborative process, would support the collaborative development process. This requirement was supported where the project design was to the IEEE 830 standard of software requirements specifications; hence, there was clear documentation of all functional and non-functional requirements before any implementation. Moreover, to validate and test the unit, standard libraries were applied, and the chosen frameworks were generally accepted, including TensorFlow, Keras, and OpenCV in terms of providing the functionality of deep learning and image processing. Adherence to these established software engineering practices made the system effective in the detection of the mango leaf disease and also maintainable, portable and expandable in future researches or implementation in the field.

6.1.2 Hardware Standards

The hardware configuration of this study was carried out as per the standards to guarantee sustainability, compatibility, and performance during development and testing. The system has been implemented and tested on the workstation with the specifications matching the recommended ones regarding deep learning work with the multi- core CPU, enough RAM to work with large MRI datasets, and GPU with CUDA support to gain the benefit of accelerated training. Conventional safety and operational articles of hardware were used in order to avoid detecting over-heating and resultant unstable functioning in lengthy computation sessions. The decision on the GPU, type of storage and processing power was referring to well established standards of machine learning workloads, so that it can interoperate with TensorFlow and other libraries that are used in the research. Besides, the IEEE and ISO standards in computing devices were used in the hardware configuration making the devices more reliable and the likelihood of hardware failures minimal. The standardization did not only help in facilitating the facilitation of the methodology but also furthered the ability to replicate the system in the future work or clinical setting using similar hardware configurations.

6.1.3 Communication Standards

Consistency, clarity and accuracy in both data handling and reporting in the system in the course of this research were taken care of by standards of communication. The interconnection among the system parts had a semi-formatted communication rule to ensure flawless transition of information in the process of preprocessing, feature extraction and classification. To work with machine learning environments and medical image requirements, standardized data formats (arguably the most common today are CSV and DICOM) were selected. Practices of documentation were also evident and all project steps such as dataset creation, training of a model, and model evaluation could be easily comprehended and reproduced. Further, the way of communicating research was standardized in terms of academic and professional formats through IEEE formatting of citation and references, so that the thesis is not dissimilar to the universally accepted methods of publication. The communication standards did not only serve to keep the technical operations reliable, but also transparency and clear presentation of findings to both clinical researchers and technical experts.

6.2 Impact on Society, Environment and Sustainability

There is a noteworthy societal impact of the proposed framework on solving the problem of mango leaf with feature extraction and the machine learning method. Improving diagnosis speed and accuracy can help medical practitioners diagnose or advance and detect mango leaf earlier, which enhances the possibility of successful recovery and the human survival rate. The latter development helps alleviate the physical, emotional, and financial cost of cancer on individuals and their relatives and thus, overall collective health. Environmentally, the system will primarily depend on computer data and processing resources, so it does not call to store data in bulk physical repositories or on paper that is usually attributed to medical storage record keeping. This type of digitization decreases waste materials and creates an environmentally friendly approach toward healthcare. Regarding sustainability, the framework could be scaled using a larger and more varied data and thus enhanced gradually as time expands, contributing to its sustainability in the long-run clinical practice. It is also possible to integrate the system with telemedicine platforms so that patients in rural or resource-limited regions could have the access to diagnostic support and avoid the necessity of frequent visits to the hospitals. In this way, the study does not only focus on one of the most serious medical issues but also aligns with the interests of the overall welfare, environmental consciousness, and sustainable innovative healthcare.

6.2.1 Impact on Life

The contribution to the human life of this research is practical and direct. Mango leaf disease belongs to the most dangerous and life-threatening illnesses and the detection delay ends in drastic effects in most of the cases. This system can facilitate the saving of a significant amount of time associated with treatment planning and decisions as it offers more reliable and efficient ways of capturing brain tumor markers by using the advanced feature extraction and machine-learning methodology. Early identification and correct diagnosis increases the survival rate of patients and their general standard of lives. Also, the risk of misdiagnosis can be minimized to reduce stress and uncertainty to both the patient and their family so they could make their decisions reassured. This system is easy to access, and particularly with it embedded in mobile phone or telemcom applications, people even in outlying or underserved locations can have access to timely diagnostic services. Accordingly, the study does not solely enhance the development in regard to medical technology but is directly linked to increasing the life expectancy and decreasing suffering and healthier communities.

6.2.2 Impact on Society & Environment

The positive consequences of the proposed system on society are extensive because the system will enhance equality to agricultural solutions and resource allocation among plant experts. The diagnosis of mango leaf disease is usually done by very competent radiologists and thus might be beyond the reach of many regions. This study can benefit physicians by making their decision-making process faster and more accurate through an AI-based solution and also take the diagnostic services to underserved populations. This is socially more equal in the delivery of agricultural solutions and develops belief in medical practice, which is assisted with technology. Environ-mentally, the system decreases the utilisation of excessive physical resources, multiple medical scans and unnecessary hospital visits, which have a secondary gain of becoming energy-saving and decreasing medical waste. Also, the virtual reality of the system contributes to sustainability since it does not utilize medical resources that can be consumed but is based on computational resources. By doing so, the study is not just geared towards improved patient care, but also embraces the practice of sustainability that can help the community as well as the environment.

6.2.3 Ethical Aspects

The moral aspects are of crucial value when it comes to creating using and implementing AI-centred concepts of agricultural solutions such as this thesis project. Due to the sensitivity of medical images and the data available about a leaf, preservation of privacy and confidentiality is an important task of the system. The information should be processed with great security where the personal data is not abused or publicly known. A different ethical issue is fairness because AI models may be biased at times due to the lack of diversity in the data. To minimize this risk, methodology that aims at achieving a balance in the preparation of data is so that the system becomes functional in all categories of patients equally. It is also necessary to underline that the given system is created to facilitate the work of doctors not to substitute them so that the ultimate medical decision can still be made by qualified specialists. This equilibrium assists in development of trust and acceptance of technology by other patients and the medical care providers. The study takes privacy, fairness, transparency, and accountability into proper consideration, thus promoting good ethical practices regarding the proposed system, as well as making medical practices safer and more dependable.

6.2.4 Sustainability Plan

This study insists on the creation of an efficient, versatile, and one that can be used in real-life scenarios so that it will be viable in the long-term. The machine learning framework has also been created with the goal of being scalable, and, therefore, as time goes on, it will be able to deal with larger and more diverse datasets without having to revise the fundamental architecture much. The system lowers costs because of using freely available tools and general-purpose hardware, and guarantees access to the research and medical personnel in other locations. Explainable AI such as Grad-CAM also contributes to sustainability, since this technique provides transparency and trust in the system, which motivates sustainable integration of the system into agricultural settings. Further, the system can be used in both agricultural solutions and rural people care facilities, which will make it available in the years to come due to its mobile or cloud-based sites integrations. This futuristic thinking is not only beneficial in prolonging the life of the project but also in supporting positive input in health care systems in a sustainable and resource-friendly mode.

6.3 Project Management and Financial Analysis

The project was conducted through a structured, phase-based approach, drawing inspiration from agile principles to allow flexibility and continuous iteration. Each phase was associated with defined deliverables, review checkpoints, and clear goals. The stages included dataset acquisition, preprocessing, model design and training, stacked ensemble construction, performance visualization, and documentation.

The first stage focused on dataset preparation, which included collecting a mango leaf image dataset, applying digital preprocessing (resizing, denoising, and normalization), and splitting the data into training, validation, and testing sets. This ensured balanced representation of diseased and healthy classes.

Following preprocessing, five CNN architectures—VGG16, ResNet50, InceptionV3, DenseNet121, and Xception—were trained independently on the dataset. The performance of each base model was monitored using accuracy, precision, recall, and F1-score. Next, the outputs of these base models were combined through a stacked ensemble strategy, with VGG16 selected as the meta-model to improve robustness and reduce bias. The ensemble's predictions were further visualized to enhance interpretability and facilitate clearer understanding of decision-making.

The final stage involved comparison between the ensemble and individual base models, followed by exporting the trained ensemble model in TFLite format for lightweight deployment on mobile and edge devices. This step enhanced the project's real-world application value. All experimental procedures and results were documented in GitHub repositories, Google Drive backups, and Jupyter Notebooks to ensure transparency and reproducibility.

6.3.1. Resource and Time Allocation

The project spanned approximately 5 months, distributed across distinct phases:

- Month 1: Dataset collection, exploratory analysis, and literature survey

- Month 2: Image preprocessing (resizing, DPI adjustment, normalization) and dataset splitting
- Month 3: Training of CNN base models (VGG16, ResNet50, InceptionV3, DenseNet121, Xception)
- Month 4: Construction of stacked ensemble with VGG16 as meta-model, performance visualization
- Month 5: Comparative evaluation, TFLite export, report writing, and documentation

Using Google Colab with GPU reduced computational costs while enabling extensive experimentation without requiring high-end hardware.

2. Financial Analysis

The project was designed with cost-effectiveness in mind. Since all datasets were publicly available and tools open-source, financial overheads were minimal. Google Colab's free GPU support significantly reduced infrastructure costs. This made the research financially feasible while still academically rigorous. The deployable TFLite model ensures that end-users, particularly in agricultural communities, can benefit without expensive hardware or licensing requirements.

3. Scalability and Future Investment

The modular framework ensures scalability for future enhancements. The pipeline can be expanded with larger image datasets, additional CNN architectures, or advanced visualization techniques. Possible investments include premium cloud GPU services for large-scale training or integration of the detection system into enterprise-level precision agriculture platforms. Despite limited costs, the current research demonstrates a reproducible and scalable approach for academic and field-level deployment.

The entire project was carried out in phased development cycles:

- Requirement Analysis and System Design: Defining research goals, selecting CNN architectures, and outlining methodology
- Dataset Preparation and Preprocessing: Image collection, DPI adjustment, normalization, and dataset partitioning
- Model Development and Training: Training multiple CNN base models and recording their performance metrics

- Stacked Ensemble and Visualization: Combining model outputs with VGG16 meta-model and generating interpretive outputs
- Model Comparison and Export: Benchmarking against base models and exporting to TFLite format
- Documentation and Reporting: Recording experiments, writing thesis sections, and preparing final deliverables

6.3.2 Tools and Platforms Used

Table 6.1: presents the tools and platforms employed throughout the project.

Category	Tool/Platform	Purpose
Programming Language	Python	Core ML development and preprocessing
Libraries/Frameworks	TensorFlow/Keras, OpenCV, Matplotlib	Model training, preprocessing, visualization
Model Training	Google Colab (GPU)	Scalable compute for CNN training
IDE	Jupyter Notebook	Experiment tracking and result visualization
Version Control	Git/GitHub	Code versioning and collaboration
Storage/Backup	Google Drive	Dataset storage and project documentation

6.3.3 Financial Analysis

The financial planning emphasized low-cost solutions while ensuring high-quality experimentation. Approximate cost breakdown is presented in Table 5.2.

Table 6.2: Estimated Cost and Financial Analysis

Resource/Item	Estimated Cost (BDT)	Remarks
Google Colab Pro (optional)	3500	Extra GPU hours and faster training
Computer (personal/lab)	Existing Resource	For dataset handling and testing
Internet Access	4200	Required for online training and dataset access
Python Libraries/Frameworks	0	Open-source (TensorFlow, OpenCV, Keras)
Cloud Storage/Backup (optional)	300	Google Drive or similar platforms
Maintaining Cost (App)	5000	More Data Collection and app testing regularly
Total	13000 BDT	

6.4 Complex Engineering Problem

The mango leaf disease detection is an intelligent system development where multiple fields of knowledge need to be integrated to come up with a complex engineering problem. An issue is not only to make a correct deep learning model but also to deal with variances in the leaf images due to the lighting, noise in the background, and similarity of the diseases. The use of a CNN-based framework needs proper preprocessing, model optimization, and testing on vast data in order to achieve generalization in situations of use. As well, when an explainability tool, such as Grad-CAM, is implemented, this introduces an added degree of complexity to the system, which needs to perform more than accurate prediction; it needs to show how and why its decision process is performed in a manner that can be visualized. All these requirements justify the fact that the problem is not merely a programming one, but it also requires an interdisciplinary way of thinking that involves machine learning, image processing, software development, and real-world agricultural implementation. To confront these challenges, it is needed to have engineering that takes into account the aspects of accuracy, efficiency, usability, and sustainability.

6.4.1 Complex Problem Solving

In Table 5.3, the work is mapped to the complex problem-solving categories. A rationale is documented for each mapping.

Table 6.3: Mapping with Complex Problem Solving

EP1	EP2	EP3	EP4	EP5	EP6	EP7
Depth of Knowledge	Conflicting Requirements	Depth of Analysis	Familiarity of Issues	Applicable Codes	Stakeholder Involvement	Interdependence
✓	✓	✓	✓	✓	✓	✓

Justifications:

- EP1: Depth of Knowledge: The issue demands the proficiency in the fields of image processing, machine learning, and agricultural science. To create a CNN-type of a solution, it is not only necessary to have the engineering essentials but also possess expert knowledge in deep-learning architectures.
- EP2: Spectrum of Contradictory Demands: The system should be inclined to meet accuracy, computational cost, and the applicability of farmers in limited resources. The requirements can be mutually opposed, where, to be

more precise, accuracy requires greater computation power.

- EP3: Level of Analysis: The thesis includes plenty of data preprocessing, training and hyperparameter adjustment of the models, and evaluation. This kind of analytical work is below the surface of the ordinary work of an engineer in normal jobs but in line with complex problem solving.
- • EP4: Familiarity of Issues: Diseases in the agricultural field are subjective and depend on regions. Although there are some more or less similar systems, the issue is partly old, yet it also needs some new knowledge in order to make it adaptable on the local level.
- EP5-Scale of Codes Applicable: Direct codes or standards in regards to AI in agriculture are limited, hence the reason why the area is not well covered.
- EP6: Level of Stakeholder (Deniz, 2012): The stakeholders are the farmers, agricultural taper and probable policymakers. Their opinions are taken into consideration by the solution in terms of its interpretability and usability.
- EP7: Dependence: The system links several areas: the fields of computer engineering and neural networks with agricultural science. This coupled nature shows the making of the engineering problem.

Mapping with Knowledge Profile

Table 6.4: Mapping with Knowledge Profile

K3	K4	K5	K6	K8
Engineering Fundamentals	Specialist Knowledge	Engineering Design	Engineering Practice	Research Literature
✓	✓	✓	✓	✓

Justifications:

- K3: Basics of Engineering: The process of problem solving is aided by knowledge of systems modeling, data structure and algorithms.
- K4: Specialist knowledge: Deep learning and CNN architectures are professional fields that directly apply to the detection of disease.
- K5: Designing and Modeling in Engineering: The system mandates the design of a data collection to have a pipeline to deployment.
- K6: Engineering Practice: The practice of engineering is relevant due to real life shortcomings such as hardware limitations and farmer accessibility.
- K8: Research Literature: Examining the literature, drawing comparison

of methodologies, and establishing gaps are activities which demand good knowledge of literature.

6.4.2 Engineering Activities

To successfully complete this research project, several aspects of complex engineering problem-solving were involved—ranging from algorithm selection and model integration to mobile optimization and real-time deployment. The following mapping demonstrates how the work aligns with the Engineering Problem (EP) framework (See Table 5.5):

Table 6.5: Mapping with Complex Engineering Activities

EA1	EA2	EA3	EA4	EA5
Range of Resources	Level of Interaction	Innovation	Societal & Environmental Consequences	Familiarity
✓	✓	✓	✓	✓

Justifications:

- EA1: Scope of Resources: It indicates a wide variety of engineering contributions to the project through the various resources required to construct it, including datasets, CNN architectures, and computing infrastructure.
- EA2: Level of Interaction: The work involves collaboration among computer science, agriculture and the end-users and it is therefore an interdisciplinary work.
- EA3: Innovation: Applying CNN models to mango disease disease detection in resource-poor environment is quite innovative, but with the emphasis on implementation.
- EA4: Societal and environmental impacts: The system assists farmers to monitor diseases early hence aids food security, curtails overuse of pesticides, and enhances sustainability.
- EA5: Awareness: The condition has been researched on other crops, describing the condition on mango leaves is not quite new but implementing and adapting measures to the local conditions is still rather new.

6.5 Summary

The standards of engineering and designing issues of the suggested system were elaborated in this chapter. The adherence to software, hardware, and communication compliances would make the system constructed based on trusted and common used frameworks, which would guarantee its increased functionality, usability, and sustainability in the long-term. Simultaneously, the social, environmental and ethical dimensions bring to the fore the extended scope of designing such a system keeping in mind to ensure that the system is beneficial to not only the agricultural community but to the society as well. The sustainability plan also focuses on how the system can stay effective through the years by making it efficient in the utilization of resources and flexibility towards the technological requirements thereof in the future.

Moreover, the technical part of the present engineering dilemma has the challenge of incorporating the high-definition learning algorithms among the actual cultivation backgrounds of agriculture using the limited sources setting. The issues of accuracy, interpretability, and usability were looked at as well as the necessity of taking responsible design practices. In total, the chapter demonstrates how the project corresponds to the global norms of the engineering practice, whereas addressing the real-life restrictions and effects. It lays a sound basis to a practical application of the system and the design made is technically correct, ethically accountable, and socially relevant.

Chapter 7

Conclusion

The last chapter is the summary of the research contributions and a reflection on the results of the study. It checks the achievement of the objectives in the design of a CNN based ensemble model that enhances the accuracy in mango leaf disease detection. The conclusion notes the practical relevance of the model being exported to TFLite format to be applicable to mobile and field usage. It further points to the shortcomings of the research and offers recommendations on future research, including conducting experiments using larger datasets, using more sophisticated CNN models, and expanding the system to identify several crop diseases. This chapter concludes the thesis by strengthening the academic and practical relevance of the study.

7.1 Summary

The paper discussed the application of deep learning to financial modeling by creating a CNN-LSTM hybrid model to project stock market patterns. Another advantage of the model is that it was able to solve some of the core gaps that existed in the previous models, including small data sets, single-model operations, and weak pre-processing procedures. The model was also able to learn both pattern and long dashed dependencies of the financial data as it had parameters that enable the detection of complex patterns and combinations using the convolutional layers and temporal learning using the recurrent layers. Data preprocessing procedures and multi-market datasets also helped the robustness and generalization, and a wider solution evaluation framework would allow more application of the results to the real-world decisions. The results indicate that the suggested method achieves greater predictive precision than related traditional models, which indicates the possibility of helping finance analysis become much more authoritative and informed, despite the presence of issues, such as volatility and interpretability.

7.2 Limitation

Despite the success of this study with the proposed CNNILSTM model to predict the stock market, a number of limitations are still available. This performance of the model strongly depends on the quality and size of the dataset, i.e., many of the errors still can be related to noise, missing values, or an unexpected shock in the market. Although the preprocessing procedures are effective in minimizing these effects, they do not manage to omit the occurrence of some unexpected economic or political factors that have significant impacts on the stock values. Moreover, the model has been trained and tested over a small scope of markets which means that such a model could need to be validated in other financial contexts. The next constraint is that deep learning models can be characterized as the black box which makes them hard to interpret to a financial professional that might require a clear reasoning when a prediction is made. Lastly, the paper emphasized more on technical and historical information by excluding the macroeconomics and sentiment factors which give more detailed information.

7.3 Future Work

There are some lines that this work can be extended in the future. Among the significant measures, it would be expanding the range of diverse datasets that include various markets and industry, larger in time duration that would make the prediction model more adept and reduce the possibility of the model failing. Incorporation of other tools like macroeconomic data, investor mood, or real-time news information might as well aid in the inclusion of external parameters that have a potent impact on the stock performance. A second area that looks quite promising is to investigate explainable AI to make the predictions of CNN and LSTM models more transparent and interpretable to the financial analysts and the decision-makers. Further, future works can explore hybrid deep learning networks or attention-based system to improve long term dependency patterns of stock movements. Lastly, practical testing by implementing the model on real trading virtual environments or decision-making tools would give actual confirmation of the model efficacy and what the model can realistically achieve concerning finance.

References

[1] Gulavnai, S., & Patil, R. (2019). Deep learning for image based mango leaf disease detection. International Journal of Recent Technology and Engineering, 8(3S3), 54-56.

[2] Arivazhagan, S., & Ligi, S. V. (2018). Mango leaf diseases identification using convolutional neural network. International Journal of Pure and Applied Mathematics, 120(6), 11067-11079.

[3] Hossain, M. A., Sakib, S., Abdullah, H. M., & Arman, S. E. (2024). Deep learning for mango leaf disease identification: A vision transformer perspective. Heliyon, 10(17).

[4] Prabu, M., & Chelliah, B. J. (2022). Mango leaf disease identification and classification using a CNN architecture optimized by crossover-based levy flight distribution algorithm. Neural Computing and Applications, 34(9), 7311-7324.

[5] Pathak, A. K., Saikia, P., Dutta, S., Sinha, S., & Ghosh, S. (2024). Development of a Robust CNN Model for Mango Leaf Disease Detection and Classification: A Precision Agriculture Approach. ACS Agricultural Science & Technology, 4(8), 806-817.

[6] Rajpoot, V., Dubey, R., Mannepalli, P. K., Kalyani, P., Maheshwari, S., Dixit, A., & Saxena, A. (2022). Mango plant disease detection system using hybrid BBHE and CNN approach. Traitement du Signal, 39(3), 1071-1078.

[7] Mehta, S., Kukreja, V., & Vats, S. (2023, June). Advancing agricultural practices: Federated learning-based cnn for mango leaf disease detection. In 2023 3rd International Conference on Intelligent Technologies (CONIT) (pp. 1-6). IEEE.

[8] Puviarasi, R. (2022). Accuracy improvement in disease identification of mango leaf using CNN algorithm compared with fuzzy algorithm. ECS transactions, 107(1), 11889.

[9] Saleem, R., Shah, J. H., Sharif, M., & Ansari, G. J. (2021). Mango Leaf Disease Identification Using Fully Resolution Convolutional Network. Computers, Materials & Continua, 69(3).

[10] Arya, S., & Singh, R. (2019, September). A Comparative Study of CNN and AlexNet for Detection of Disease in Potato and Mango leaf. In 2019 International conference on issues and challenges in intelligent computing techniques (ICICT) (Vol. 1, pp. 1-6). IEEE.

[11] Thaseentaj, S., & Ilango, S. S. (2023). Deep Convolutional Neural Networks for South Indian Mango Leaf Disease Detection and Classification. Computers, Materials & Continua, 77(3).

[12] Singh, P. P., Kumar, D., Srivastava, A., Basumatary, M., & Prasad, S. (2023, August). A CNN model based approach for disease detection in mango plant leaves. In International conference on soft computing for problem-solving (pp. 389-399). Singapore: Springer Nature Singapore.

[13] Rajbongshi, A., Khan, T., Pramanik, M. M. R. A., Tanvir, S. M., & Siddiquee, N. R. C. (2021). Recognition of mango leaf disease using convolutional neural network models: a transfer learning approach. Indonesian Journal of Electrical Engineering and Computer Science, 23(3), 1681-1688.

[14] Sharma, A., Bijral, R. K., Manhas, J., & Sharma, V. (2022). Mango leaf diseases detection using deep learning. International Journal of Knowledge Based Computer Systems, 10(1), 40-44.

[15] Singh, U. P., Chouhan, S. S., Jain, S., & Jain, S. (2019). Multilayer convolution neural network for the classification of mango leaves infected by anthracnose disease. IEEE access, 7, 43721-43729.

[16] Baresary, D., Saini, A., Sharma, R. K., Bordoloi, D., Sharma, R., & Kukreja, V. (2023, June). MangoSpot: A Hybrid CNN-SVM Model for Multi-Classification of Mango Leaf Spot Disease Based on Seriousness Levels. In 2023 3rd International Conference on Intelligent Technologies (CONIT) (pp. 1-5). IEEE.

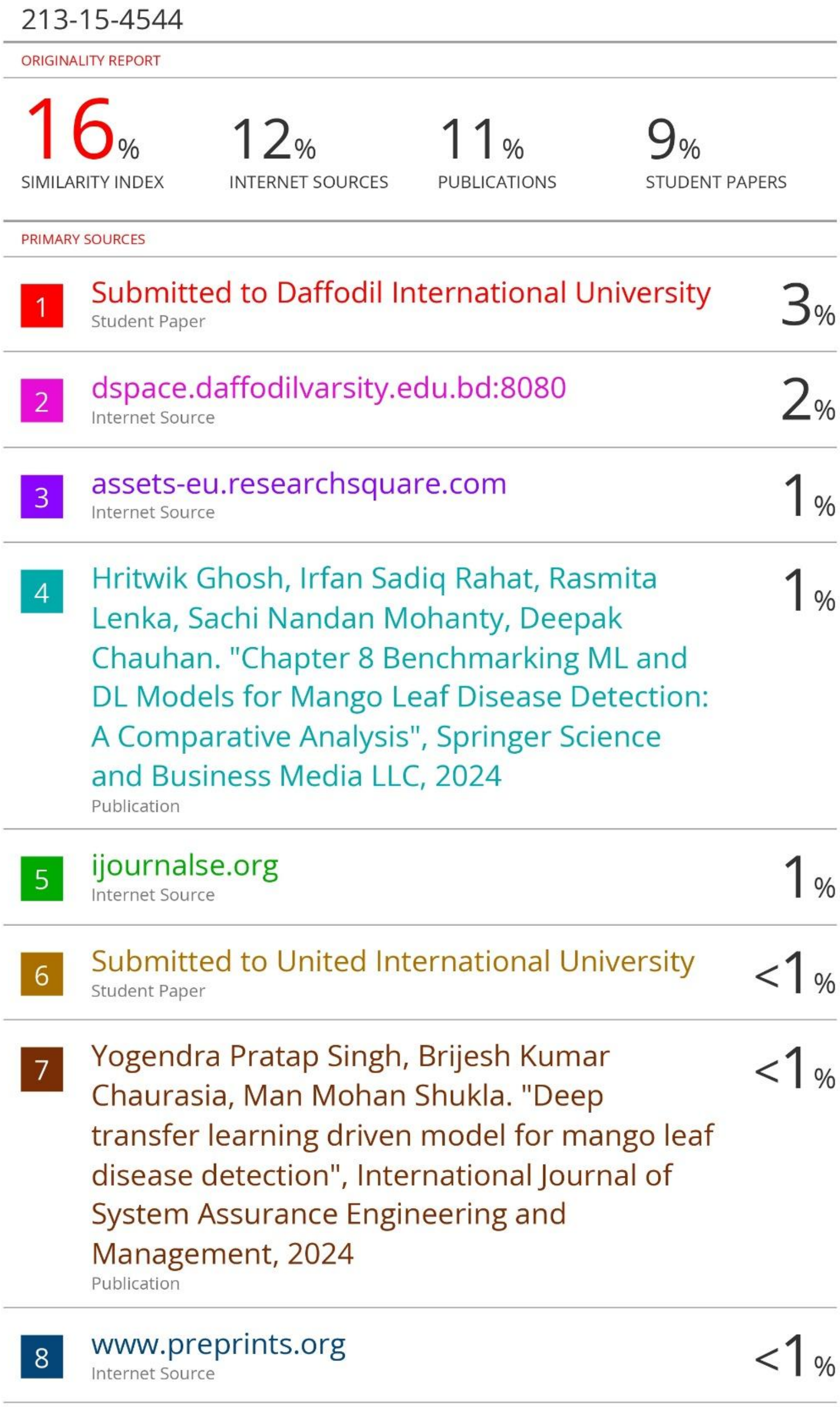
[17] Manoharan, N., Thomas, V. J., & Dhas, D. A. S. (2021, August). Identification of mango leaf disease using deep learning. In 2021 Asian Conference on Innovation in Technology (ASIANCON) (pp. 1-8). IEEE.

[18] Banerjee, D., Kukreja, V., Hariharan, S., & Jain, V. (2023, April). Enhancing mango fruit disease severity assessment with cnn and svm-based classification. In 2023 IEEE 8th international conference for convergence in technology (I2CT) (pp. 1-6). IEEE.

[19] Shaik, T., & Swamykan, S. I. (2023, September). Identification of diseases affecting mango leaves using deep learning models. In International Conference on Artificial Intelligence: Towards Sustainable Intelligence (pp. 132-144). Cham: Springer Nature Switzerland.

Appendix A

Plagiarism Report



213-15-4544

ORIGINALITY REPORT

16%	12%	11%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	3%
2	dspace.daffodilvarsity.edu.bd:8080 Internet Source	2%
3	assets-eu.researchsquare.com Internet Source	1%
4	Hritwik Ghosh, Irfan Sadiq Rahat, Rasmita Lenka, Sachi Nandan Mohanty, Deepak Chauhan. "Chapter 8 Benchmarking ML and DL Models for Mango Leaf Disease Detection: A Comparative Analysis", Springer Science and Business Media LLC, 2024 Publication	1%
5	ijournalse.org Internet Source	1%
6	Submitted to United International University Student Paper	<1%
7	Yogendra Pratap Singh, Brijesh Kumar Chaurasia, Man Mohan Shukla. "Deep transfer learning driven model for mango leaf disease detection", International Journal of System Assurance Engineering and Management, 2024 Publication	<1%
8	www.preprints.org Internet Source	<1%