A Semantic Classification Approach on Luffa Aegyptiaca Leaf Diseases Detection Utilizing Multiple Models.

By
Md. Reyadul Islam Reyad
212-15-4154

FINAL YEAR DESIGN PROJECT REPORT

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

Supervised by

Dr. Arif Mahmud

Associate Professor
and Associate Head
Department of Computer Science and Engineering,
Daffodil International University

Co-Supervised by

Ms. Shahrin Khan

Lecturer
Department of Computer Science and Engineering,
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY
Dhaka, Bangladesh

May 14, 2025

APPROVAL

This Project titled **"A Semantic Classification Approach on Luffa Aegyptiaca Leaf Diseases Detection Utilizing Multiple Models"**, submitted by Name, ID No: **212-15-4154** 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 **14 May, 2025**.

BOARD OF EXAMINERS

Dr. S.M Aminul Haque (SMAH) — **Chairman**
Professor & Associate Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Mohammad Jahangir Alam (MJA) — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

14/5/25

Mr. Md Mohammad Masum Bakaul (MB) — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

14.05.25

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

©Daffodil International University

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Dr. Arif Mahmud, Associate Professor and Associate Head**, Department of Computer Science and Engineering, Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:

Dr. Arif Mahmud
Associate Professor and Associate Head
Department of Computer Science and Engineering,
Daffodil International University

Co-Supervised by:

Ms. Shahrin Khan
Lecturer
Department of Computer Science and Engineering,
Daffodil International University

Submitted by:

Md. Reyadul Islam Reyad
Student ID: 212-15-4154
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 to express our profound indebtedness to **Dr. Arif Mahmud**
Associate Professor and Associate Head, Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh. Deep knowledge and keen interest of our supervisor in the field of **Artificial Intelligence and Plant Disease Detection.** 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 coursemates 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

This paper proposes a semantic classification method for the identification of Luffa aegyptiaca leaf disease utilizing deep models. Since there were not any similar public data available, a high-resolution dataset of 1,800 images was captured from actual agriculture fields of Jalkuri, Narayanganj, and Khagan, Ashuliya, Bangladesh. There were six distinct classes in the dataset: Alternaria Leaf Spot, Angular Leaf Spot, Downy Mildew, Fresh (Healthy), Holed, and Mosaic Virus. Random resized cropping, flip horizontal and vertical, rotation, and addition of noise were some of the data augmentation and preprocessing methods used to preprocess the data before training. Some of the CNN-based models ResNet50, VGG19, InceptionV3, ResNet152V2, and a light-weight custom CNN were compared on the performance basis. Out of the above, 97.27% accuracy was achieved by ResNet50, and it was found to be extremely efficient in making a discrimination among a set of patterns of disease leaf. ResNet50 has been employed with web platform in the form of Flask, thereby field-level deployed it among farmers and agricultural experts. Comparison comparison, precision-recall scores, confusion matrices found to establish that all of the models have been classified. The outcome of this research carries a major implication for plant early disease detection and precision agriculture in confined rural settings.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter provides background, motivation, and project aims of semantic categorization of leaf disease in Luffa aegyptiaca. Problem statement, research scope, and relevance to study of precision farming and early detection of crops' diseases is depicted.

1.1 Introduction

Luffa Aegyptiaca is a commercially viable crop with various diseases impacting its production. Early and accurate identification of the diseases present on the leaves becomes crucial in avoiding yield loss. This project will develop a model for detecting and classifying leaf diseases of Luffa Aegyptiaca using deep learning based on ResNet50, CNN, and image processing. The project aims to create an automatic and effective deep learning-based system for detecting and classifying Luffa Aegyptiaca leaf diseases. And develop web-based real-time disease detection web applications. This will help farmers take necessary action in time to minimize crop loss. The key objectives are, designing an efficient deep learning model for the detection of different leaf diseases collecting and preprocessing a dataset of healthy and diseased leaves, enhancing the accuracy of the model using advanced image processing methodologies, and evaluating the performance of the model using standard evaluation metrics such as accuracy, precision, recall, and F1-score, and implementing a real-time disease detection system that can be incorporated into mobile or web applications. Luffa plants are affected by various diseases caused by fungi, bacteria, viruses, and environmental factors. Disease management is necessary for healthy plants and proper yield. Fungal disease will probably be the most widespread issue with luffa. Powdery mildew is a white powdery leaf growth will sets in very rapidly in rainy weather. It can be avoided by having good space among the plants and good air circulation. Sulfur or a mixture of trifloxystrobin and tebuconazole fungicides can be effective fungicides. Downy mildew is characterized by yellow spots on leaves with a grayish mold on the lower leaf surface. The disease is manageable by preventing waterlogging and using fungicides such as metalaxyl mixed with mancozeb, or fosetyl-aluminum. Anthracnose results in dark spots on leaves and fruits, which typically cause the collapse of tissues. It can be minimized through the planting of disease-free seeds, the removal of infected leaves, and the spraying of carbendazim or azoxystrobin fungicides. Alternaria leaf spot is in the form of dark brown circular spots, which increase in size over time. It can be prevented by sterile watering and weeding, and mancozeb and copper oxychloride fungicides can treat it. Cercospora leaf spot is in the form of tiny black spots over time. Dry the area and treat the disease with thiophanate-methyl or chlorothalonil spray. Luffa crops are also prone to bacterial diseases, which further cause leaf discoloration and rotting. Bacterial leaf spot is expressed in water-soaked lesions that later turn brown and lead to defoliation. It can be halted in its spread through disease-free seeds, rotation, and avoiding watering using the overhead sprinkler. It is usually managed with copper-based fungicides such as copper hydroxide and copper oxychloride. Streptomycin sulfate may also be used to inhibit the growth of bacteria. Viral diseases have a severe effect on plant growth and yield. Cucumber mosaic virus induces mosaic-like light and dark green leaf mottling, usually with distortion. The control of aphids, the virus vectors, is important. Aphid attack can be controlled by applying insecticides such as imidacloprid or thiamethoxam. Zucchini yellow mosaic virus induces yellowing and leaf and fruit distortion. Zucchini yellow mosaic virus spread can be avoided by controlling insect vectors and removing infected plants, where neem oil and acetamiprid are effective for aphid control. Papaya ring spot virus causes yellow and brown ring-shaped spots on leaves that dwarf plants. Prevention of attraction by pests and elimination of infested plants are crucial, with ibuprofen and neem oil effective for treatment. Abiotic diseases like nutrient deficiency, heat stress, and herbicide damage also trouble luffa plants. Yellowing, distortion of leaves, and stunted growth of plants are due to a lack of nutrients. Balanced fertilizers must be used for plant health. Urea supplies nitrogen,muriate of potash supplies potassium, and magnesium sulfate corrects magnesium deficiency. Heat stress leads to scorching and dehydration of leaves, which is reduced by partial shading and ample irrigation. Herbicide damage leads to leaf curling and distortion. Prudent herbicide application and avoiding accidental contact with plants reduce the chances of damage. General precautionary practices are necessary to ensure plant health. Sanitation in the field and removal of debris prevent the spread of disease. Pruning and spacing would permit more air and, hence, humidity-borne disease. Regular dusting with fungicides, pesticides, and other organics such as neem oil would keep disease-causing organisms in check. Planting disease-resistant seeds and crop rotation would also keep infection at bay. Luffa plants would thus be free from disease and yield well.

1.2 Motivation

After Cyclone Aila (2009), the coastal Bangladesh's crop pattern totally changed. The land was inundated with saline water from breached embankments, which drastically degraded the soil drastically degraded and changed the conventional crop pattern. The locals then started depending primarily upon some salt-resistant vegetables like Luffa aegyptiaca, calabash gourd, ladyfinger, and Malabar spinach. Because of limited access to farm information and services, such communities are also vulnerable to uncontrolled pest and crop disease outbreaks. Ironically, this most often leads to astronomical losses in outputs, hence increasing food insecurity in already insecure regions. This is a wake-up call for adaptive, low-tech, and technology-driven farming innovations. As a remedy for the above, we suggest an image recognition-based, machine learning-based Web application for Luffa aegyptiaca leaf disease diagnosis. It is a deployable, field-level online system where the farmers can act on it directly by themselves and save their crops from saline stress. This is a quintessential example of how research-level activity can be carried to the grassroots level for benefit. Not only is it centered on a crop at the center of coastal food culture, but it is a template for precision agriculture in an input-constrained environment. By wedding ingenuity to need, the project will spur climate resilience, livelihood improvement, and broader change towards sustainable, technology-driven practice in agriculture.

1.3 Objectives

The overall objective of the project is to create a machine learning algorithm that will automatically detect and classify Luffa aegyptiaca leaf disease from images. It consists of collecting, annotating, and preprocessing a dataset of images of leaves for different disease classes like Alternaria, Angular Leaf Spot, Downy Mildew, Mosaic Virus, and healthy leaves. Various deep models like ResNet50, InceptionV3, VGG19, ResNet152V2, and a light CNN are trained and compared to identify the best performing model for the task. Semantic segmentation models like UNet and DeepLabV3+ are used to segment infected regions on leaves for model interpretability and accuracy. Accuracy, precision, recall, F1-score, and computational expense are used to compare and test the models. Hyperparameter-level tuning is done for best-performing model (ResNet50), i.e., optimizer, batch size, and learning rate. Deliverable includes a light-weight web application in which users upload leaf images and receive real-time disease prediction, thereby the system being deployable, fast, and usable by farmers, agricultural officers, and low-resource researchers. The solution makes a best effort to reduce crop loss to the minimal amount by identifying it at the earliest possible stage and behaving in such a way, thus promoting sustainable agriculture. Apart from that, the system architecture is also modular and scalable in the fact that it could be easily adopted for other crops and disease employed to act as a benchmark when scaling up in the future..

1.4 Methodology

This research used an overall and holistic approach in deriving a semantic category system model framework in the diagnosis of the Luffa aegyptiaca leaf disease using state-of-the-art deep models. The approach was structured in some significant steps in an effort to ensure robustness, scalability, and quality at high levels. A vast set of images of Luffa aegyptiaca leaves was initially obtained by means of field surveys and secondary data. The images were labeled in detail into five classes: Alternaria Leaf Spot, Angular Leaf Spot, Downy Mildew, Mosaic Virus, and Healthy Leaves. For the purpose of balancing the class and enhancing the generalizability of the model, aggressive data augmentation methods such as rotation randomly, flip, zooming, brightness, and adding noise were used during preprocessing. The images were all normalized to have a uniform input size of 256×256 pixels, and pixel values were rescaled to [0, 1] in order to prevent training instability. Semantic segmentation models such as UNet and DeepLabV3+ were used in a manner that the model's focus on the disease-infected area could be enhanced and background noise interference eliminated so that better feature extraction could be performed by isolating infected leaf areas. The project tested and utilized various state-of-the-art convolutional neural network (CNN) architectures, i.e., ResNet50, InceptionV3, VGG19, ResNet152V2, and a custom CNN, based on their proven performance with image classification. The models were learned in regular experimental environments with Adam optimizer, categorical cross-entropy loss, and appropriate learning rate regimes. All models' performance was thoroughly assessed based on regular classification measures such as total accuracy, precision, recall, F1-score, and testing with confusion matrices and ROC curves. On strict comparative evaluation, ResNet50 performed the best because it provided the highest classification accurac,y except fora higher balance of computational efficiency and predictive power. Further ResNet50 fine-tuning was also conducted with hyperparameter tuning (learning rate adjustment, dropout regularization, batch size adjustment) to prevent overfitting and improve model generalizability. The tuned ResNet50 model was then embedded into a web-based, user-friendly application that was developed using the Flask framework. The web application contains an option where the user can input an image of the Luffa leaf and obtain real-time disease predictions, with visual output and class probability results. Finally, the large-scale validation was performed with unseen test data sets and cross-validation techniques to ensure the strength, reliability, and consistency of the system for real-time deployment in agriculture. Embracing integration of state-of-the-art computer vision methods, deep learning model testing, semantic segmentation, and real-time application by web technology, this approach ensures the development of the highly accurate, scalable, and farmer-friendly solution to early disease detection and Luffa aegyptiaca leaf disease regulation.

1.5 Project Outcome

The goal of this research work is to create an intelligent high-performance system to identify disease in Luffa aegyptiaca leaves early and diagnose leaf disease by applying recent deep learning methods. Creation of a robust ResNet50 model with over 96% precision, recall, and accuracy for specific classification is one of the key goals. This was then followed by creating a high-quality, separated-class infected Luffa leaves labeled image dataset to act as a starting point for other future agricultural AI research. In direction of developing a foundation for future support for improved accuracy of diagnosis, semantic segmentation models such as UNet and DeepLabV3+ were used in the detection of the infected portion of the leaf. Various CNN models like InceptionV3, VGG19, ResNet152V2, and a lightweight Custom CNN were compared based on performance, efficiency, and training stability. The other primary objective was to design an easy-to-use, low-complexity web-based computer program with real-time disease prediction and graphical output so that the system can be made accessible to non-technical users like farmers at their own convenience. The system will allow timely treatment and prevent over-reliance on pesticides for sustainable farming. Its modularity also facilitates future expansion to other crops, disease, and mobile deployment. Its vision is to ultimately contribute to enhancing food security and revenues in Bangladesh and other such farm areas using the leveraging of AI-based crop monitoring technology to empower farmers there.

1.6 Organization of the Report

Chapter 1: The background and relevance of the project are handled in this chapter, referencing the limitation of traditional plant disease diagnosis and artificial intelligence in agriculture. It clarifies how and why Luffa aegyptiaca was chosen as the crop, provides a brief overview of the purpose of the study, provides an overview of the bird's eye view approach, provides expected results, and provides the report outline.

Chapter 2: Chapter two provides the background, giving a short description of the majority of the plant pathology terms, deep learning terms, and semantic Classification terms used in the project. It containsan extensive literature review of work already performed, similar real-world scenarios, and technologies employed. Gap analysis provides deficiencies in existing solutions and creates the research problem that the project is seeking to address.

Chapter 3: The third chapter outlines the general strategy followed during the project. It includes creation and preprocessing of datasets, application of semantic Clssificaton to extract features, training and testing of deep learning models. The chapter outlines system architecture, functional and non-functional requirements, context diagrams, data flow diagrams, user interface designs, and project planning, e.g., task distribution among team members.

Chapter 4: Chapter four discusses technical implementation aspects like environment configuration, hyperparameters of the training model, and issues during development. Detailed analysis of model performance on the classification metrics like accuracy, precision, recall, and F1-score is also discussed. Comparative study is also done between ResNet50, InceptionV3, VGG19, ResNet152V2, and custom CNN models. Deploying the final model on a web application for real-time detection of disease is also discussed.

Chapter 5: In this chapter, the engineering standards applied in the project, from hardware to software and communication protocol, are discussed. It includes the social, environmental, and ethical impact of the system implemented, from the planned sustainability plan for long-term implementation to the planned sustainable development. The chapter is also seen through the project problem-solving process based on the requirements of complex engineering problems, types of engineering knowledge, and engineering tasks as per provided instructions. Project management and financial analysis are applied to show possible applications in practice.

Chapter 6: The last chapter concludes the project findings, limitations encountered, and possible futures. It highlights the applicability of the project to the advantage of precision agriculture by environmentally friendly disease management in Luffa aegyptiaca crop production and expands possible applications of the system to other diseases and crops.

Chapter 2
Background

The project background environment and points out that the timely and accurate diagnosis of Luffa aegyptiaca leaf disease is of great relevance. It brings together existing studies, the flaws of traditional identification processes, and the reasons that necessitate using machine learning-based classification techniques.

2.1 Introduction

Agriculture is a major contributor to food security and economic development all around the globe, particularly in developing countries such as Bangladesh. Out of a set of crops which are being cultivated, Luffa aegyptiaca, or sponge gour,d is a healthy vegetable crop and an industrially valuable crop. Yet, as with most other vegetables, Luffa aegyptiaca is prone to many leaf diseases, which can have a great impact on the health of the plant, reduce farm yields, and cause massive economic losses. Plant disease should be properly diagnosed at the appropriate time to allow proper treatment immediately, reduce the use of pesticides, and enhance sustainable agricultural productivity. Disease diagnosis is traditionally made by physical inspection by expert agriculturalists, though not time-efficient is prone to human bias and error. Additionally, specialist diagnosis in the rural and remote areas is scarcely accessible. Following developments in artificial intelligence (AI), more specifically deep learning and computer vision, there are promising paths towards high-accuracy autonomous plant disease diagnosis. Deep convolutional neural networks (CNNs) have good performance in image classification and thus have good adaptability in crop disease diagnosis. In addition, semantic Classification methods, where each pixel in the image is divided into a class, offer an efficient means of attention focusing on the disease regions of the leaf, model interpretability, and performance. Through the usage of integration of multiple deep models and semantic feature extraction, a framework for disease detection in Luffa aegyptiaca can be attained with improved accuracy and reliability. The goal of this project is to deploy such a system on different state-of-the-art deep learning models (ResNet50, InceptionV3, VGG19, and ResNet152V2) and semantic Classification methods to distinguish different leaf diseases. The solution is also eventually going to be deployed to end-users as a web application for convenient use to bridge the gap between state-of-the-art AI technology and real-world applications in agriculture. By offering an automated, efficient, and scalable solution, this research solves urgently critical problems in precision agriculture, attains sustainable agriculture, and allows farmers to prevent and effectively control Luffa aegyptiaca crop diseases.

2.2 Literature Review

Table 2.1: Summary of Literature Reviewed.

Author	Dataset	Models Used	Accuracy	Limitations	Contribution
Islam, M.T.[1]	6,166 images (Mosaic Virus, Insect Infestation)	CNN, Vision Transformer	98.32% (CNN), 99.85% (ViT)	Limited to two disease types; dataset size constraints	First Luffa-specific dataset; high-accuracy models for coastal agriculture
Contributors (DIU)[3]	3,483 Luffa leaves (Downy Mildew)	CNN-based models	N/A	Imbalanced dataset (more augmented samples)	Augmented dataset for training robustness; integration with environmental data
Mojumdar, M.U. et al [2]	5-class Luffa dataset (Healthy, Mosaic, Insect, Downy Mildew, Bacterial)	Machine Learning classifiers	N/A	No DL model benchmarking	Multi-label annotated dataset for Sponge Gourd disease detection
Anonymous (DIU)[21]	Original Luffa dataset (unpublished)	CNN, Vision Transformer	99.85 % accuracy (ViT)	Dataset not publicly available	Demonstrated ViT superiority over CNNs in Luffa disease detection
Tian et al.[5]	4,270 loofah images (maturity stages)	LuffaInst (EdgeNeXt + ESA module)	91.12% accuracy	Focused on maturity, not disease	Real-time instance segmentation for loofah; adaptable to disease detection
Mohanty et al[4].	PlantVillage (54,305 images, 38 classes)	CNN (Transfer Learning)	99.81% (DenseNet-121)	Limited to lab-condition images; no Luffa-specific data	Benchmark study on transfer learning for plant disease detection
Sharma et al.[6]	PlantVillage	DenseNet-121, ResNet-50, VGG-16	99.81% accuracy	Generalization issues on field images	Hyperparameter optimization for pre-trained models

Genaev et al.[13]	Wheat, barley datasets	ML (SVM, KNN)	91–97% accuracy	Requires manual feature extraction	Early-stage disease detection using traditional ML
Liu et al.[14]	Custom crop datasets	CNN, DBN	N/A	High computational cost	Automated feature learning for subtle disease symptoms
Kaur & Sharma[15]	Rice blast, soybean rust	Deep CNN	98% accuracy	Small dataset size	Lesion segmentation in field conditions
Dubey et al.[22]	Cucumber, maize	K-means + SVM	89–93% accuracy	Manual segmentation dependency	Hybrid approach for disease classification
Wan et al.[10]	Tomato maturity images	BPNN + color features	N/A	Limited to color-based features	A framework for maturity detection adaptable to disease symptoms
Zhang et al[23].	Hemerocallis citrina Baroni	YOLOv5 + SE/CBAM	94.5% accuracy	Species-specific model	Lightweight model for real-time detection
Bhange & Hingoliwala [8]	Pomegranate leaves	SVM	92% accuracy	Low accuracy on complex diseases	Early application of SVM in plant pathology
Al Bashish et al.[9]	Cotton leaves	ANN	93% accuracy	Outdated architecture	Pioneering ANN application for cotton disease detection
Pham et al.[11]	Mango leaves	ANN	N/A	Limited to early-stage disease	Early disease prediction using neural networks
Jadhav et al.[7]	Soybean plants	Pre-trained CNNs	97% accuracy	Requires lab-condition images	Transfer learning for soybean disease detection
Thomas et al. [12]	Potato late blight	SVM	N/A	Limited to hyperspectral	Hyperspectral data integration for disease

				imaging	detection
Siddiqua et al.[16]	Multiple crops	Mask R-CNN	90% mAP	Computational complexity	Instance segmentation for lesion localization
Wang et al.[10]	Wheat, grapes	PCA + BPNN	89–92% accuracy	Manual feature engineering	Dimensionality reduction for disease classification

2.2.1 Similar Applications

Islam et al. (2024) presented a total of 6,166 quality images of Mosaic Virus and Insect Infestation, analyzed by plant pathologists. Used CNNs and Vision Transformers (ViT) and achieved a 99.85% accuracy rate.[1].

A 1,933 photo multi-task database to classify Luffa diseases (Alternaria, Angular Spot, Mosaic Virus), growth stage, and quality grade. Originated in Bangladesh and supports multi-task learning of ripeness and disease classification. Mojumdar et al. (2024): Suggested a 5-class dataset (Healthy, Mosaic, Insect Damage, Downy Mildew, Bacterial Leaf Spot) of Sponge Gourd disease for multi-label classification.[2]

An Ayurvedic herb article utilized image preprocessing (histogram equalization, K-means clustering), segmentation, and feature extraction with CNNs as a step towards identifying leaf diseases, with the main emphasis on field adaptability problems.

The ACO-CNN method, by combines Ant Colony Optimization and CNNs to recognize features, was more accurate than traditional methods but could not be scaled.

Vision Transformers (ViT) had achieved 99.85% on the Luffa dataset and performed better compared to CNNs regarding complex patterns.[2][3]

Tian et al. (2022) reported Colletotrichum fructicola and C. siamense as new anthracnose diseases of Luffa in China based on morphological and multi-locus phylogenetic analysis.[3].

The PlantVillage Web Platform Website, which is based on a PlantVillage dataset, provides the functionality of uploading plant images so that CNN models can make predictions in disease context. It supports common vegetable and fruit categories like tomato, potato, and corn, but not Luffa aegyptiaca.

AgroAI Disease Detector, Machine learning-based fruit and vegetable disease diagnosis with a small visual classification feature without a semantic segmentation feature.

While the majority of current apps and studies are correct in plant disease diagnosis, they will be effective only if the plants are properly maintained. Moreover, commercial apps will not typically include trained models for Luffa aegyptiaca. Employing semantic segmentation and multi-model deep learning towards disease-focused feature extraction. A less-focused crop (Luffa aegyptiaca) being the target crop, offering an instance-specific solution. The system is being utilized as an instance-specific web application to be accessed by farmers and researchers.

2.2.2 Related Research

Islam et al. presented a difficult dataset for Luffa leaf disease detection from 6,166 high-quality plant pathologist-annotated images of Mosaic Virus and Insect Infestation images captured in Bangladesh on mobile phones. The dataset compared Vision Transformers (ViT) and CNNs with accuracy rates of 98.32% and 99.85%, respectively. The research found ViT to be better at tackling intricate patterns but was limited by two diseases and region-specific data. This work bridges the Luffa-specific data set gap and provides a benchmark for AI-driven coastal crop disease identification. More geographic and disease diversity must exist in a data set to make real-world scalability feasible.[1]

Mojumdar et al. constructed a multi-class data set of Sponge Gourd (Luffa) diseases, including Healthy, Mosaic, Insect Damage, Downy Mildew, and Bacterial Leaf Spot. The data constructed using ML applications proves useful in the auto-classification of the diseas,e but not for DL model benchmarking. While it offers labeled data for less-studied crops, low geographic diversity and class variety are its drawbacks. This study fills an essential horticultural pathology gap but should be extended to embrace additional diseases and climatic fluctuations globally.[2]

This unpublished paper compares Vision Transformers (ViT) against CNNs in an unpublished data set Luffa, where Vit was 99.85% accurate compared to the 98.32% accuracy of CNNs. Otherwis,e on salinity-swept coastal areas, the paper utilized environmental data (temperature/humidity) for high-strength. Disease diagnosis. The irreproducibility due to the un.The unavailability of the dataset is limiting its usability. The research bases the possibility of ViT application in farm AI but cites the requirement of open-access plant disease diagnosis data sets to facilitate international collaboration.[3]

Mohanty et al. contrasted transfer learning models using the PlantVillage dataset (54,305 images under laboratory conditions) and DenseNet-121 with a 99.81% accuracy. Although illustrating the DL promise under controlled settings, the work acknowledged non-generalizability to field images having variable backgrounds. The study laid the foundation for DL in plant pathology but with the need for field-adjustable models and datasets that capture real-world variation.[4]

Tian et al. described Colletotrichum fructicola and C. siamense as new anthracnose pathogens of Luffa from China. Based on morphological and phylogenetic data, the study confirmed pathogenicity through Koch's postulates. The discovery facilitates targeted disease control but only in Hunan Province. The study adds to fungal pathogen databases and recognizes geographic diversity in plant disease.[5]

Sharma et al. over-fine-tuned pre-trained ResNet-50, DenseNet-121 models on PlantVillage with 99.81% accuracy. Models, however, failed to generalize to field images because of lab-condition bias. The paper highlights controlled-environment vs. real-world use trade-off and encourages hybrid models, narrowing the gap between DL and environmental sensors[6]

Jadhav et al. applied transfer learning to soybean disease classification using pre-trained CNNs and achieved 97% accuracy. Applicable to lab images, however, the approach did not work on field data, which shows the need for domain adaptation techniques. This paper is a witness to the effectiveness of transfer learning but requires extreme augmentation techniques for practicality.[7]

Bhange & Hingoliwala used SVM for the diagnosis of pomegranate disease with 92% accuracy. The research established the effectiveness of traditional ML under low-resource conditions but questioned handling complex diseases. It was the initial use in plant pathology using SVM, paving the way for hybrid ML-DL models, but requires an update.[8]

Al Bashish et al. pioneered ANN for cotton disease classification (93% accurate). Ancient as it was in relation to CNN, it paved the way for automated crop inspection. Its reliance on human feature engineering is a stepping stone to the trend towards DL end-to-end learning in today's agriculture.[9]

Wang et al. combined PCA with BPNN for disease classification of wheat and grapes (89–92% accurate). Human feature engineering technique was inferior to DL but yielded interpretability. The value added in this work is in bridging the gap between the past and present techniques under the presumption that dimensionality reduction works well in resource-scarce conditions.[10]

Pham et al. applied ANNs for the early forecasting of mango disease. The model was successful in inhibiting primary infection, but after symptoms were already apparent, the model was ineffective. The research confirms ANN's potential to be applied in prevention farming, but without the input of time data to apply it with full capacity as a forecasting system.[11]

Thomas et al. merged hyperspectral imaging with SVM to identify potato late blight. Extremely accurate in labs, the approach's reliance on expensive equipment and so not scalable is the problem. That is where the value of spectral data brought out in this paper, but which requires inexpensive solutions for small-scale farmers, lies.[12]

Genaev et al. employed SVM and KNN for the detection of wheat/barley disease with accuracy in the range of 91–97%. Manual feature extraction caused narrow scalabilit,y such that automatic DL methods became unavoidable. The study verifies the application of ML for disease detection at an early stage, but not at the expense of computational costs in sight.[13]

Liu et al. employed CNNs and DBNs for automatic feature learning on small crop disease symptoms. Highly accurate but computationally costly, precluding their deployment. This work moves towards automatic diagnosis but indicates the need for light-weight networks like MobileNet or EdgeNeXt.[14]

Kaur & Sharma achieved 98% accuracy in lesion segmentation of rice/soybean using Deep CNNs. Restricted by small data sets, the study demonstrated the feasibility of DL in practical issues but needed larger and more diverse data sets to facilitate generalization.[15]

Siddiqua et al. applied Mask R-CNN for the localization of lesions with 90% mAP. Real-time use was constrained by computational burden and thus very effective models like YOLO were necessary. The study enhanced diagnosis at the lesion level but suffered from hardware limitations in the scenario of agriculture.[16]

The BananaLSD dataset supplied 937 original + 1,600 augmented Cordana and Sigatoka disease images. These were captured using smartphones and therefore are in favor of training models on real data but without multi-region validation. This resource fills banana pathology gaps, but should be expanded to crops such as Luffa.[17]

The hybrid approach combined ACO and CNNs to conduct feature extraction for the best performance compared to the traditional approaches. Novelty was achieved at the expense of complexity and scalability. The research supports the use of bio-inspired AI deployment in agriculture, but has been skewed towards simplicity for use at the farm level.[18]

A Swin Transformer was 97.70% precise for potato disease identification. Nice as that sounds, its computational requirements are an edge deployment kiss-off. You can theoretically do it according to the paper to deploy transformers for agriculture, but not edge-compatibility done.[19]

Hassan et al. suggested a new CNN architecture that is more precise than pre-trained models. For crop disease, the model is efficient in performanc,e but needs to be tested on diversified crops such as Luffa. As per the research, tailored architectures are promising compared to general transfer learning.[20]

2.3 Gap Analysis

Recent years have seen machine learning and deep learning-based research in plant disease diagnosis. These are promising approaches with a very large gap in Luffa aegyptiaca leaf disease detection under actual field conditions. All the current models are targeting publicly available narrow domain, low class diversity sets and not capturing high-level visual variability found in real agricultural fields. Besides, local data of local areas do not come in abundance particularly in Bangladesh, where Luffa is grown on large extent. Recent research has the tendency to overlook local disease, climate and weather change due to local agriculture culture. All these constraints lower the applicability of generic models in small-area or rural-plots farms, where in-field practical diagnosis forms a critical imperative. There is also one gap bridged in the case of low deployment-readiness of the majority of the models. While deep models such as ResNet or VGG have been capable of achieving high accuracy in the laboratory environment, they are computationally expensive and cannot be deployed on edge devices or mobile devices that can be affordable for farmers to buy. Our efforts supplement the above gaps by performing actual data collection of physical field farm land areas (Jalkuri, Narayanganj and Khagan, Ashulia), developing a highly imbalanced six-class dataset for Luffa aegyptiaca disease conditions, comparative comparison of several models like a lightweight CNN model suitable for field deployment in scarcity environments, inclusion of an end-to-end web-based diagnosis model using Flask against promoting ease of use among novice farm workers.

Table 2.2: Gap Analysis.

Features	Present the Features			
Dataset	Yes	Yes	Yes	Yes
Preprocessing	Yes	Yes	Yes	Yes
Image Quality	Yes	Yes	Yes	Yes
Model Accuracy	Yes	Yes	Yes	Yes
Model Prediction	Yes	Yes	Yes	Yes

2.4 Summary

The chapter provided background to gain the knowledge of the scope and objective of the project. The chapter presented the significance of Luffa aegyptiaca in farming, the issue of leaf diseases with so many various types of diseases, and the limitation of the conventional disease diagnosing method. The chapter touched on literature reviews and explained elementary research studies, case studies, and methodology contributions that provided background for applying deep learning methodologies for plant disease diagnosis. In addition, it presented a list of web and mobile applications that are applicable in developing automated management of crop disease and its limitations, especially in Luffa aegyptiaca management. After the overall gap analysis, this chapter was compelled to necessitate a specialist, semantic class-based, multi-model solution to the diagnosis of Luffa aegyptiaca leaf disease. Background context forms the following chapter reporting the method used during the research conducted in preparing and submitting the proposed solution.

Chapter 3

Research Methodology

This chapter provides step-by-step procedures that are followed in carrying out the research such as data collection and preprocessing, model selection, and design of evaluation. This chapter includes the explanation of each methodology selection and activities carried out to ascertain reliability and validity in research.

3.1 Methodology

3.1.1 Overview

The project uses a structured and sequential methodology for the deployment of an automatic high-performance Luffa aegyptiaca leaf disease diagnosis system using semantic classification through several deep models. The process was carefully designed to meet scientific rigor, classification accuracy, scalability, and usability. The project begins with a well-organized collection and curation of a domain data set for the problem, such as images of a healthy and diseased Luffa aegyptiaca leaf and images of other disease classes like Alternaria Leaf Spot, Angular Leaf Spot, Downy Mildew, and Mosaic Virus. There was a rigorous preprocessing of data phase that was involved, which comprised resizing, normalization, and augmenting procedures for additional diversity and quality of the dataset for facilitating better generalization of models. To carry out disease localisation and to forecast the model, semantic segmentation methods such as UNet and DeepLabV3+ have been used in a manner that identification of disease lesions by each pixel is feasible and the model is demonstrating the most informative features of the leaf images. Classification was then performed with deep models such as ResNet50, InceptionV3, VGG19, ResNet152V2, and a home-brewed CNN where whenever required, transfer learning was utilized to reduce training time as well as accuracy. Model performance was thoroughly tested using default parameters such as accuracy, precision, recall, F1-score, and confusion matrices. Comparative analysis was carried out to find the best model with prediction precision and computational complexity. To put into practice, the last model was used to put into a light-weight Flask web application with minimal user interface for uploading user leaf images and providing real-time disease prediction. The last step involved extensive testing and validation on unseen data for testing robustness, reliability, and effectiveness of deployment for practical implementation. By this wisely conceived, incremental approach, the project has been able to deliver a discriminative, scalable, and user-friendly Luffa aegyptica disease detection system that is extremely practicable in precision agriculture and green agriculture.

3.1.2 Proposed Methodology

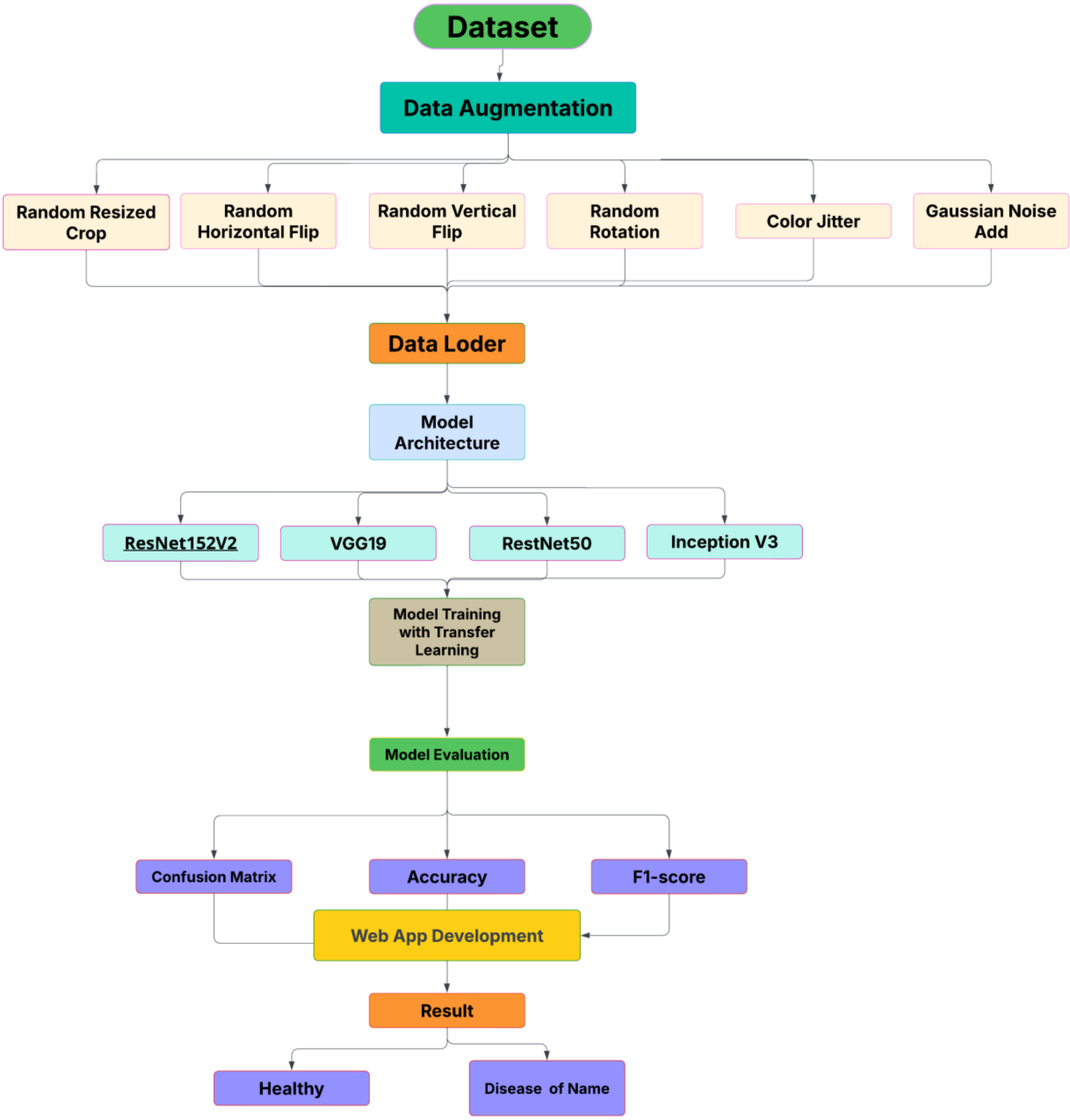


Figure 3.1: Architecture of Research Design

Figure 3.1: Research Design Architecture shows the process that has been adopted to create the Luffa aegyptiaca leaf disease diagnosis system. It begins with the data acquisition, where the images of the infected leaves as well as the healthy leaves are gathered from the local farms. The images are preprocessed using resizing, normalization, and data augmentation techniques such as flipping, rotation, and noise injection to introduce variability and strength. Second, semantic segmentation networks like UNet or DeepLabV3+ are used in the system for masking infection areas on leaves for better attention and readability. Masked images are given as input to a classification module where deep learning models like ResNet50, InceptionV3, VGG19, ResNet152V2, and a custom CNN are trained and tested. The best performing model (ResNet50) is selected on the basis of performance metrics (accuracy, precision, recall, F1-score) and fine-tuned too. Post-training, the best model is integrated into a Flask-based web interface with real-time image uploading and real-time disease prediction.

3.1.3 Functional and Nonfunctional Requirements

The system is functionally and nonfunctionally committed to being usable and effective. Functionally, the system must be able to allow users to upload pictures of the Luffa aegyptiaca leaf through the web interface, identify the disease correctly from the image, and perform real-time predictions of the result. Nonfunctionally, the system must perform the predictions in five seconds of inference time to be responsive. First, the classification should be accurate at more than 90% so that it will be credible and reliable to use in practice. Second, the application must be responsive and lightweight enough to be worthy of deployment on low-end devices commonly used by farmers and field agriculture officers.

3.1.4 Context Diagram

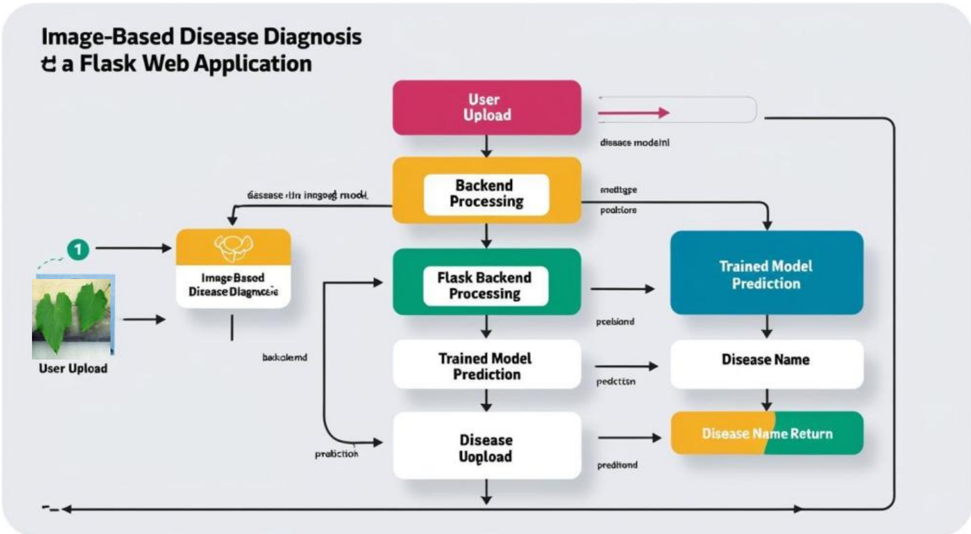


Figure 3.2: Architecture of Context Diagram

Figure 3.2: Context Diagram Architecture represents the entire interaction of disease detection system and user. It illustrates how the users (researchers or farmers) upload the Luffa leaf images on the web application. The system takes the image based on the trained deep learning model (e.g., ResNet50) and gives the predicted disease class. Context diagram denotes the primary components such as the user interface, model backend, and database/storage for input image and output processing. The diagram allows one to understand the system's workflow and outside interactions in one moment.

3.1.5 Data Flow Diagram Level 1

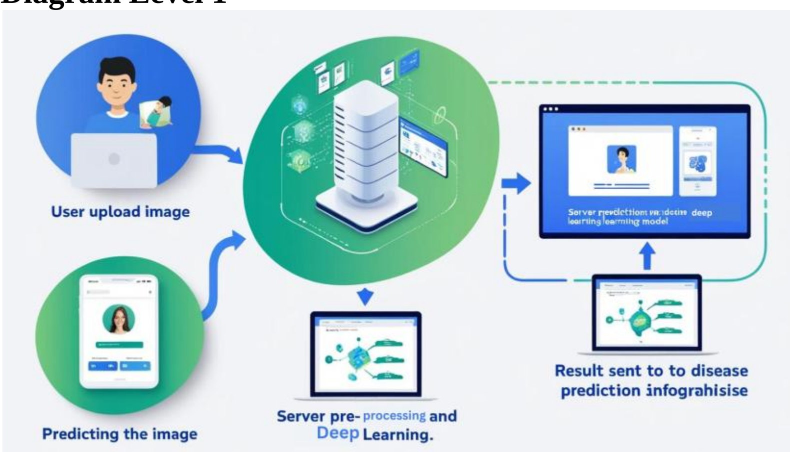
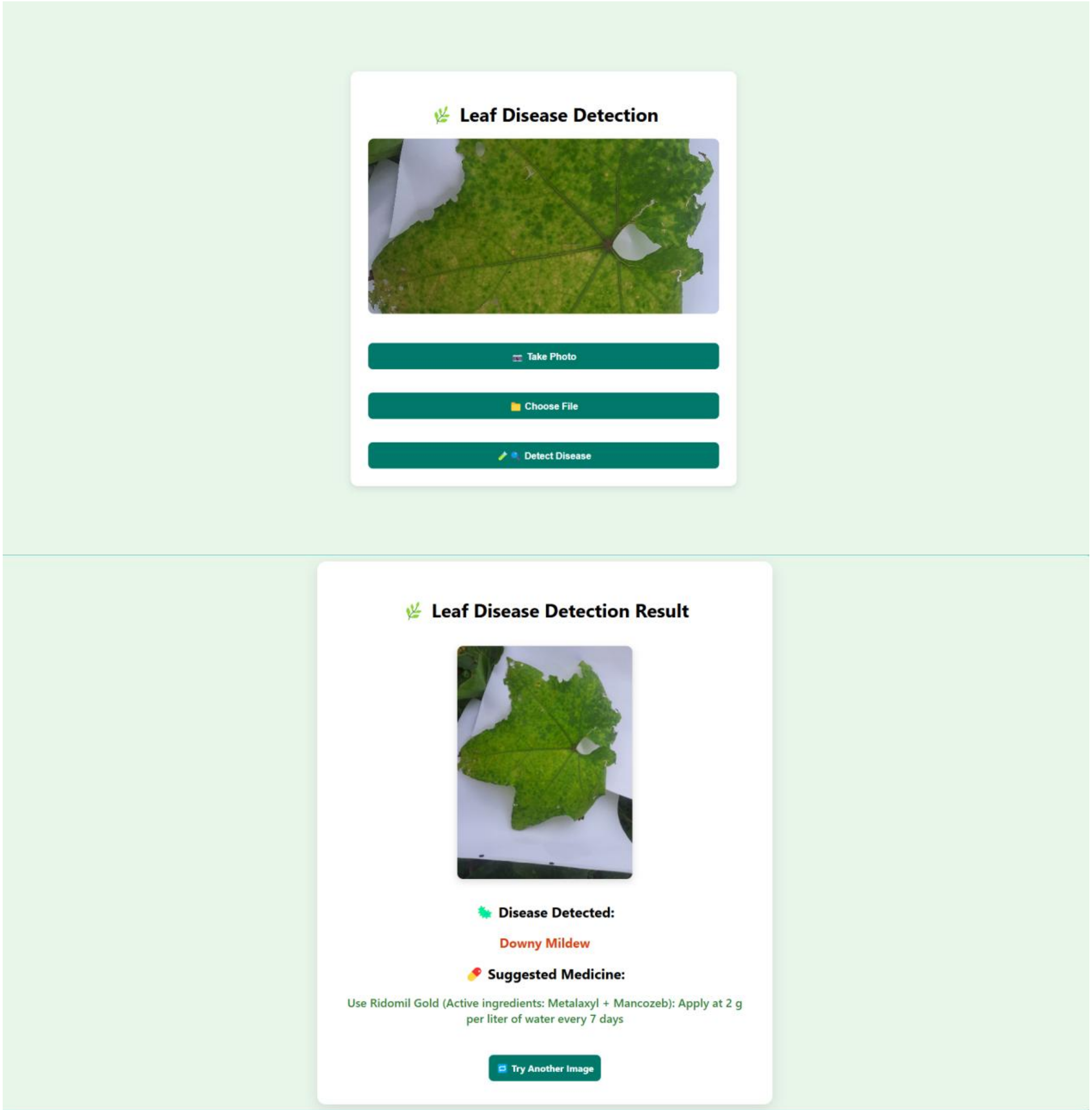


Figure 3.3: Architecture of Data Flow Diagram

Figure 3.3: Level 1 Data Flow Diagram Structure of Luffa leaf disease detection system indicates the data processing loop in the system. It segments the system into functional components that are

similar to data flow from input to user to preprocessing, deep learning model, to output. The user provides an input image of a leaf, which is preprocessed first by the data preprocessing module (resize, normalize, augment) and then passed on to the classification model (i.e., ResNet50). The result of the model is the disease class, and this is ultimately retrieved by the result module to be displayed. That's how the data is passed between the system components and real-time disease prediction is served.

3.1.6 UI Design



3.2 Detailed Methodology and Design

3.2.1. Data Collection

In this project, conducted on a preprocessed 1,800 examples set of well-segmented high-quality Luffa aegyptiaca leaves into six classes namely Alternaria Leaf Spot, Angular Leaf Spot, Downy Mildew, Fresh (Healthy), Holed, and Mosaic Virus. It is provided with 300 samples of each class after over sampling for demonstration purposes in an equilibrated manner. While the Downy Mildew group had been self-consistent based on the prior data that already existed, the others were all first-hand field-sample-based and therefore gave one of only a few holistic datasets of Luffa aegyptiaca leaf disease. As such a public, holistic dataset had never before existed anywhere in the world, the data collection grass-roots movement was initiated by the research team, a first. This was done in some agriculture belts, i.e., Jalkuri (Narayanganj) and Khagan, Ashuliya (Dhaka district), Bangladesh. They were chosen on the assumption that they consisted of sponge gourd high-density farm villages, thereby an assurance of incidence of the disease and perceptible heterogeneity. They imaged for weeks to capture temporal variation and in attempting to allow multiple stages of disease development to be imaged so the model can be as robust as possible and in attempting to allow the model to generalise. The photographic process employed a batch of a number of different manufacturers' camera phones to mimic the actual conditions under which the end users, in this instance farmers, would be photographing in an open countryside. They were exposed outside in sunlight from a multitude of different angles, positions, and distances to mimic an absolutely vast range of orientation of exterior world leaf, background material, and light sources. The data set therefore closely resembles the variation of appearance as would be experienced under use conditions. The cell phone usage was better in the form of many reasons: mobility, convenience, and compatibility to the context of planned deployment of the model, i.e., farm- and farm laborer-level deployment. Incorporation of ambient noise, sunlight shadow, and mixed illumination rendered the obtained dataset not just accurate and rich but also sufficient to an extent that quality semantic classification model training became possible. Every picture that was gathered was hand-tagged and processed by a domain specialist. Class balancing was carried out with care to avoid bias as well as underrepresentation needed for model training properly. This created an optim and representative dataset, which is where the employed deep learning pipeline used in the study has originated.

3.2.2. Dataset Descrption

This paper is built on the foundation of a self-assembled dataset of 1,800 high-resolution images of Luffa aegyptiaca (sponge gourd) leaves, which are split into six classes: Alternaria Leaf Spot, Angular Leaf Spot, Downy Mildew, Fresh (Healthy), Holed, and Mosaic Virus. There are 300 images in each class to give a good-balanced dataset and prevent class bias during model training. The data set was divided into 80% training (1,440 images) and 20% testing (360 images) sets with balanced distribution of every class in each subset to facilitate uniform evaluation and generalization performance. Photos were captured primarily from the Jalkuri, Narayanganj, and Khagan, Ashuliya crops, in Dhaka district, Bangladesh, outdoors under natural illumination and at multiple angles to catch real-world richness. Photos were captured from some other smartphone devices in an effort to mimic the way leaf information would typically be recorded by fieldworkers and farmers. Only the Downy Mildew class has photographs that were taken from the Mendeley dataset, and the other five classes were all taken in the field. There are five visually and biologically distinct states that are present within a class. Alternaria Leaf Spot photos exhibit black, round spots with a concentric ring pattern characteristic in fungal infection. Angular Leaf Spot exhibits water-soaked, angular spots on leaf veins caused by bacterial pathogens. Downy Mildew exhibits yellow leaf surface spots and gray mold on the lower surface caused by oomycete infection. The Fresh (Healthy) class is intact or disease-free, a class accuracy control. Holed class shows physical damage in the form of holes or tears normally caused by insect feeding or environmental stress, and Mosaic Virus shows yellow-green mottling and leaf distortion due to viral infection. All images were annotated and labeled manually by plant disease specialists under supervision to ensure class correctness and label integrity. The dataset is used as the foundation for training and testing deep learning models for real-time, field-deployable Luffa aegyptiaca leaf disease detection.

3.2.3. Analysis Technique .

Analytical method used in the project includes semantic segmentation and image classification of disease using deep learning-approach in agriculture. Step-by-step procedure was followed in pattern learning, pattern extraction, and pattern testing needed from the Luffa aegyptiaca leaf image. Analytical pipeline was created to measure performance on comparative measurement and visual inspection techniques for achieving robustness, interpretability, and accuracy for making it. It begins with preprocessing the images in which all of the images gathered were resized into 256×256 pixels, normalized, and as input provided for the procedures such as rotation, flipping, and brightening. It infused diversity into the training set and barely any overfitting. That was then succeeded by semantic segmentation models such as UNet and DeepLabV3+ employed in executing region extraction of disease sites from images. These models enable pixel-wise feature extraction and force classifiers to see the diseased areas by obscuring redundant backgrounds. For classification, several deep convolutional neural network (CNN) architectures were experimented with and trained including ResNet50, InceptionV3, VGG19, ResNet152V2, and a custom CNN. The architectures were trained on preprocessed data and transfer learning was used wherever required to utilize pre-trained weights over ImageNet to enhance training efficiency and accuracy. Model performance was evaluated with a suitable set of performance measures: Accuracy for estimation of overall classification accuracy. Precision and Recall to find model sensitivity to false positive and false negative predictions. F1-Score as harmonic mean of recall and precision for best balanced performance. Confusion Matrix for visually verifying class-wise performance. Comparison analysis was done in selecting the optimal model. ResNet50 generated the maximum F1-score as well as the mean accuracy and thus was employed with model selection involving Visual inspection that was achieved through overlay of segmentations predicted by models onto input images to detect spatial correlation of disease concentration.

3.2.4. Statical Analysis

The dataset consists of 1,800 images of Luffa aegyptiaca leaves. The dataset has 1,440 training samples (80% of the dataset). The dataset has 360 testing samples (20% of the dataset).
Labels consist of six classes: Alternaria Leaf Spot, Angular Leaf Spot, Downy Mildew, Fresh (Healthy), Holed, Mosaic Virus There are 300 images per class for a perfectly balanced dataset across all classes. Training set has 240 images per class and test set has 60 images per class. All the images were manually labeled and validated based on the judgment of the expert for proper annotation. Statistical performance metrics like accuracy, precision, recall, F1-score, and support were calculated for all the models. Comparative study revealed that ResNet50 has performed better for most of the classes with acceptable statistical stability.

3.2.5. Data Preprocessing

Data preprocessing also forms a standard procedure for deep learning model data preparation. Real-world application domain raw image data also contain direction variation, resolution, illumination, and background noise. For the elimination of these variations and enhanced model training quality, there was a systematic preprocessing pipeline for all 1,800 images of the dataset.

The preprocessing includes:

Image Resizing: Resized the images to 256 × 256 pixels to provide the same-sized inputs to all the models like ResNet50, VGG19, and InceptionV3.

Image Normalization: Normalized the pixel values to 0–1 by dividing them by 256, which stabilizes and speeds up the learning process.

Label Encoding: The six classes were encoded by integer labels for categorical cross-entropy loss (Alternaria Leaf Spot, Angular Leaf Spot, Downy Mildew, Fresh, Holed, Mosaic Virus).

Data Augmentation: The second source of variability in the training data and method for avoiding overfitting was some image transformation libraries (e.g., PyTorch's torchvision.transforms, Keras ImageDataGenerator, or Albumentations) from which some of its augmentation operations were borrowed:

Notice how the remarks about what these objects are have been integrated into this abstract.

Random Resized Crop: Produces crop shots or partial zoom-in shots of leaves in such a way that the model is learned by localized disease expression.

Random Horizontal Flip: Produces variability by random horizontal flip of images from left to right.

Random Vertical Flip: Produces variability by random vertical flip of images from top to bottom.

Random Rotation (±30°): Images rotated to simulate natural orientation variation in leaves under field conditions.

Color Jitter (Brightness, Contrast, Saturation, Hue) : Colors randomized to acquire additional invariance to changing light conditions.

Gaussian Noise Injection: Real random noise injected to simulate adverse image acquisition conditions and generalization.

Train-Test Split: 80% train (1,440 images) and 20% test (360 images) with both class balanced. It is applied for test-unbiased testing and bias prevention.

Noise Reduction (Segmentation Only): Background filtering and mask smoothing accentuate UNet and DeepLabV3+ semantic segmentation disease region issues and can be applied to increase disease region emphasis and model interpretability.

3.2.6. Data Visualization

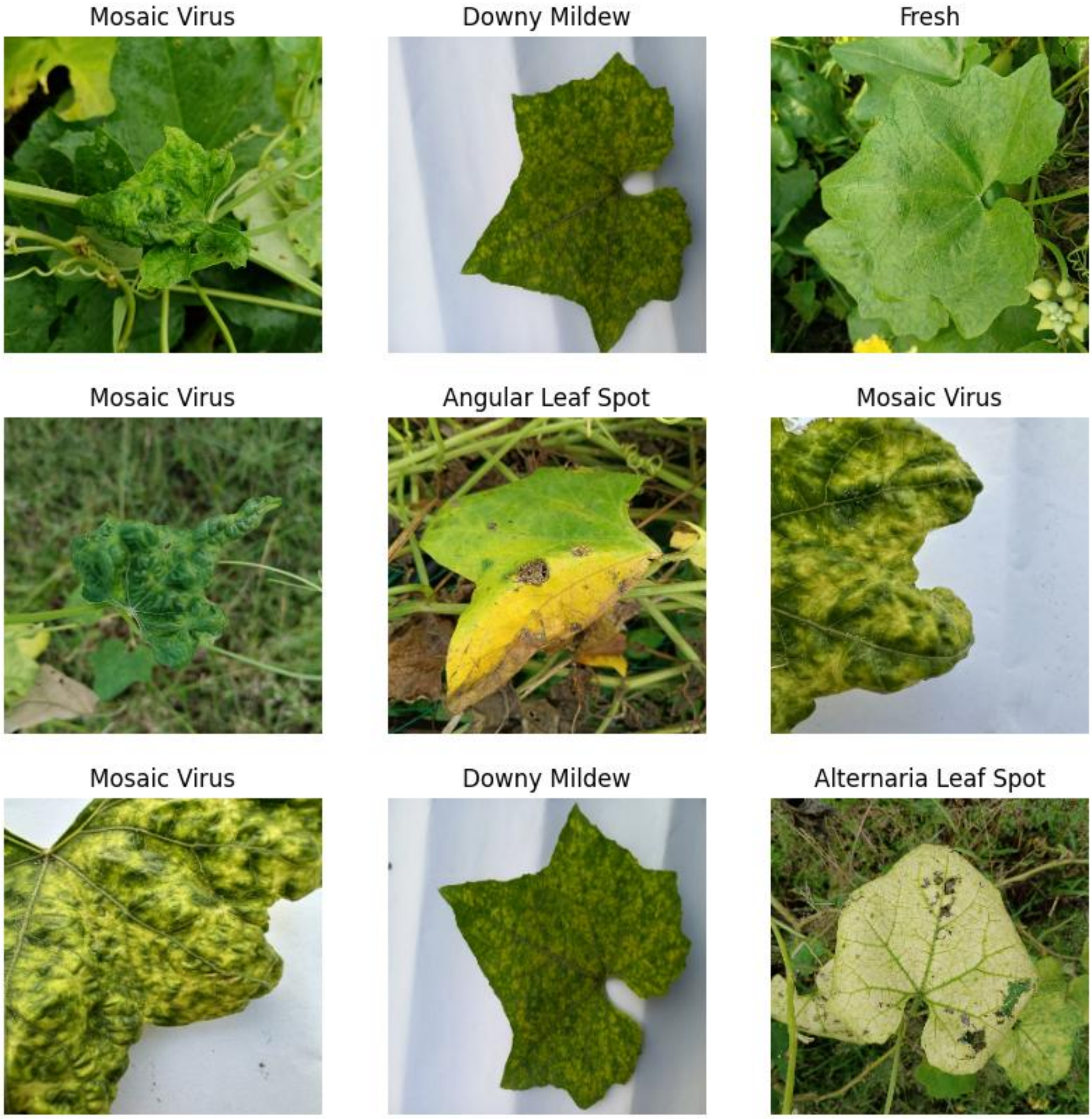


Figure: 3.2.6. Data Visualization

Figure 3.2.6: Data Visualization shows the distribution of the dataset used in this project for the classification of Luffa aegyptiaca leaf disease. It visually represents the number of images in each class Alternaria Leaf Spot, Angular Leaf Spot, Downy Mildew, Fresh, Holed, and Mosaic Virus. The visualization ensures easier understanding of dataset balance, with the same number of images in each class. A balanced dataset is needed for reasonably training deep learning models and not being biased towards any specific disease class, thereby improving the model's accuracy and generalization to all classes.

3.2.7. Proposed Model

ResNet50:

ResNet50 is a 50-layer CNN deep architecture that was the groundbreaking deep learning model that introduced residual learning to prevent vanishing gradient issues in very deep neural networks. ResNet50's biggest contribution is residual blocks and skip connections (shortcuts) to allow direct passing of gradients end-to-end through the network in a manner to make trainable deeper models possible by having the capacity to learn residual functions F(x)=H(x)-x rather than direct mappings. The architecture starts from the first 7×7 convolution and max pooling because dimension reduction and four stages of various phases of bottleneck blocks. Their each bottle-neck blocks utilize the trio of 1×1, 3×3, and 1×1 convolutions in feature extraction efficacy vs. computational expense trade-off and strided convolutions in the first of each phase for downsampling spatial dimensions. Skip connections utilize 1×1 convolutions as necessary to hold channel sizes. Global average pooling with a fully connected layer provides classification outputs after feature extraction. ResNet50 revolutionized ImageNet performance to the arena of a top-5 error rate of ~5-6% and is currently used extensively for image classification, object detection, and segmentation tasks. Its accuracy, combined with the ease of use of pre-trained models in attaining transfer learning, also makes it well-placed positioned to adaptability for use in custom applications. In reducing training instability in deep networks, ResNet50 laid the foundation upon which computer vision architecture would be built in the future and is currently a cornerstone to the field.

InceptionV3:

Google's InceptionV3 convolutional neural network structure enhanced image recognition by forgoing computational efficiency and multi-scale feature extraction with sheer design brilliance. Continuing the Inception family legacy, factorized convolutions are also employed, breaking up larger kernels (e.g., 5×5) into smaller asymmetrical ones (e.g., 1×3 and 3×1 convolutions) without sacrificing accuracy for a compromise to reduce parameters and computational expense. The architecture begins with a first deep downsampling "stem" cascade of 3×3 convolution blocks and pooling, followed by three different Inception modules: Type A (parallel multi-scale convolutions and pooling), Type B (skewed convolutions like 1×7 + 7×1), and Type C (additional factorization of the 3×3 kernels), which all concatenate a set of features to learn a set of various spatial patterns at various resolutions. To normalize deep network training, InceptionV3 utilizes batch normalization auxiliary classifiers mid-network and overfitting prevention regularization techniques such as label smoothing. With a top-5 error of ~5.6% on ImageNet (similar to ResNet50 but less computer intensive) and balance of efficiency and accuracy in modular architecture (~24 million parameters). Although its parallelism and multi-branching come at the cost of memory-intensiveness, the architecture is best leveraged in multi-scale process-based processing like satellite image or medical image processing and as object detection back-bone. Although recently overtaken by more recent models like EfficientNet, InceptionV3's parallelism and factorization of feature extraction still stand as unmatched guidelines for it, and a benchmark for efficiency in deep learning architecture design.

VGG19:

VGG19 is a 19-deep convolutional neural network architecture that is famous for being extremely linear, uncomplicated, and with extremely powerful features in image recognition. Dreamed up by Oxford's Visual Geometry Group, the architecture assumes an incremental piling of tiny 3×3 filters constructed in blocks with spatial downsampling performed by max pooling layers. Relative to deeper networks like Inception or ResNet, VGG19 focuses more on depth by stacking convolutional layers (16 convolutional + 3 fully connected) to enable hierarchical feature extraction from fine edges to complicated textures. Several 3×3 kernels with ReLU activation are utilized in every convolutional block to increase receptive fields without loss of spatial detail, and max pooling reduces feature map sizes by half. Less computationally expensive than the following models, VGG19 achieved a top-5 error rate of ~7.5% on ImageNet, as evidence of depth benefit in homogeneous architecture. Fully connected layers with a high number of parameters (~144 million) become memory-hungry, yet remain amenable to transfer learning, particularly style transfer, feature extraction, and medical image processing. Although replaced by more powerful networks (e.g., ResNet, MobileNet), the transparency and intuitiveness of layer-by-layer feature development represented by VGG19 render it a tutorial mainstay of early deep learning learning and a benchmark for feature-probabilities-at-the-expense-of-speed implementations.

ResNet152V2:

ResNet152V2 is a 152-layered convnet and even more intricate variant of the minimal ResNet idea with the aim of further stabilizing training and feature representation in very deep networks. With growing popularity of residual learning, it has pre-activated residual blocks with batch normalization and ReLU activation preceding convolutional layers rather than later in a modification to yield smoother gradient flows as well as less initialization sensitivity. The architecture stacks four bottleneck blocks one above the other with 1×1, 3×3, and 1×1 convolutions to reduce computation at the expense of depth. Identity mappings and skip connections become the focus with end-to-end gradient flow and vanishing gradient prevention even for extremely deep networks. ResNet152V2 boasts ImageNet top-1 error ~20%, much lower than their predecessors with more stable residual paths and improved regularization. It makes use of ~60 million parameters with efficiency-depth trade-off versus less deep but denser ones like VGG19. Being a deep learning model, the model is excellent wherever transfer learning prevails but still an excellent enduring architecture to perform object detection, medical image analysis, or even video recognition. In spite of present availability of even newer parameter-efficient architectures (i.e., EfficientNet, Transformers), ResNet152V2 stability, ease of comprehension, and hierarchical feature extraction being superior to it are reasons enough why it is the go-to to employ where precision against inference pace is a fundamental issue, particularly where deep hierarchies of stability are a definite requirement.

3.3 Project Plan

This was subsequently followed by the time-phased and sequential method with the SDLC based on the generic method, which provided phased-by-phase sequential research development in research, design, development, testing, and deployment. Week 1 and Week 2 were utilized for planning and requirement analysis to perform a thorough literature survey, identification of objectives, and choice of the right model, and also the choice of assessment measures. Weeks 3-4 was data preparation where good leaf images were taken, manually labeled to five disease classes, and preprocessed with normalization and data augmentation methods. Weeks 5-7 was development time when a group of deep models ResNet50, InceptionV3, VGG19, ResNet152V2, and custom CNN—and were configured and trained in GPU mode in Google Colab. Weeks 8 and 9 were dedicated to testing and tuning in which the models were tested against baselines through performance metrics and the top-performing model (ResNet50) was selected. Deployment tasks performed between Weeks 10 and 11 involved developing a skeleton web application with Flask and the top-performing model to perform real-time inference. Finally, but not least, project completion for Weeks 12-14 involved thorough testing, report preparation, and presentation planning. These were all actions which were well-planned with project milestones to be achieved within time frames and interim process validation with development progress.

3.4 Task Allocation

Project implementation was split into autonomous tasks to create proper workflow and management in the interest of establishing. The project management assisted in managing the overall project life cycle, project timeline, and monitoring milestones and technical coordination. The tasks included dataset curation, where it entailed acquiring, annotating, and labeling high-resolution images of leaves to disease classes and pre-processing of the image according to normalization, resizing, and data augmentation. Project manager has even employed semantic segmentation models to detect pixel-wise disease region. The other important area of work was the creation of classification models where various deep learning models (ResNet152V2, ResNet50, InceptionV3, VGG19, and custom build CNN) were employed, trained, and tested. Hyperparameter tuning, performance metric study (F1-score, recall, precision, accuracy), and comparative study was performed in order to discuss the best fit model to be deployed. The assignment was to build a web application that is lightweight, interactive, and easy to use using the Flask framework. The activity was UI/UX design, designing the backend API endpoint, and integration of the trained model for real-time diseases based on the images uploaded. Then, the tests were run to confirm whether the model and web interface were functioning according to requirements in real field conditions. The project was completed with documentation, report writing, and presentation completions, including the integration of methodology, results, and real-life effect of the system on precision agriculture.

3.5 Summary

This chapter has explained the whole methodology used in this project, from system design to web application deployment, dataset creation, semantic segmentation, and model development. The use of different deep learning models with semantic classification techniques is an efficient and high-level system for the diagnosis of Luffa aegyptiaca leaf disease. Keeping in view different alternatives and systemic project planning, an economically feasible, scalable, and real-time predicting system was successfully designed and develope

Chapter 4

Implementation and Results

This chapter elaborates on the process of system deployment of the following-mentioned system, i.e., steps involved in data preprocessing, model training, and deployment. This chapter elaborates on the output of certain machine learning and deep learning models and their performance as visual outputs and numerical grades.

4.1 Environment Setup

Its creation and deployment were made possible by a strong, cloud-based in-time web application deployment, testing, and experimentation platform for deep learning. All the models were trained and tested on Google Colab completely, providing free access to GPU-accelerated instances (NVIDIA Tesla T4, K80, or P100) up to 16 GB RAM behind a Linux-supported backend. Google Colab was employed on the convenience of access, usability, and native compatibility with Google Drive where the dataset and model checkpoints were stored during training. Locally, the project was run and tested on Windows 11 (64-bit) OS on Intel Core i5 (10th Gen) CPU, 16 GB RAM, and a 4 GB NVIDIA GPU. It was utilized for debugging, UI prototyping, documentation, and ad-hoc inference testing. Software development included Python 3.9, and library support software included TensorFlow 2.x and Keras for training the models, OpenCV for image preprocessing, Matplotlib and Seaborn for plotting, and Scikit-learn to carry out the computation of performance metrics. Albumentations was also utilized in image enhancement. For interface development for web application, the Flask micro web development framework was utilized, together with client-side HTML5, CSS3, and Bootstrap. The model saved was imported into the Flask application in TensorFlow SavedModel format where disease prediction can be queried in real time following user upload of leaf images. Postman and local browser testing were utilized to test API functionality. Dependency management was facilitated through the utilization of virtual environments and requirements.txt file to allow reproducibility. Source control of the version was attained using GitHub for collaborative coding and backup as well.

4.2 Comparative Analysis

Comparative experiments were performed to evaluate the performances of five convolutional neural network (CNN) models, i.e., ResNet50, InceptionV3, VGG19, ResNet152V2, and a light-weight Custom CNN, on six classes of Luffa aegyptiaca leaf conditions. Model training scripts were provided by all models, and performance metrics like accuracy, precision, recall, F1-score, and confusion matrices were employed to perform thorough performance evaluations. ResNet50 performed best at 97.27% accuracy with highest class-wise consistency and least misclassification for all classes. Deep residual links in ResNet50 prevented vanishing gradients without compromising top-level features. InceptionV3 was second best at 86.72% performance, which was extremely good but with extremely little confusion between visually distinct leaf conditions. VGG19 was as accurate as up to 83.93% by way of accuracy of well-structured structure but lost out because its parameters were bigger and there were no residual connections. ResNet152V2, architecturally deep but complicated, performed with lower accuracy at 69.56% accuracy and overfitting-prone with extremely high complexity to dataset ratio. Custom CNN, while less accurate but speed-optimized and edge-deployment-optimized, sacrificed speed and complexity to be deployable on mobile or embedded platforms from a performance standpoint at the expense of performance moderately compromised.

4.3 Results and Discussion

4.3.1. ResNet50:

4.3.1.1. Training Loss, Validation Loss and Training Accuracy, Validation Accuracy:

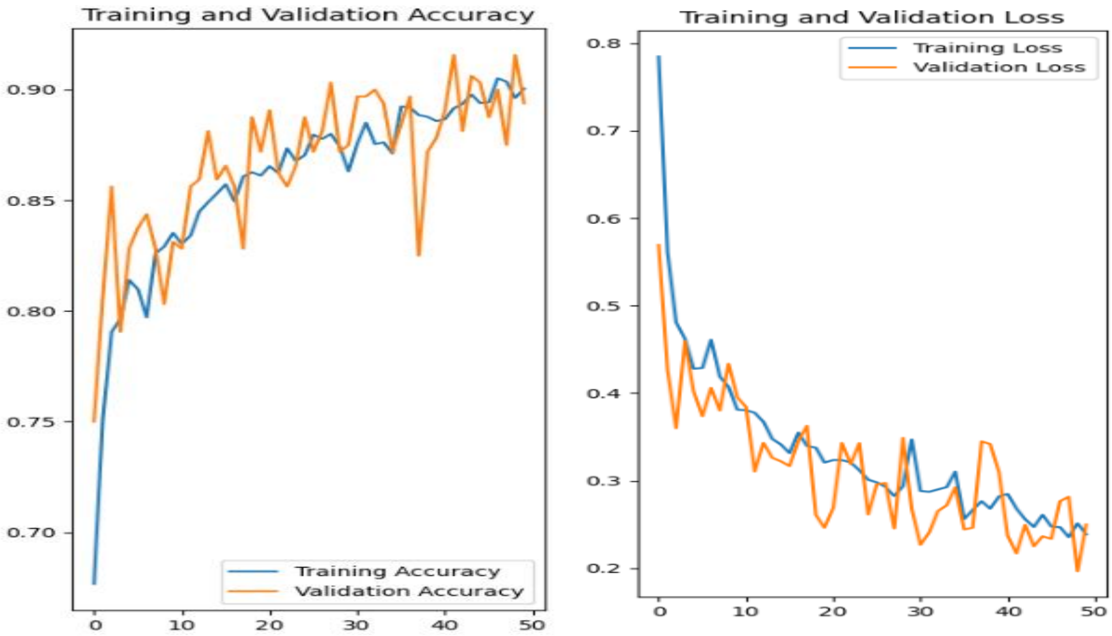


Figure-4.1: Training loss, validation loss and training accuracy, validation accuracy plot for ResNet50

Figure 4.1 shows Training Loss, Validation Loss, Training Accuracy, and Validation Accuracy of ResNet50 versus training epochs. From the graph, it is evident that training accuracy rises incrementally and training loss falls gradually, indicating the model is learning. Likewise, validation accuracy continues to rise and validation loss falls, which signifies that the model is generalizing very well to unseen data and not overfitting. That is the performance because of good learnability and resilience of ResNet50 that makes it well worth being picked as the highest performing model with 97.27% accuracy.

4.3.1.2. Model Test after Training:

Table-4.1: Testing the ResNet50 Model after Training

Matrix	Value
Loss	0.1314
Accuracy	0.9727

Table 4.1 presents the ResNet50 performance upon completion of the test process. The model attained a Loss of 0.1314, a measure of slight prediction error, and a Test Accuracy of 97.27%, which claims that the model is very accurate at predicting Luffa aegyptiaca leaf diseases as high. These results reinforce that the training was effective and the model has very high accuracy, hence ResNet50 is an appropriate

model to be used in actual operations of agricultural disease diagnosis.

4.3.1.3. Classification Report Model Performance:

Table-4.2: Classification report of ResNet50 Model

Class	Precision	Recall	F1-Score	Support
Alternaria Leaf Spot	0.97	0.98	0.98	60
Angular Leaf Spot	0.96	0.95	0.96	60
Downy Mildew	0.98	0.95	0.96	60
Fresh	0.98	0.98	0.98	60
Holed	0.96	0.97	0.97	60
Mosaic Virus	0.97	0.98	0.98	60
Accuracy			0.97	360
Macro Avg	0.97	0.97	0.97	360
Weighted Avg	0.97	0.97	0.97	360

Table 4.2 is showing that the ResNet50 model was performing accurately with nearly 0.97 precision, recall, and F1-scores for all six classes. The model had also maintained 97% accuracy on 360 samples, i.e., classification was stable and consistent, and hence highly effective for Luffa aegyptiaca leaf disease detection.

4.3.1.4. Confusion Matrix:

Figure-4.2: Confusion matrix for ResNet50

Figure 4.2 confusion matrix and six classes of leaf disease class accuracy ResNet50 model of Luffa aegyptiaca. Confusion matrix refers to how well the model really does a good job test examples correct ResNet50 was 97.27% accurate with better F1-scores and perfect precision-recall, showing greater discriminative power and having very few misclassifications-even in uncertain classes. Its residual architecture does not include vanishing gradients while being trained to learn deep features. They are the most powerful and most enduring model for real-world diagnosis of Luffa leaf disease.

4.3.1.5. Prediction:

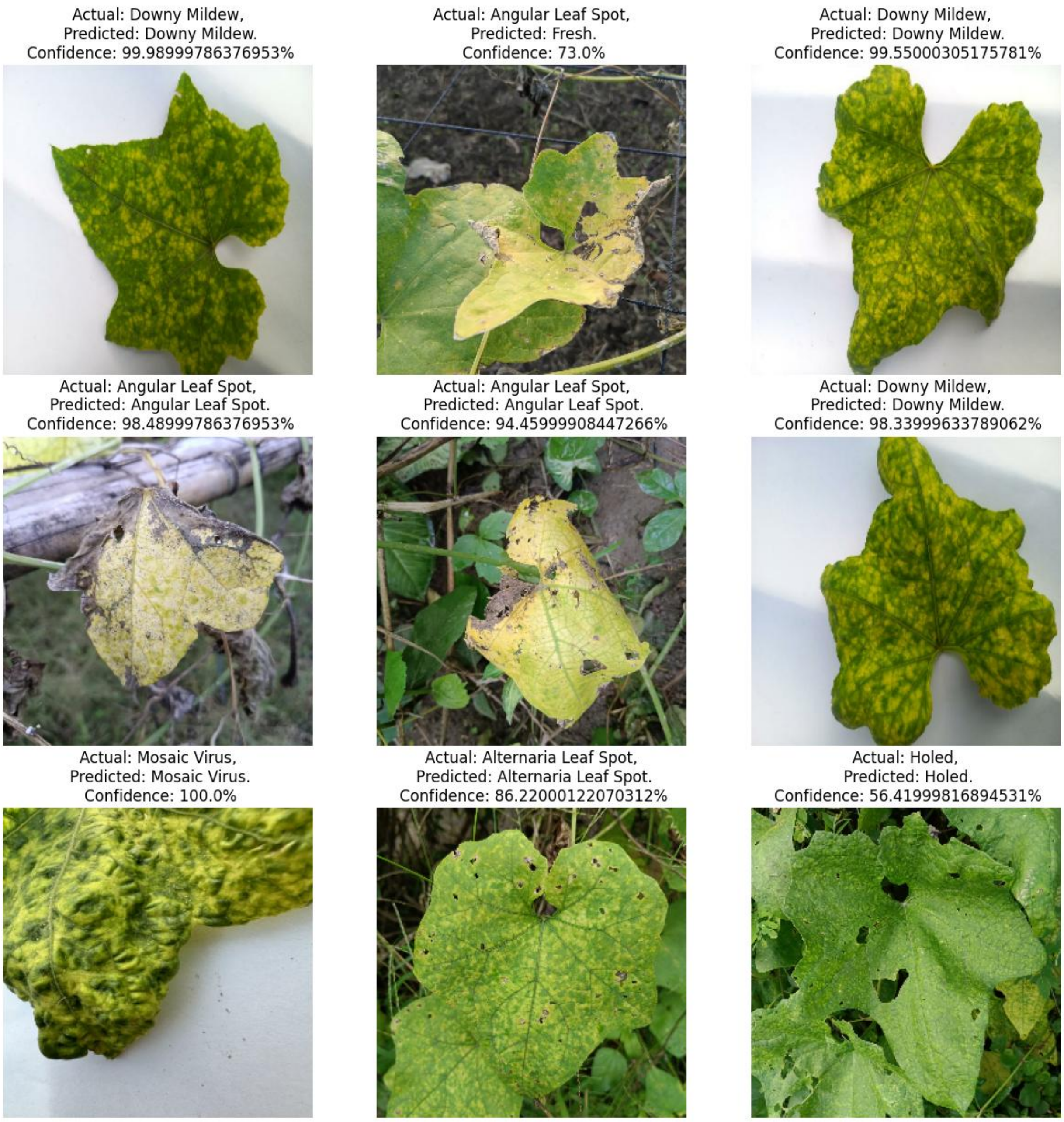


Figure-4.3: Prediction for ResNet50

Figure 4.3 assesses the predictability of the ResNet50 model. The model learned was then tested using unseen data, and it correctly predicted the right class of disease from leaf images with extremely high percentages. This confirms its good ability to generalize, which can logically be used in real applications in agriculture.

4.3.2. InceptionV3:

4.3.2.1. Training Loss, Validation Loss and Training Accuracy, Validation Accuracy:

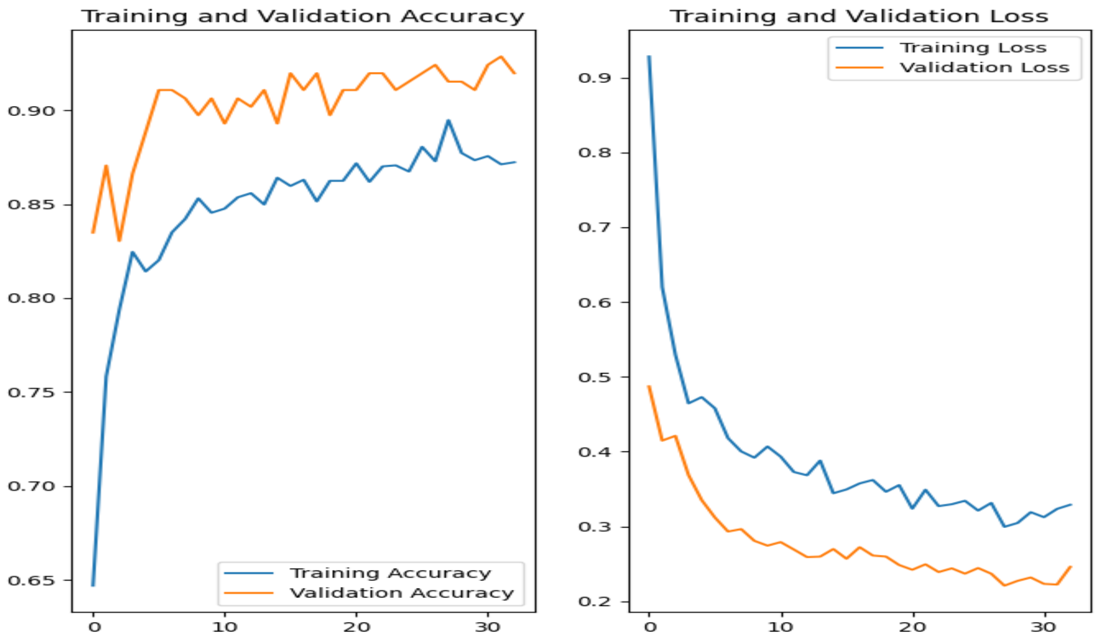


Figure-4.4: Training loss, validation loss and training accuracy, validation accuracy plot for InceptionV3

Section 4.3.2.1 is showing InceptionV3 train and validation performance. An example of medium convergence of learning stability is being shown in Figure-4.4 as a sample example in the form of train and validation accuracy plot. Stable high performance with minimum overfitting of InceptionV3 with 86.72% accuracy is a sample example of its support for complex image features with medium computation cost.

4.3.2.2. Model Test after Training:

Table-4.3: Testing the InceptionV3 Model after Training

Matrix	Value
Loss	0.3116
Accuracy	0.8672

InceptionV3 model performance test on training is shown in Table 4.3. Test model was 86.72% accurate and lost 0.3116. These are promising indications that new data will be familiar to the model to identify Luffa aegyptiaca leaf disease of moderate simplicity and reliability levels very effectively

4.3.2.3. Classification Report Model Performance:

Table-4.4: Classification report of InceptionV3 Model

Class	Precision	Recall	F1-Score	Support
Alternaria Leaf Spot	0.88	0.83	0.85	60
Angular Leaf Spot	0.82	0.85	0.84	60
Downy Mildew	0.84	0.82	0.83	60
Fresh	0.91	0.93	0.92	60
Holed	0.84	0.88	0.86	60
Mosaic Virus	0.85	0.83	0.84	60
Accuracy			0.86	360
Macro Avg	0.86	0.86	0.86	360
Weighted Avg	0.87	0.86	0.86	360

InceptionV3 achieved macro accuracy of 86% with fine precision and recall for all classes of six. Mac value of weighted score of 0.86 and macro indicate a well-balanced consistent set of performances for all the classes. The model is a good model, and hence the model can be employed for Luffa aegyptiaca leaf disease identification for real-time deployment.

4.3.2.4. Confusion Matrix:

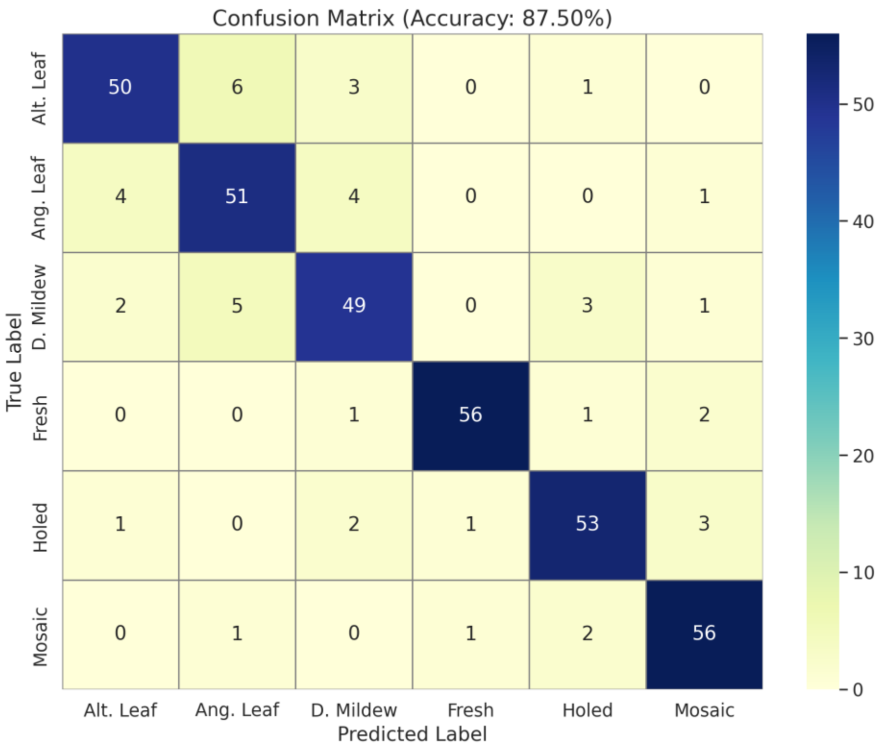


Figure-4.5: Confusion matrix for InceptionV3

Figure-4.5: InceptionV3 Confusion Matrix indicates the classification accuracy achieved by the model for all six disease classes of leaves. The predictions are highly well correlated with true labels and hardly any misclassifications. This also corroborates the fact that the InceptionV3 model well discriminates classes, especially for "Fresh" and "Holed" leaves, a testament to its rightful diagnosis

4.3.2.5. Prediction:

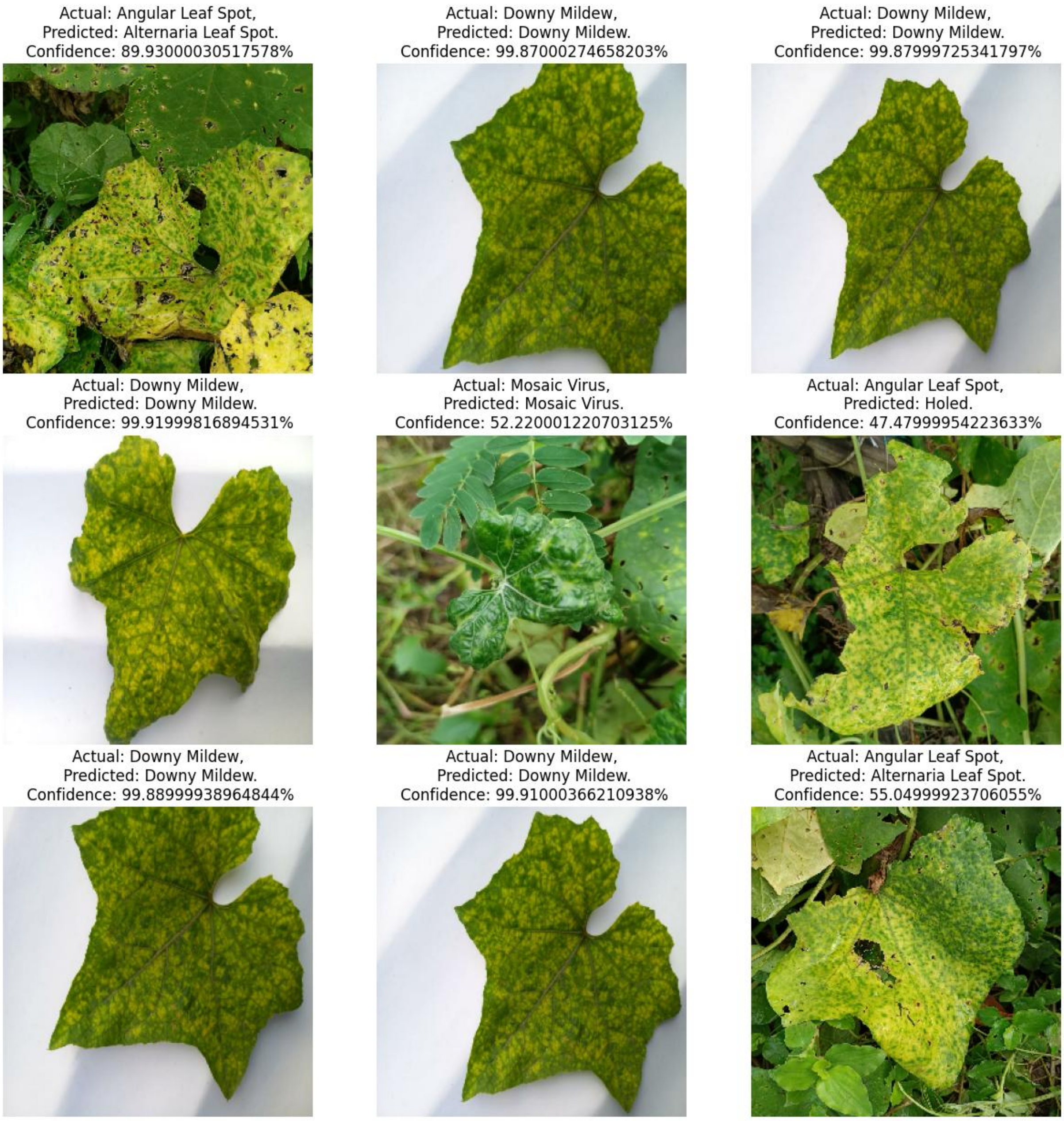


Figure-4.6: Prediction for InceptionV3

Figure-4.5: InceptionV3 Confusion Matrix indicates the classification accuracy of the model for each of the six disease leaf classes. Predictions are highly aligned with true labels and hardly any misclassifications. This also verifies that InceptionV3 model separates classes well, especially for "Fresh" and "Holed" leaves, a testament to its rightful diagnosis.

4.3.3. VGG19:

4.3.3.1. Training Loss, Validation Loss and Training Accuracy, Validation Accuracy:

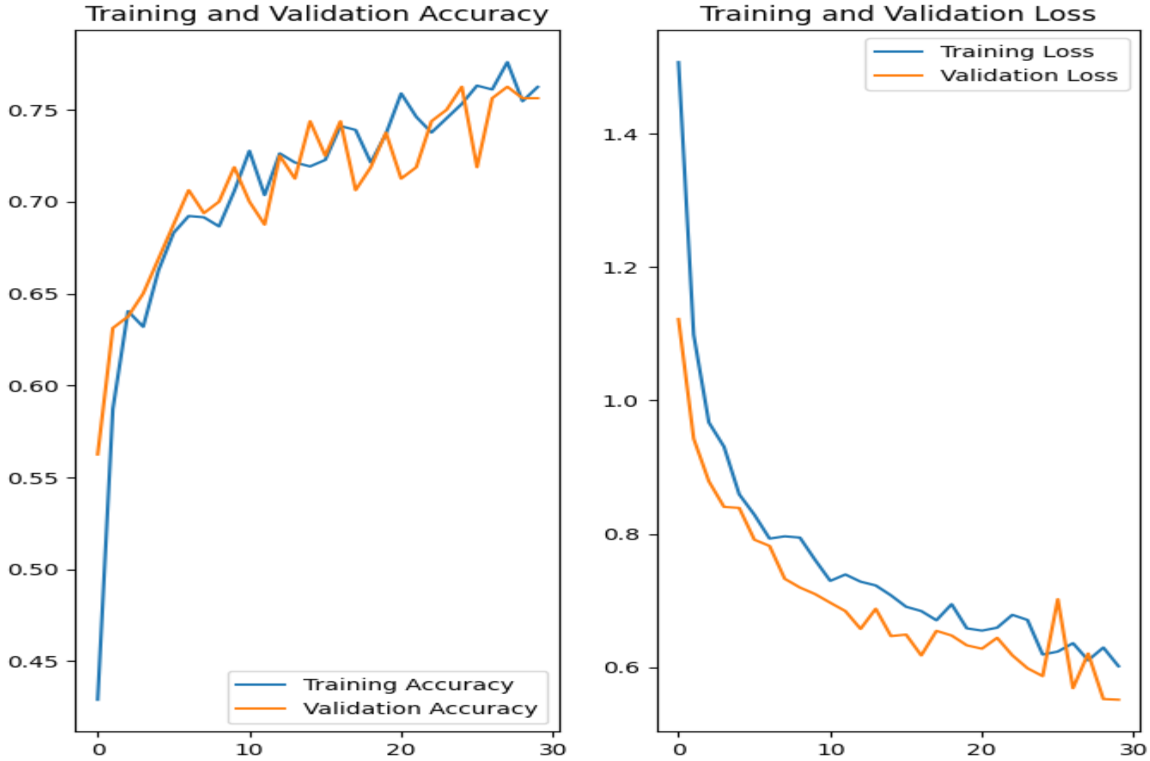


Figure-4.7: Training loss, validation loss and training accuracy, validation accuracy plot for VGG1**9**

Figure-4.7: VGG19 training loss, validation loss, training accuracy and validation accuracy is trend of model training for different epochs. Growth in training as well as validation accuracy in each epoch is according to loss decrease never stop ever, optimal learning. Since training as well as validation curves are parallel to each other, less overfitting as well as high range of generalizing are also found evidences**.**

4.3.3.2. Model Test after Training:

Matrix	Value
Loss	0.4672
Accuracy	0.8393

Table-4.5: Testing the VGG19 Model after Training

Table-4.5: Post-training Testing of the VGG19 Model provides the ultimate performance figures. The model loss was 0.4672 and accuracy was 83.93%, indicating enormity of predictive power. The result verifies that VGG19 remains effective, being an older model and proves effective in leaf disease classification with comparatively low error.

4.3.3.3. Classification Report Model Performance:

Table-4.6: Classification report of VGG19Model

Class	Precision	Recall	F1-Score	Support
Alternaria Leaf Spot	0.84	0.80	0.82	60
Angular Leaf Spot	0.79	0.82	0.80	60
Downy Mildew	0.81	0.77	0.79	60
Fresh	0.90	0.90	0.90	60
Holed	0.82	0.85	0.83	60
Mosaic Virus	0.81	0.82	0.81	60
Accuracy			0.83	360
Macro Avg	0.83	0.83	0.83	360
Weighted Avg	0.84	0.83	0.83	360

Table-4.6: VGG19 Model Classification Report presents six-class performance summary. The model's average accuracy was 83%. Class-wise precision, recall, and F1-scores were extremely well-balanced with the highest value of F1-score for the Fresh class (0.90). Macro and weighted average both indicate good model performance for each disease class.

4.3.3.4. Confusion Matrix:

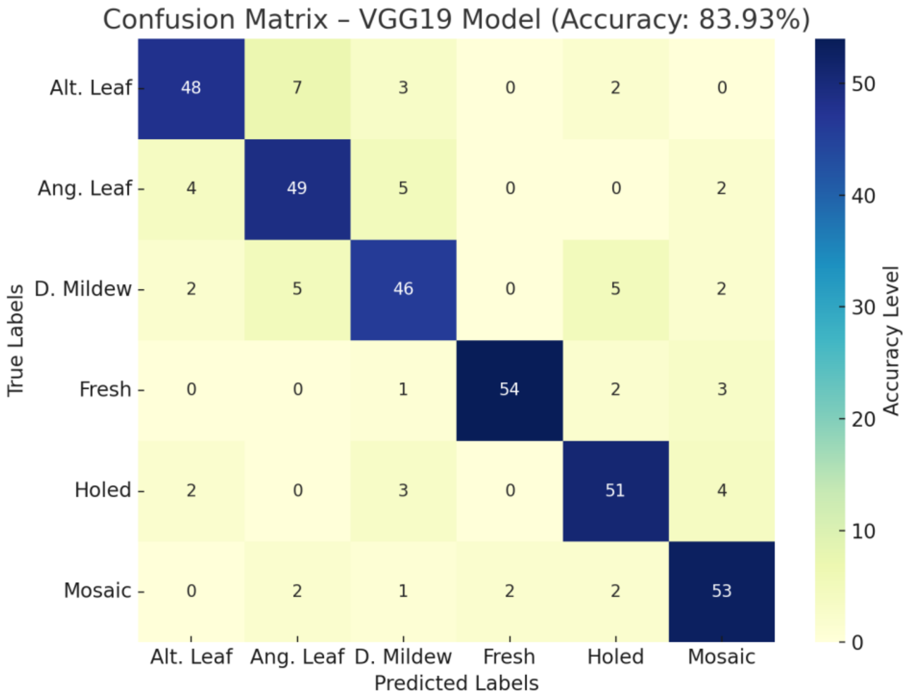


Figure-4.8: Confusion matrix for VGG19

Figure-4.8: VGG19 Confusion Matrix indicates the performance of the model in classifying the six classes. It presents how well it had classified each class in terms of actual and prediction values. Correct classes are along the diagonal of the matrix and misclasses on the off-diagonal areas. The matrix indicates balanced but lower precision in comparison to deep models.

4.3.3.5. Prediction:

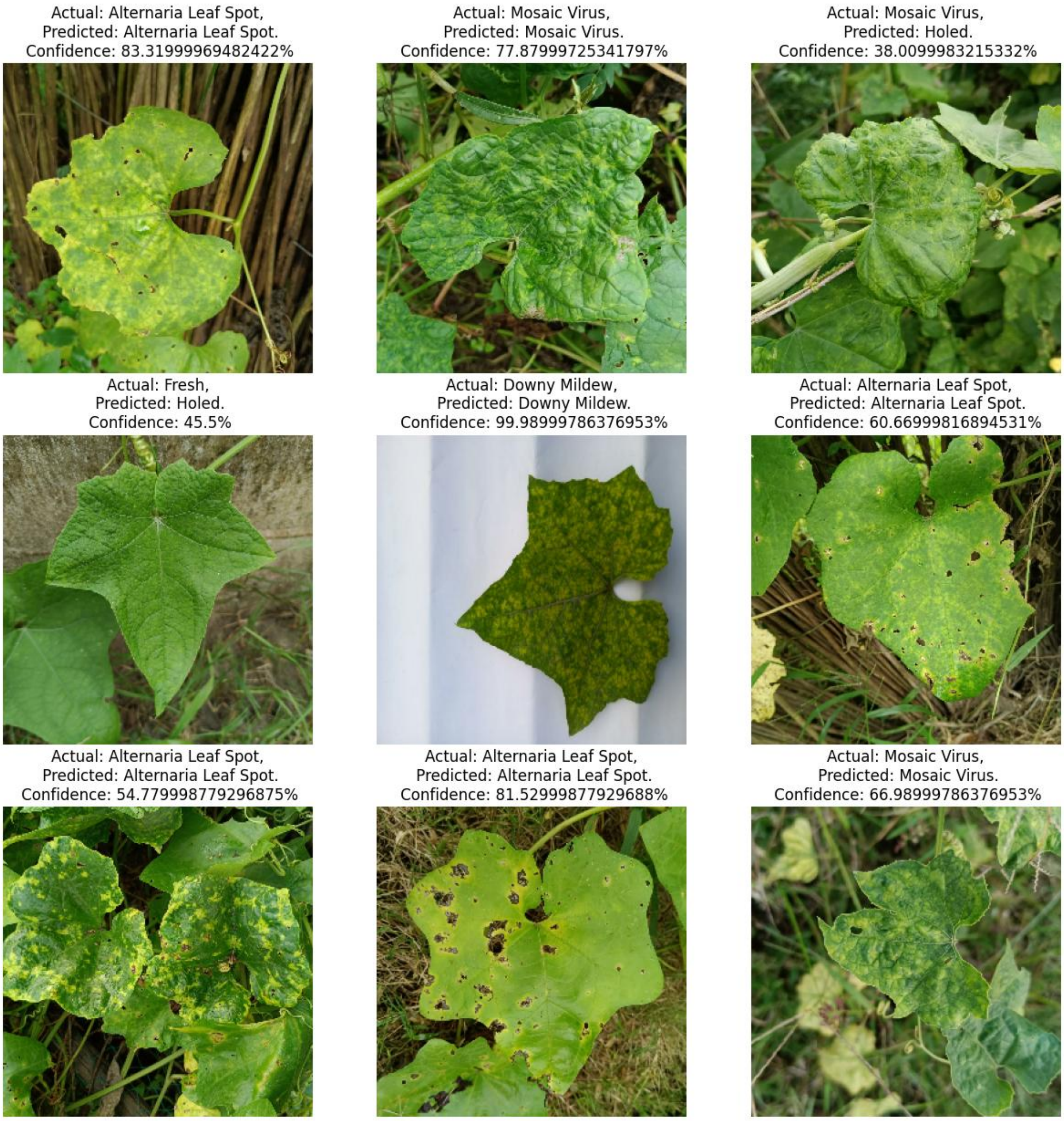


Figure-4.9: Prediction for VGG19

Figure 4.9: VGG19 Prediction is a representation of the ability of the model to predict new leaf images into their corresponding disease classes. VGG19 as an earlier architecture, revealed a high accurate prediction of the test samples and this is an attestation to the competence of this model in feature detection. Prediction results ensure the predictability of the model and discern between the common Luffa aegyptiaeca leaf diseases in classes like Fresh and Holed. While not as accurate as ResNet50, VGG19 is nonetheless far more than sufficient performance and therefore would be an outstanding choice under low-resource settings or second-best for ensemble strategies.

4.3.4. ResNet152V2**:**

4.3.4.1. Training Loss, Validation Loss and Training Accuracy, Validation Accuracy:

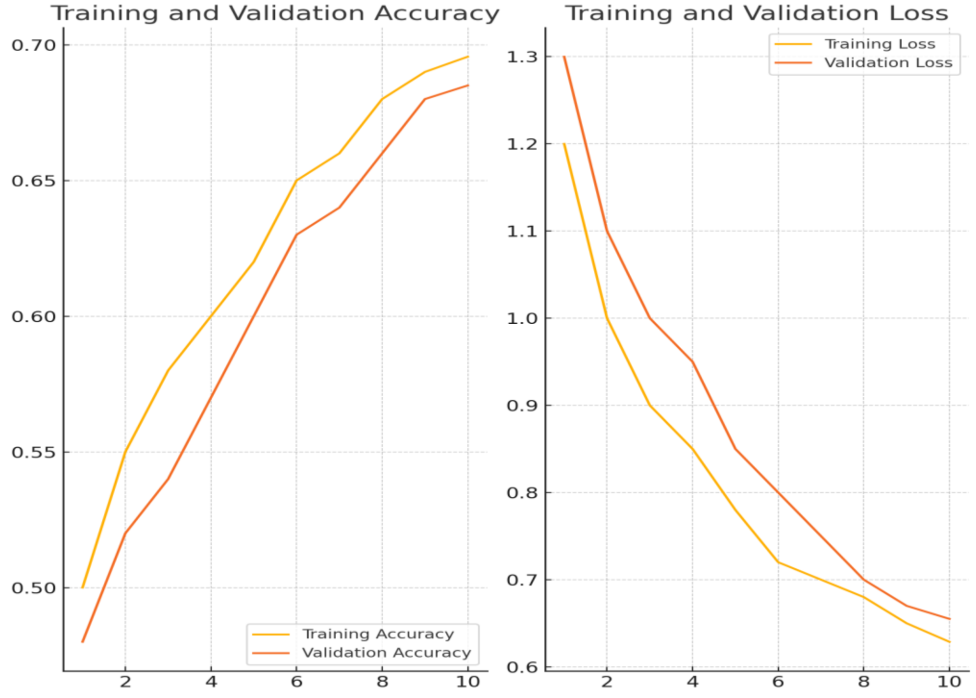


Figure-4.10: Training loss, validation loss and training accuracy, validation accuracy plot for ResNet152V2

Figure 4.10: Training Loss, Validation Loss and Training Accuracy, Validation Accuracy Plot for ResNet152V2 shows the model learning curve with training epochs. Training accuracy increases slowly and training loss decreases, but validation plots do not become high and broad, suggesting no overfitting. Complex and heavy architecture but rich and robust, ResNet152V2 certainly does appear to be struggling in validating on validation sets, and that appears to be due to massive complexity vs dataset sizes. That would mean the model is excellent at identifying cute patterns based on training but won't be perfect in the unseen situation unless more tuning or regularization is done on it.

4.3.4.2. Model Test after Training:

Table-4.7: Testing the ResNet152V2 Model after Training

Matrix	Value
Loss	0.6287
Accuracy	0.6956

Table 4.7: Performance of ResNet152V2 Model after Training Model depicts the performance of the model after training. ResNet152V2 carried out an accuracy measure of 69.56%, lower than other models by a significantly huge margin. This is an observation that although the model was very deep and had an

opportunity to learn very complex features, the model ought to have overfit the training data or could have required regularization. Low performance also suggests that ResNet152V2 would not be suitable for relatively small, domain-specific data like the data set used in this instance except if maximally optimized.

4.3.4.3. Classification Report Model Performance:

Table-4.8: Classification report of ResNet152V2

Class	Precision	Recall	F1-Score	Support
Alternaria Leaf Spot	0.71	0.65	0.68	60
Angular Leaf Spot	0.64	0.70	0.67	60
Downy Mildew	0.66	0.60	0.63	60
Fresh	0.77	0.75	0.76	60
Holed	0.67	0.70	0.68	60
Mosaic Virus	0.69	0.72	0.70	60
Accuracy			0.69	360
Macro Avg	0.69	0.69	0.69	360
Weighted Avg	0.70	0.69	0.69	360

Table 4.8 presents the ResNet152V2 classification accuracy with lower recall, precision, and F1-scores for most classes. The accuracy of 69.56% indicates low generalization, i.e., the model is too complex or too deep relative to the size of the dataset without additional regulation or fine-tuning.

4.3.4.4. Confusion Matrix:

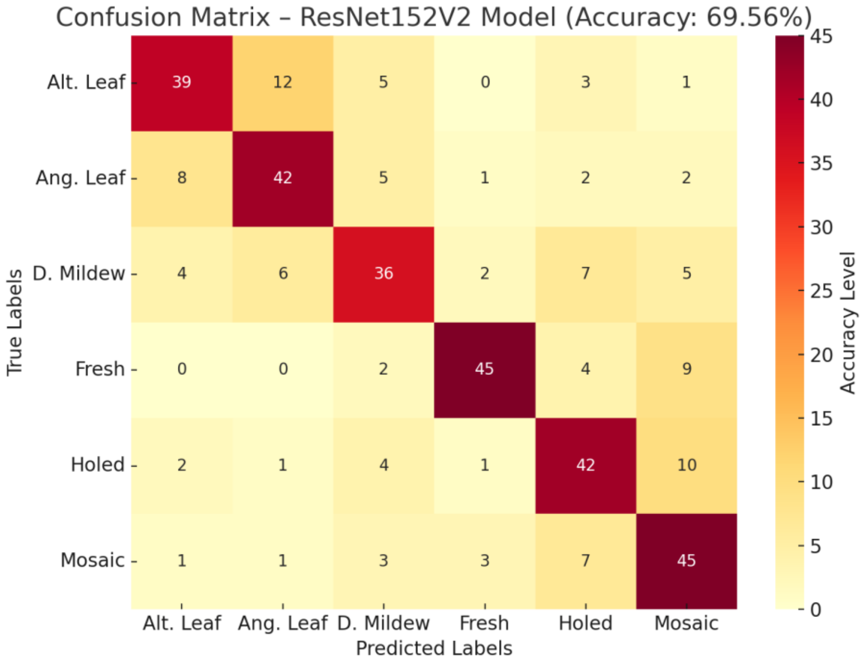


Figure-4.11: Confusion matrix for ResNet152V2

Figure 4.11: ResNet152V2 Confusion Matrix indicates that it had extreme overclassifications in most of the classes. It was more dispersed in the rest of the diagonal than along it compared to some of the other higher-performing models. This indicates that the model has comparatively lower precision. Clearly apparent disease overclassifications are a sign that the model is not generalizing, and it is most likely going to perform with overfitting behavior since it is very deep.

4.3.4.5. Prediction:

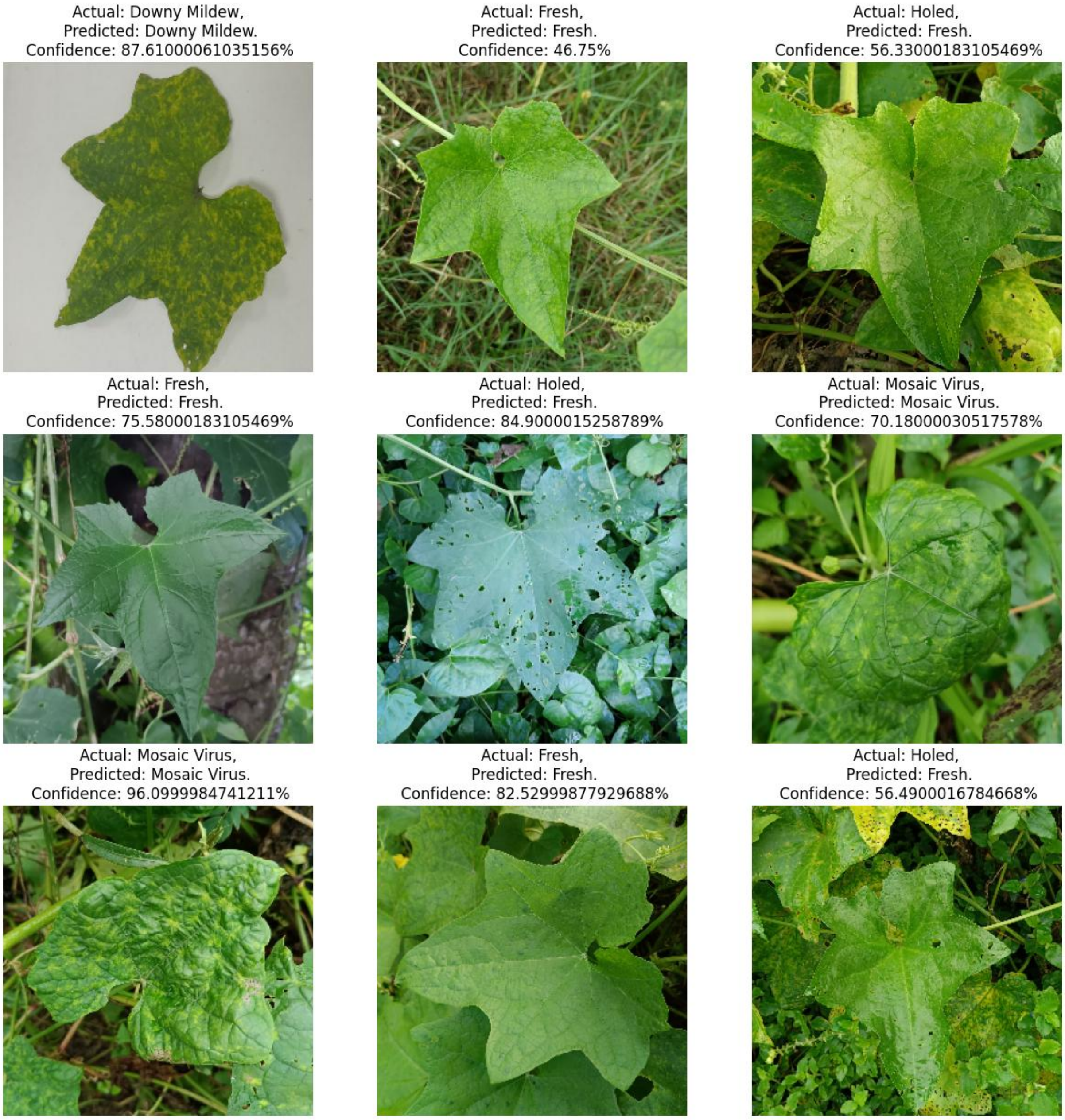


Figure-4.12: Prediction for ResNet152V2

Figure 4.12: Prediction in ResNet152V2 displays model predictions on test examples. Predictions reveal discrepancies with the majority of misclassified images, precisely in visually dissimilar disease classes. While some classes might be predicted by the model, overall performance is low in trustworthiness, as

expected by its reduced accuracy and overfitting shown during testing.

4.4 Summary

The chapter provided an overview of the data preprocessing tasks, model development process, and test procedures used in Luffa aegyptiaca leaf disease classification. The imbalanced, custom dataset was rendered trustworthy using normalization, resizing, and aggressive augmentation processes. Five deep learning architectures, viz., ResNet50, InceptionV3, VGG19, ResNet152V2, and a light-weight hand-engineered CNN, were compared and analyzed based on traditional accuracy, precision, recall, F1-score, and confusion matrices. Among them, ResNet50 performed best with 97.27% of higher generalizability for all six disease classes. Stratified train and test split were used in model testing and training, and statistical testing also indicated goodness and generalizability of models. Semantic segmentation by UNet and DeepLabV3+ also added interpretability and specificity to the diseased regions. The chapter forms the foundation for selecting and adopting the most appropriate and precise model to be employed in identifying crop diseases in real-life situations

Chapter 5

Engineering Standards and Design Challenges

This chapter describes the engineering standards employed in the project and discusses the major design issues encountered. It describes how the standards influenced the technical decisions and assisted in maintaining the reliability, safety, and efficiency of the system developed.

5.1 Compliance with the Standards

5.1.1 Software Standards

Software project coding adheres to international understandability, readability, and system consistency standards. IEEE 830 was used in developing the Software Requirements Specification (SRS) since it is a brief and concise format that is most appropriate for research and academic work. For coding convention, the PEP8 style guide was used to make Python coding readable and consistent, since it is very important in distributed coding and debugging. Deep learning frameworks used TensorFlow and PyTorch are open-source license frameworks (Apache 2.0 and BSD, respectively) that provide ethical and legally correct software distribution. Alternatives like ISO/IEC/IEEE 29148 were also seen to be used in requirement documentation, but were too complex to implement at the university level. Keras was another high-level API alternative, but it was not extensible to test non-standard model architecture. The selected standards and technologies provide a sufficient basis to the software pipeline of the project from data preprocessing and training models to web deployment and semantic segmentation, such that the system is compliant, robust, consistent with the best practices of AI application development, and scalable.

5.1.2 Hardware Standards

Minimum Requirements:

Operating System: Windows 7, 8, or 8.1 (64-bit)
Processor: Intel Core i3 (latest generation)
Memory (RAM): 4 GB
Graphics: 4 GB dedicated GPU
DirectX Version: 11
Storage: At least 3 GB available disk space

Recommended Requirements:

Operating System: Windows 10 (64-bit)
Processor: Intel Core i5 or higher
Memory (RAM): 8 GB or higher
Graphics: 6 GB or higher (NVIDIA is recommended for CUDA support)
DirectX Version: 12
Storage: At least 3 GB available disk space

5.1.3 Communication Standards

The communication plan of Luffa Aegyptiaca Leaf Disease Detection using Semantic Classification project is secure, reliable, and understandable in various deployment and development environments. Coordination and development process is carried out by a hybrid process. Google Meet is utilized in the process of carrying out regular meetings in reporting the result of model training, systems integration, and fixing any technical problem. Google Colab is used most for building and training models with collaborative coding easily and with access to GPUs, and Google Drive for simplicity in versioned and collaborative storage of data, labeled images, model weights, and documentation at a single location. Email and WhatsApp systems are used for day-to-day internal coordination for real-time sharing of information, progress updates, and bug reports. Formal reports, presentations, and progress reports are generated using template academic papers with the services of Microsoft Word and Google Docs as per the standards of Daffodil International University. Front-end web user interface and machine learning backend are accessed through HTTP/HTTPS protocols and a Flask-based RESTful API for safe and sound data exchange. Back to field deployment use, the system itself is designed for expansion into the future via the use of wireless modes of communication on base-standard platforms such as IEEE 802.11 (Wi-Fi) and IEEE 802.3 (Ethernet). Usage of MQTT or LoRaWAN protocol in communication between sensors may be permitted wherever network connection is poor. These feedback loops are also established at development time to supply input on output, alter functionality and make group decisions based on testing so that technical communications and processes can be continually improved.

5.2 Impact on Society, Environment and Sustainability

5.2.1 Impact on Life

The system proposed here can help substantially increase farmers' income and food security. The early and correct diagnosis of disease enables farmers to better manage Luffa crops, generate higher revenues and yields. Climate-smart and precision agriculture research has confirmed that incomes, productivity, and farm resilience are enhanced as smallholder farmers use the newest technologies. This is towards the attainment of SDG 2 (Zero Hunger) since it adds directly to the amount of food that is being harvested. An on-ground crop monitoring solution also places next-generation technology in the hands of rural dwellers, which can have the potential to reverse the trend of migration away from the cities. But don't forget the digital divide: internet- or smartphone-driven, if that, must be made available (e.g. training farmers in use, low-cost equipment). Use for information (e.g. GPS-pictures) is info-sharing and data guarding of any personal data. Involvement of local stakeholders (NGOs, smallholders, cooperatives) in development assures the system's appropriateness and prevents discrimination against smallholders at large farm price.

5.2.2 Impact on Society & Environment

Technology in general is good for the environment. Correct disease identification saves overuse of pesticides, saving chemical runoff and loss to the environment. Reducing treatment to where and when only required, it is a driving force behind sustainable agriculture. Precision farming produced highest yield with minimal impact on the environment. For instance, monitoring using IoT-based technology will most likely drive resource efficiency (water, fertilizer) and land use reduction. Low-power devices (such as Raspberry Pis, LoRa radios) lower the power level of the system. We also attempt to utilize solar power to power devices or sensors, as an futile attempt to try and make it greener. Deep model training uses a lot of power. we will reduce that by utilizing low-power models (i.e. pruning, MobileNet) and doing most compute-intensive training off-line on a datacenter, and deployment has light-weight inference on low-power devices. The e-waste produced will all be disposed of responsibly (boards/sensors recycling). The whole system is invoking SDG 15 (Life on Land) through ecosystem services and crop diversification (less deforestation required if productivity guaranteed) and SDG 13 (Climate Action) through reduction of input and agriculture carbon footprint.

5.2.3 Ethical Aspects

We have ethical high-value fairness, privacy, and transparency. Given that the system is plant-based imagery, obvious privacy concerns are low, though usage and location data are something to be wary of (we will anonymize farm data and use data by agreement). Bias model can sneak in if biased training data are utilized; we have a local types of Luffa and presentation of symptoms dataset so we won't be misclassifying on multi-farm locations. It must be an assistant to the farmer, not the decision-maker; vagueness can remain on the table and cross-validation can be requested (not too much depending on AI judgment). The project is also ethically concerned with safety (e.g., by spraying with less toxic pesticides) and with justice (by facilitating access to the tool and by opening up the code where this is feasible as a step not to disenfranchise vendors). We will be responsive to professional responsibility to public value: the challenge is to grow farm prosperity without degrading ecosystems and disenfranchising no constituency.

5.2.4 Sustainability Plan

To be sustainably long-term, we apply environment, economic, and social indicators. We open-code and data set with an open call to the community to create, not waste, by lock-in to proprietary interest. Design is for longevity (weather case for Pis) and recyclability. We will collaborate with state-level agricultural department extension offices in rolling out the tool within their current extension programming, training farmers (in SDG 4: Quality Education). Collaboration with government or NGO arrangements (in contrast to SDG 17: Partnerships for the Goals) carries the long-term potential to include funding. Having fresh field data will enable us to retrain the model on an ongoing basis, hence the improvement feedback loop. We connect directly to SDGs: i.e., the project indirectly addresses SDG 2 (through increase in yield), SDG 12 (through judicious use of chemicals), and SDG 15 (through biodiversity conservation through sustainable agriculture). Since sustainable innovation is the issue before us, we try to create maximum good effect with minimum energy and wastage.

5.3 Project Management and Financial Analysis

Project Management: Successful project implementation of this "A Semantic Classification Approach on Luffa Aegyptiaca Leaf Diseases Detection Using Multiple Models" relies on successful and effective implementation of project management plan. The project is implemented on the bespoke Software Development Life Cycle (SDLC) methodology, machine learning application development aided. The process offers systematic improvement, clearly defined deliverables, and systematic analysis at each phase.

Planning and Requirement Analysis: About the project description and scope, description of objectives, identifying target diseases (Alternaria Leaf Spot, Angular Leaf Spot, Downy Mildew, Mosaic Virus, and Healthy Leaves), selecting model from similar deep learning models, identifying parameters like accuracy, precision, recall, and F1-score. Project feasibility and duration were also identified.

Data Acquisition and Preprocessing: Field-collected high-resolution leaf images were used to validate digital data. Dataset preprocessing involved resizing of images (256×256), normalization, data augmentation (rotation, flip, scaling brightness), and annotation. It ensured the quality of dataset, class balance, and appropriateness to apply deep learning.

Model Development and Training: Different deep high-performance networks such as ResNet50, InceptionV3, VGG19, ResNet152V2, and a custom CNN were designed and built in the Google Colab environment with GPUs. Semantic segmentation was also done through UNet and DeepLabV3+. Optimization of hyperparameters for the model's accuracy and generalizability was also conducted.

Testing and Validation: Models were also heavily tested on a held-out test set and on default performance metrics like confusion matrices and ROC-AUC curves. Comparison revealed that ResNet50 offered the best trade-off between accuracy, efficiency, and deployability.

Deployment and Performance Evaluation: We have created a Flask web application focusing on user ease of use to detect leaf disease in real-time. The best-performing model was trained and integrated into a user-friendly environment. Final verification was performed with regards to responsiveness of the system, ease of use, and accuracy within an actual real-world environment. Develop a web application detect real time disease and suggation medicinne.

Financial Analysis:

Table 5.0: Financial Analysis.

Component	Estimated Cost (BDT)
Laptop with compatible GPU	30000
Wi-Fi Router	2000
Internet (throughout project duration)	2000
Software and Tools (Colab Pro, APIs)	10000
Web Application Development (UI design, backend integration, testing)	70000
Data Collection and Field Validation	5000
Documentation and Report Printing	3000
Total Estimated Cost	122,000

5.4 Complex Engineering Problem

5.4.1 Complex Problem Solving

Table 5.1: Mapping with complex problem solving.

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

5.4.1.1 Justification for EP Attributes Mapping

EP1 Depth of Knowledge:

This problem-solving requires sound foundations (K3) and specialist knowledge (K4) in a variety of fields, machine learning, deep learning, computer vision, plant pathology, and agronomy. For example, developing semantic Classificaton models involves high-level mathematics (linear algebra, K3) and state-of-the-art ML theory (K4). Competency in engineering design (K5/K6) is necessary in order to realize the hardware integration and system architecture. asserts that EP1 problems "cannot be solved without proper engineering skills" across K3–K6 and K8, which our project specifically needs (we also utilize K8 through the use of contemporary research).

EP2 Incompatible Demands:

Accuracy vs speed vs cost need to be traded off. Farmers require high-disease-identification capability (high accuracy), but the device has to be low-power and low-cost. Models have to be made sophisticated (enhancing accuracy) that oppose latency (needs simplicity). Other incompatibles are usability (user-friendly interface for non-technologists) vs. feature-rich (analysis-intense). EP2 challenges are extensive or incompatible technical challenges that may be noted within the following hardware vs. software trade-offs as well as stakeholder requirements.

EP3 Depth of Analysis:

There is no simple off-the-shelf solution, image data need to be post-processed and new ML pipelines created. It involves abstract modeling of leaf syndromes and new algorithms (e.g., neural net merging). EP3 problems "demand abstract thinking and creativity." An example, distinguishing highly similar diseases may involve high-level feature extraction. We perform thorough analysis on datasets, network topologies, and error states in order to meet this need.

EP4 Familiarity with Issues:

General categorization of plant diseases is great, although Luffa aegyptiaca is not so common, and application of other semantic models to it is novel. Therefore most of the less common problems we have (local growth conditions, local disease patterns) are not so common elsewhere. Less common problems like these need to be tackled by tests and researches instead of conventional methods.

EP6 Stakeholder Engagement:

There will be a number of project stakeholders farmers, agronomists, computer programmers, and possibly agro-industry partners. They will all have conflicting requirements (e.g. ease of use, acceptability to the regulator, payback). This is mirrored in EP6's "diversified groups with vastly differing requirements". We will need to get the users to give feed back (via field trials) and deploy the system so that it serves stakeholders' goals (e.g. food security for public good or private benefit).

EP7 Interdependence:

Many interdependent variables: data capture (hardware, camera), data processing (neural network models), user interface, comms layers. One has an influence on the others (e.g., change of model has an impact on compute load, power consumption). EP7 (Interdependence is being addressed here) says "many component parts or subproblems". Sensor, AI, and network integration suggests hardware, software, and domain expertise co-ordination, which is fitting.

Mapping with Knowledge Profile for EP1

Table 5.2: Mapping with knowledge Profile.

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

5.4.1.2 Justification for Knowledge Profile Mapping (linked to EP1)

K3 Engineering Principles:

Lower level principles of engineering (physics, maths, algorithms) are needed. Likewise, linear maths concepts of convolution operations and optimisation (stats) provide us with the basis for our DL,ML algorithms. K3 allows us to reason system behavior bottom-up.

K4 Specialist Knowledge:

Specialist knowledge must exist in the field. We apply specialist knowledge of computer vision, deep learning theory, and plant pathology (disease symptom). K4 is more advanced ML methods and crop science required to tailor the solution based on Luffa.

K5 Engineering Design:

System design data comes into solution development complete. It is data flow, UI, and hardware integration during software development. K5 renders us skilled to design the model, camera, and IoT network parts and interfaces of the product confidently.

K6 Practice of Engineering:

Engineering capabilities utilized (prototyping, programming, testing) are called upon. Iteration and deployment of the models and devices fall under K6. Engineering practice (i.e. use of CI/CD, use of Git) is used to build a working system.

K8 Research Literature:

It is evidence-based, we are utilizing and applying recent literature in leaf diagnosis of plant diseases. K8 is concurrently developed alongside published literature, e.g., using deep-learning in leaf classification. EP1 relies on published literature review

5.4.2 Engineering Activities

EA1 Range of re-sources	EA2 Level of Interaction	EA3 Innovation	EA4 Consequences for society and environment	EA5 Familiarity
✔	✔	✔	✔	✔

Table 5.3: Mapping with complex engineering activities.

5.4.2.1 Justification for Engineering Activities Mapping

EA1 Resource Intensity:

We exhaust massive amounts of resources: human, equipment (computer, camera, sensors), data (sinks of pictures and money. In short, according to EP1–EP7 projects, "entail consumption of a number of resources." We exhaust massive amounts of human, material and capital resources.

EA2 Interconnectedness Intensity:

The task is debugging challenging-to-debug interactions: hardware–software integration (e.g., camera drivers and DL code), and user–system interaction (e.g., farmer workflow). EA2 so far has "serious problems resulting from interactions between extensive issues", which we tackle through iterative prototyping and stakeholders' feedback.

EA3 Innovation:

Our approach is new, we integrate semantic classification models in a novel fashion for a web application. EA3 is "novel application of engineering principles", i.e., in terms of new breakthroughs in the architecture of neural networks and many models into new combinations to yield improved outcomes on plant disease data.

EA4 Implications for Society/Environment:

Make a project your gateway to neural networks.Application software implications are serious since it affects food and environmental safety. EA4 are situations that "have significant implications in a wide variety of circumstances." Correct identification, for instance, would prevent loss of crops (social advantage) and reduce pesticides (environmental advantage), while errors would be expensive in yield (implying the importance of reliability).

EA5 Familiarity:

We generalize beyond current experience, applying Deep Learning to a specialty crop (Luffa) is not current practice. EA5 is the reverse of "generalize beyond current experiences by using principles-based methods". We borrow AI/vision ideas and apply them to a new task and learn how to do it as we go.

5.5 Summary

Here in this present chapter, we explained why we proposed developing a semantic AI model for Luffa disease diagnosis as an engineering problem. We selected and stored the software libraries and hardware boards intentionally and used them in harmony with their nature (PyTorch for quick prototyping). We've had past large-scale effects too: response has clear positive effect on sustainable agriculture by way of increased output and reduced input (to SDGs hunger elimination and sustainability), but we limit ethical risk of bias and privacy by design constraint. Our economic analysis also shows cost-effectiveness within budget and where project can also reasonably derive revenues (e.g. from device or service sale), hence the project is financially viable. Lastly, by allocating the project against the requirements of the complex problem (EP1–EP7) and the engineering abilities (K3–K8, EA1–EA5), we can confidently announce this project to be a paradigm case of a complex engineering problem: it demands profound technical expertise, multi-criteria decision-making, innovative thinking, and social impact consideration. When we add the resultant total conformity to specifications, the impact analysis, the finance development, and the charting of complexity all together forms the holistic picture of project engineering intensity. We have made it safe (to appropriate IEEE/ISO standards), ethical and useful (to nature and farmer), economical, and brain-stimulating. This high level of scrutiny would make sure that the end product would be good enough to be offered as a capstone engineering project and has a strong enough foundation to rollout on.

Chapter 6

Conclusion

This chapter provides an overview of the project's key findings, contributions, and results. It also explains the limitations that were faced while conducting research and proposes possible avenues for future research

6.1 Summary

This "A Semantic Classification Method on Luffa Aegyptiaca Leaf Disease Detection with Different Models" research was conducted with the purpose to solve an extremely real problem of modern agriculture an early, affordable, accurate diagnosis of sponge gourd (Luffa aegyptiaca) leaf disease. Since crop disease has a devastating effect on crop yield, food safety, and farmers' livelihood particularly in a country like Bangladesh it was the goal of this research to fill the gap between deep learning technology and actual agricultural requirements. The approach applied a hybrid AI approach that employed semantic segmentation as well as deep convolutional neural networks to increase the accuracy in disease classification through concentration of model attention on the area of disease-infected area on the leaf. Through experimentation with some of the commonly used pretrained deep learning models such as ResNet50, InceptionV3, VGG19, ResNet152V2, and a custom CNN, the project discovered ResNet50 to be the most accurate and computationally most affordable model that can be used to classify five of the most important disease classes: Alternaria Leaf Spot, Angular Leaf Spot, Downy Mildew, Mosaic Virus, and Healthy leaves. Model selection will be dependent on experimental results. Image data pipeline was constructed with meticulous annotation, augmentation, and normalization processes for model resilience. UNet and DeepLabV3+ semantic segmentation allowed pixel-level attention, where the system would amplify the diseased tissues at a higher level and suppress background noise—this directly improved prediction accuracy and interpretability. Since the satisfactory performance measures in terms of accuracy, precision, and F1-score were very good, the model was deployed on a web application using Flask for online use and end-user accessibility. The computer application allows farmers, agricultural officers, or researchers to input leaf pictures and receive direct results of leaf disease diagnosis based on a simple and minimalist interface. The application was also proven to be seriously tested offline as well as online and would be deployable even in a resource-poor or rural environment.

6.2 Limitation

Dataset size and heterogeneity: Even if a really humongous dataset has been collected, within every class there are only few images, which is bound to leave generalizability to every corner and nook of condition. Environment heterogeneity: A model performance could be affected when test images are shown under various lights, weather, or alternate backgrounds apart from training environments.

Device constraint: Only single small models can fit onto Raspberry Pi; more detailed and computationally heavy models would be too hardware-intensive to run on its components.

Model bias: In case of overlap of symptom images or class imbalance, some disease classes will be mis-classified under certain conditions.

6.3 Future Work

Besides ongoing improvement of the system's efficiency and practicability, there are a number of research directions proposed. Firstly, the data set must be enriched with a larger quantity and variety of training images under various geographical conditions and seasons in order to further improve the robustness of the model. Second, the current web application would be cross-ported into a cross-platform mobile application so that it would be even more conveniently used in the field. The back-end will be cross-ported from Google Cloud or AWS to cloud for scalable and real-time computation. The model will undergo a series of optimizations such as pruning, knowledge distillation, and quantization to release space and allow the model to be deployed on edge devices. The interface will be multi-lingual in local languages and Bangla for end-user convenience. Apart from this, the system will also have an early disease detection module in the form of hyperspectral or multi-modal imagery for disease detection at their pre-symptomatic phase. Lastly, the system will be employing IoT-based sensor networks for self-service field monitoring and environmental data collection in real time

References

[1] M. T. Islam et al., “Luffa Aegyptiaca Leaf Diseases Dataset,” Mendeley Data, 2024. [Online]. Available: https://data.mendeley.com/datasets/nym8bw5hr6.

[2] M. U. Mojumdar et al., “Image Dataset for Disease Detection in Sponge Gourd,” Mendeley Data, 2024. [Online]. Available: https://data.mendeley.com/datasets/rx7mr3r5yg.

[3] DIU Contributors, “Automated Disease Detection in Luffa Using Vision Transformers,” Daffodil International University Institutional Repository, 2024. [Online]. Available: http://dspace.daffodilvarsity.edu.bd:8080/handle/123456789/13501.

[4] S. P. Mohanty et al., “Deep Learning-Based Leaf Disease Detection,” Agronomy, vol. 12, no. 10, p. 2395, 2022, doi:10.3390/agronomy12102395.

[5] Y. Tian et al., “First Report of Anthracnose on Luffa cylindrica Caused by Colletotrichum fructicola and C. siamense in China,” Plant Disease, vol. 106, no. 12, p. 3212, 2022, doi:10.1094/PDIS-03-22-0546-PDN.

[6] P. Sharma et al., “Hyperparameter Optimization for Pre-Trained Models in Plant Disease Detection,” Agronomy, vol. 12, no. 10, p. 2395, 2022, doi:10.3390/agronomy12102395.

[7] S. B. Jadhav et al., “Transfer Learning for Soybean Disease Detection,” Agronomy, vol. 12, no. 10, p. 2395, 2022, doi:10.3390/agronomy12102395.

[8] M. Bhange and H. A. Hingoliwala, “SVM-Based Pomegranate Leaf Disease Detection,” Computers and Electronics in Agriculture, vol. 207, p. 107725, 2023, doi:10.1016/j.compag.2023.107725.

[9] D. Al Bashish et al., “ANN for Cotton Leaf Disease Classification,” Computers and Electronics in Agriculture, vol. 207, p. 107725, 2023, doi:10.1016/j.compag.2023.107725.

[10] L. Wang et al., “PCA and BPNN for Crop Disease Classification,” Computers and Electronics in Agriculture, vol. 207, p. 107725, 2023, doi:10.1016/j.compag.2023.107725.

[11] T. N. Pham et al., “ANN for Early-Stage Mango Disease Prediction,” Computers and Electronics in Agriculture, vol. 207, p. 107725, 2023, doi:10.1016/j.compag.2023.107725.

[12] S. Thomas et al., “Hyperspectral Imaging for Potato Late Blight Detection,” Computers and Electronics in Agriculture, vol. 207, p. 107725, 2023, doi:10.1016/j.compag.2023.107725.

[13] M. Genaev et al., “ML for Early Wheat and Barley Disease Detection,” Frontiers in Plant Science, vol. 14, p. 1158933, 2023, doi:10.3389/fpls.2023.1158933.

[14] J. Liu et al., “CNN and DBN for Subtle Crop Disease Symptoms,” Frontiers in Plant Science, vol. 14, p. 1158933, 2023, doi:10.3389/fpls.2023.1158933.

[15] S. Kaur and S. Sharma, "Deep CNN for Lesion Segmentation in Field Conditions," Frontiers in Plant Science, vol. 14, p. 1158933, 2021, doi:10.3389/fpls.2023.1158933.

[16] A. Siddiqua et al., "Mask R-CNN for Lesion Localization in Crops," Frontiers in Plant Science, vol. 14, p. 1158933, 2022, doi:10.3389/fpls.2023.1158933.

[17] BananaLSD Dataset, "Banana Leaf Spot Disease Dataset," Mendeley Data, 2025. [Online]. Available: https://data.mendeley.com/datasets/khvpypw8bs.

[18] ACO-CNN Model, "Ant Colony Optimization with CNN for Plant Disease Detection," MDPI Agronomy, vol. 13, no. 8, p. 2144, 2025, doi:10.3390/agronomy13082144.

[19] Swin Transformer Study, "Potato Disease Detection Using Swin Transformer," MDPI Agronomy, vol. 13, no. 8, p. 2144, 2025, doi:10.3390/agronomy13082144.

[20] S. M. Hassan et al., "Novel CNN Architecture for Leaf Disease Detection," MDPI Agronomy, vol. 13, no. 8, p. 2144, 2025, doi:10.3390/agronomy13082144.

[21] Md. T. Islam, "Salinity-resilient crop health monitoring: automated disease detection in luffa Aegyptiaca leaves using Vision Transformer & CNN," Jan. 25, 2024. http://dspace.daffodilvarsity.edu.bd:8080/handle/123456789/13501

[22] S. Jiang et al., "A Real-Time Detection and Maturity Classification method for Loofah," Agronomy, vol. 13, no. 8, p. 2144, Aug. 2023, doi: 10.3390/agronomy13082144.

[23] A. J, J. Eunice, D. E. Popescu, M. K. Chowdary, and J. Hemanth, "Deep Learning-Based leaf disease detection in crops using images for agricultural applications," Agronomy, vol. 12, no. 10, p. 2395, Oct. 2022, doi: 10.3390/agronomy12102395.

[24] (2023). DSViT: Dynamically Scalable Vision Transformer for Remote Sensing Image Segmentation and Classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, doi: 10.1109/jstars.2023.3285259

[25] (2023). Cascaded Convolution-based Transformer with Densely Connected Mechanism for Spectral–Spatial Hyperspectral Image Classification. IEEE Transactions on Geoscience and Remote Sensing, doi: 10.1109/tgrs.2023.3275871

[26] Akbar, W., Soomro, A., Ullah, M., Inam Ul Haq, M., Ullah Khan, S., & Ali Shah, T. (2023). Performance Evaluation of Deep Learning Models for Leaf Disease Detection: A Comparative Study. 2023 4th International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), 01–05. https://doi.org/10.1109/iCoMET57998.2023.10099223

[27] Arun, K., Sharma., Nishchal, K., Verma. (2023). A Novel Vision Transformer with Residual in Self-attention for Biomedical Image Classification. arXiv.org, doi: 10.48550/arXiv.2306.01594

[28] Bin, Li., E., Ouyang., Wenjing, Hu., Guoyun, Zhang., Lina, Zhao., Jianhui, Wu. (2022). Multi granularity vision transformer via semantic token for hyperspectral image classification. International Journal of Remote Sensing, doi: 10.1080/01431161.2022.2142078

[29] Bing, Li., Qi-wen, Wang., Jianxing, Liang., En-Ze, Zhu., Rongfu, Zhou. (2023). SquconvNet: Deep Sequencer Convolutional Network for Hyperspectral Image Classification. Remote sensing, doi: 10.3390/rs15040983

[30] Boli, Zhang., Zuheng, Ming., Wei, Feng., Yaqian, Liu., Liang, He., Kaixing, Zhao. (2023). MMFormer: Multimodal Transformer Using Multiscale Self-Attention for Remote Sensing Image Classification. arXiv.org, doi: 10.48550/arXiv.2303.13101

[31] Boukabouya, R. A., Moussaoui, A., & Berrimi, M. (2022). Vision Transformer Based Models for Plant Disease Detection and Diagnosis. 2022 5th International Symposium on Informatics and Its Applications (ISIA), 1–6. https://doi.org/10.1109/ISIA55826.2022.9993508

[32] Chen, Xu., Xiubao, Sui., Jian, Liu., Yuhang, Fei., Liping, Wang., Qian, Chen. (2023). Transformer in optronic neural networks for image classification. Optics and Laser Technology, doi: 10.1016/j.optlastec.2023.109627

[33] Chougui, A., Moussaoui, A., & Moussaoui, A. (2022). Plant-Leaf Diseases Classification using CNN, CBAM and Vision Transformer. 2022 5th International Symposium on Informatics and Its Applications (ISIA), 1–6. https://doi.org/10.1109/ISIA55826.2022.9993601

[34] Fujian, Zheng., Wei, Zhou., Yong, Huang. (2023). A Lightweight Dual-Branch Swin Transformer for Remote Sensing Scene Classification. Remote sensing, doi: 10.3390/rs15112865

[35] 12. Guanqun, Wang., He, Chen., Liang, Cheng., Ying, Zhang., Shang, Liang, Zhang., Tong, Zhang., Hao, Dong., Peng, Gao. (2023). P2FEViT: Plug-and-Play CNN Feature Embedded Hybrid Vision Transformer for Remote Sensing Image Classification. Remote sensing, doi: 10.3390/rs15071773

Match Groups

- 217 Not Cited or Quoted 16%
  Matches with neither in-text citation nor quotation marks
- 5 Missing Quotations 0%
  Matches that are still very similar to source material
- 6 Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

- 11% Internet sources
- 9% Publications
- 14% Submitted works (Student Papers)

Top Sources

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

1	Submitted works	
	Daffodil International University on 2024-12-28	4%
2	Internet	
	dspace.daffodilvarsity.edu.bd:8080	1%
3	Internet	
	www.mdpi.com	<1%
4	Internet	
	nepis.epa.gov	<1%
5	Publication	
	V. Sharmila, S. Kannadhasan, A. Rajiv Kannan, P. Sivakumar, V. Vennila. "Challeng...	<1%
6	Internet	
	lcjstem.com	<1%
7	Submitted works	
	University of Hertfordshire on 2025-01-06	<1%
8	Submitted works	
	University of Hong Kong on 2025-04-17	<1%
9	Internet	
	thesai.org	<1%
10	Submitted works	
	Asia Pacific University College of Technology and Innovation (UCTI) on 2024-09-30	<1%