Decentralized Deep Learning for Rose Leaf Disease Detection: A Federated CNN Approach

By
Pithila Mostofa Sowa
ID: 213-15-4477

Syful Islam
ID: 213-15-4336

FINAL YEAR DESIGN PROJECT REPORT

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

Supervised by

Md. Firoz Hasan

Designation:
Senior Lecturer
Department of Computer Science and Engineering Daffodil International University

Co-Supervised by

Mr. Md. Asaduzzaman

Designation:
Assistant Professor
Department of Computer Science and Engineering Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY
Dhaka, Bangladesh

September 17, 2025

APPROVAL

This Project titled **"Decentralized Deep Learning for Rose Leaf Disease Detection: A Federated CNN Approach"** submitted by Pithila Mostofa Sowa, ID No: 213-15-4477 And Syful Islam, ID No: 213-15-4336 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **17 September, 2025**.

BOARD OF EXAMINERS

Dr. Sheak Rashed Haider Noori **Professor and Head** Department of Computer Science and Engineering Faculty of Science & Information Technology Daffodil International University	**Chairman**
Dr. Md. Zahid Hasan **Associate Professor** Department of Computer Science and Engineering Faculty of Science & Information Technology Daffodil International University	**Internal Examiner**
Samia Nawshin **Assistant Professor** Department of Computer Science and Engineering Faculty of Science & Information Technology Daffodil International University	**Internal Examiner**
Dr. Md. Arshad Ali **Professor** Department of Computer Science and Engineering Hajee Mohammad Danesh Science & Technology University	**External Examiner**

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Md. Firoz Hasan**, **Senior Lecturer**, Department of Computer Science and Engineering, Daffodil International University. We also declare that neither this projectnor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:

15.9.25

Md. Firoz Hasan
Senior Lecturer
Department of Computer Science and EngineeringDaffodil International University

Co-Supervised by:

15.9.25

Mr. Md Assaduzzaman
Assistant professor
Department of Computer Science and EngineeringDaffodil International University

Submitted by:

Pithila 17.09.25

Pithila Mostofa Sowa
Student ID: 213-15-4477
Department of Computer Science and EngineeringDaffodil International University

Syful Islam 17.9.25

Syful Islam
Student ID: 213-15-4336
Department of Computer Science and EngineeringDaffodil University

ACKNOWLEDGEMENTS

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

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

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

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

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

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

ABSTRACT

Rose is considered useful in agriculture and horticulture. Their leaves are however highly prone to illness such like black spot, mildew and mosaic virus. Such diseases may reduce the yield as well as quality. The conventional methods of detection are slow and they might require specialist expertise. This renders them not practical to farmers particularly in the countryside. Yesterday our project is aimed to create a smart solution that would keep all data confidential and apply deep learning and Federated Learning (FL) to effectively and safely identify rose leaf diseases. The models employed by the system to make the system fast and accurate include VGG-19, Effective Net-v2, and MobileNet V2. The farmers are able to take leaf images or upload them using their mobile phone. Training is actually done local to the device the raw data is never transferred outside of the device. The learning results are collected by a central server and this makes the model scalable, reliable and able to be applicable in other farms. It is a system that can be used on mobile devices and low-power edge devices. It is convenient, simple to use and useful in real world farming. The collaboration between AI, privacy, and usability aims to assist farmers to discover the diseases at their early stages, secure the quality of crops, and create more sustainable agriculture.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

The rose flowers are very commercial and attractive. However, it is susceptible to several diseases of the leaf that include black spot, powdery mildew, and mosaics virus. These diseases significantly degrade the quality of the flowers and might lead to serious losses in the quantity of the crop, which causes farmers and growers losses. Documenting of the diseases and diagnosis used to be carried out primarily through manual examination, which is time-consuming, most labor intensive, and prone to error. This is the reason why it cannot move rapidly. New developments provided by deep learning and computer vision suggest solutions. rapid and efficient detection of the diseases. Having said that, almost all available methodologies gather data centrally, and the risks associated with them are severe in terms of privacy. The farmers and farm businesses might not be willing to place sensitive crop images on the cloud due to fear that the information might be misapplied- or even secrets stolen. In this work, to solve this issue we suggest a FL based rose leaf disease detection system. This two-way environment of learning permits a number of users to train one common model collectively without distributing the unique illustrations, which can accomplish the compromise concerning the data eventually as well as the shared knowledge. In our system, there is an adoption of light-weight models, including MobileNet v2, and EfficientNet, in promoting efficient deployment on mobile and low-power edge devices. This ensures that even agrnomists in the rural areas can obtain good predictions without having to spoil with high-end device. by-Vgg-19 with that, they have enhanced feature extraction to achieve better accuracies. This feature breeds confidence in the system as well as system predictions are more transparent. We will equip farmers with early disease detection with precision by using AI with the privacy feature that preserves their privacy, deploying it directly to farmers that can be easy to use. It seeks to reduce crop losses and over use of chemicals, sustainable agriculture practices-and promote sustainability

1.2 Motivation

Agricultural illnesses such as diseases that afflict high-value crops, like roses, are a major problem and interfere with agricultural output. The process is error prone and slow, but early detection of such diseases would help prevent some heavy damages. Although deep learning is an effective way to offer an accurate detection, the method can have numerous privacy problems when relying on centralized data to the farmers. Federated Learning is provided as a solution to collaborative model training that does not involve sharing raw data, and is realistic to implement in the real world on resource-constrained devices. By addressing the particular need for accurate disease identification and a broader concern over data in agriculture, the research is tackling both. And it is advantageous to the man of small means in giving him reliable tools that are simple and cheap. This intersectional project enables the expansion of safer, more sustainable and technology-driven agriculture by merging AI innovation with real world farming needs.

1.3 Objectives

The primary objectives of this research are:

- Build a smart system to detect diseases in rose leaves while keeping farmers' data safe.
- Use Federated Learning to train models together without sharing raw images.
- Apply deep learning models:
    - MobileNet v2 and EfficientNet for lightweight, efficient, and high-accuracy performance.
    - VGG-19 for feature extraction and model comparison.
- Design the system to be fast, storage saving, and perform well in actual farming conditions where the resources for internet and computing are restricted.

1.4 Methodology

This study follows an experimental approach to develop a decentralized deep learning system for detecting diseases in rose leaves using federated learning. The system employs MobileNet v2, EfficientNet, and VGG-19, combining lightweight efficiency with strong feature extraction. Models are trained across multiple simulated client devices using the Federated Averaging algorithm, keeping data local and private while building a shared global model. Attention mechanisms like CBAM help the models focus on diseased areas, and the system is designed for deployment on mobile or low-resource edge devices.

To make the system practical for real-world use, part of the dataset was collected directly from a garden farm, capturing high-quality images of healthy and diseased rose leaves under natural lighting and backgrounds using a smartphone. Publicly available images were also added to increase diversity and balance.

All images were preprocessed, resized, and augmented through flipping, rotation, and brightness adjustments. The dataset was then divided among virtual clients to simulate a federated learning setup, reflecting how data would be distributed across different farms or institutions in practice.

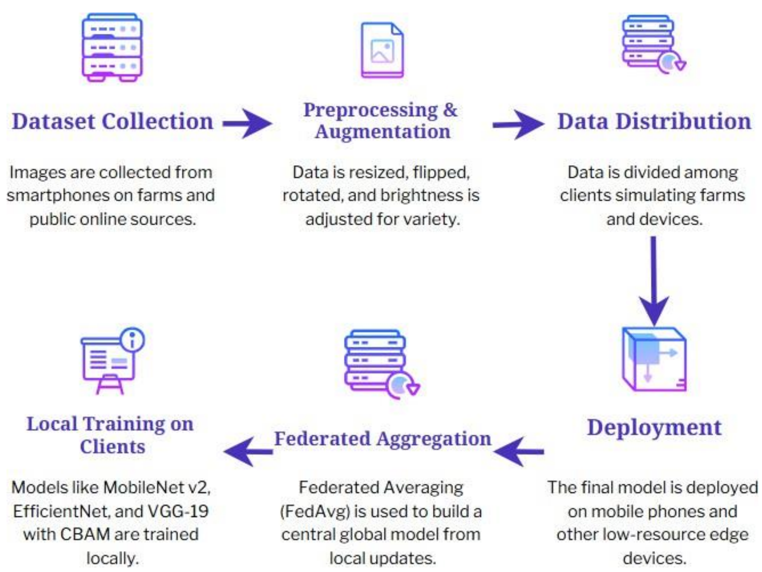

Figure 1.4.1: Schematic representation

1.5 Project Outcome

1. Accurate and Privacy-Safe Disease Detection

The system aims to identify rose leaf diseases like black spot, mildew, and mosaic virus with high precision. By leveraging federated learning, it makes the raw images privacy preserving, such that farmers or organizations can contribute to improving the model without sharing private data.

2. Lightweight and Fast Performance

It uses MobileNet v2 and EfficientNet at its core and is optimized for running efficiently on embedded low-power devices (smartphones, small processors). This is advantageous for farmers in remote area with poor computing resources.

3. Flexible and Scalable Solution

The model is capable of dealing with various data from different farms, thus can be applied to different regions. Its adaptable structure allows for the deployment in a large number of farms and big farmer networks.

4. Supporting Sustainable Farming

Early detection and accurate identification of diseases contributes to the reduction of crop loss and the utilization of chemicals. This not only reduces costs for farmers, but also supports ecologically friendly and sus-trainable agriculture.

5. Academic and Research Contribution

The study shows that, by combining MobileNet v2, EfficientNet and VGG-19 for a lightweight leaner under the federated learning framework in agriculture, useful computer vision models can be achieved in a cost-effective manner. It serves as a benchmark for future research of plant disease detection and privacy-friendly AI solutions.

1.6 Organization of the Report

This report extends as follows:

Introduction: The introduction section defines the aim and scope of the project, discusses the significance of rose plants in agriculture, and presents the problem of identifying leaf diseases and preserving privacy with the help of federated learning.

Background and Literature Review: The section takes a review of the previous research on plant disease detection, deep learning approach, and privacy-protecting approaches. It evaluates existing strategies and identifying some gaps and how the proposed system will solve these problems.

Research Methodology: This section describes the system design, the functional, non-functional requirements, the federated CNN architecture, the model choice (MobileNet v2, EfficientNet, VGG-19,data management and explainability and deployment of mobile and low-resource devices.

Implementation: In this section, the collection of the dataset is described on the garden farms and public sources including preprocessing and augmentation, federal learning configuration, model training, and addition of attention mechanisms to focus on areas with diseased leaf with use of attention mechanisms.

Results and Analysis: The section does show the results of the trained models in terms of their accuracy, their ability to preserve privacy, and visualization outputs. It will compare the performance with the conventional techniques to demonstrate the performance, speed, and applicability of this system to low-power devices.

Complex Engineering Problem and Activities: At this stage, the project is related to problem-solving criteria and knowledge profile of complex engineering problems. It outlines such engineering activities like innovation, resource management.

Conclusion and Future Work: This part sums up the success of the project, such as successful disease detection, which is effective and preserves privacy, scalability, and providing a long-term contribution to sustainable farming. It also talks about what its measures cannot do and what to do next in research as to associate the model.

Chapter 2

Background

2.1 Introduction

This paper is based on a collection of research on the theory of deep learning and federated learning applied in the area of detecting plant diseases, specifically the work Decentralized Deep Learning for Rose Leaf Disease Detection: A Federated CNN Approach. Past research has examined different convoluted neural network (CNN) designs to identify diseases of leaf. These researches were conducted in order to have accuracy as well as efficiency in computation. MobileNet v2 and EfficientNet are lightweight models that are suitable on machines with weak processing capabilities. They give quick and accurate predictions which can be used in the real world. VGG-19 and another deep architecture, in contrast, serve better features extraction. They can select the marginal properties of diseased leaves of the rose and enhance the accuracy of the classification. Moreover, attention mechanism which has been proposed in recent works like the Convolutional Block Attention Module (CBAM) has also been introduced. This allows models to focus on the most relevant parts of the leaf to the problem and it makes them more understandable. Federated learning has been applied in solving the privacy concerns. By doing this, it is possible to train a single model across several farms or clients without raw image sharing conducted on their part. Such approach ensures confidential data and access to different datasets. More so, explainable AI approaches are provided to demonstrate how the model takes into account the leaf areas. This enhances openness and contributes self-confidence to end-users. To sum up, these advancements prove that lightweight and deep CNNs with the help of Attention Modules, FL, and XAI can be useful, safe, and useful when working on the creation of a new system to detect rose leaf disease. This coincides effectively with the aims of this work.

2.2 Literature Review

Author (s)	Year	Title	Methodology	Key Findings
Thangaraj et al.	2021	Transfer Learning with EfficientNet for Agricultural Leaf Disease Classification	Applied transfer learning with EfficientNet	Found that EfficientNet models provide better accuracy and computational efficiency compared to traditional CNNs for classifying leaf diseases.
Wei et al.	2023	Privacy-Preserving Medical Imaging via Federated Learning	Federated learning with privacy-preserving techniques	Developed federated learning methods that keep sensitive image data decentralized while maintaining high accuracy.
Antico et al.	2022	Federated Learning for Crop Disease Classification: A Case Study on Maize	Federated CNN approach	Showed that federated learning can effectively detect maize diseases, even with non-uniform (non-IID) datasets.
Kabala et al.	2023	Image-based crop disease detection with federated learning	CNN and Vision Transformer models with federated learning	Explored how to use federated learning for classifying crop diseases through image analysis. This work relied on open access image datasets.
Kumar et al.	2024	Federated Transfer Learning for Rice-Leaf Disease Classification	Federated learning with transfer learning	Proposed a federated transfer learning method for classifying rice leaf diseases. This approach keeps data private and improves model performance.

Chorney et al.	2025	Federated Learning for Heterogeneous Multi-Site Crop Disease Classification	Federated learning across multiple sites	Showed how farms with very different datasets can work together using federated learning to develop a classifier for identifying and managing rice crop diseases.
Hari et al.	2025	Adaptive knowledge transfer using federated deep learning for plant leaf disease detection	Federated deep learning with intelligent weight transferring	Introduced a method designed for detecting plant leaf diseases in federated deep learning. This approach improves model accuracy by using adaptive knowledge transfer.
Mzoughi et al.	2023	Deep Learning for Plant Disease Detection: A Review	Review and comparative analysis	Summarized various deep learning approaches for plant disease detection, emphasizing the effectiveness of CNN-based models in agriculture.

Table 2.2.1: Summary of Literature Reviewed

2.2.1Similar Applications:

1. Transfer Learning for Leaf Disease Classification. Thangaraj et al. (2021) applied EfficientNet for agricultural leaf disease detection, demonstrating improved accuracy and computational efficiency compared to traditional CNNs.

2. Privacy-Preserving Medical Imaging via Federated Learning. Wei et al. (2023) showed how federated learning can keep data private while achieving high prediction accuracy.

3. Federated CNN for Crop Disease Classification. Antico et al. (2022) effectively used federated learning to detect maize diseases, even with non-IID datasets.

4. Image-Based Crop Disease Detection with Federated Learning. Kabala et al. (2023) combined CNN and Vision Transformer models with federated learning to classify crop diseases using open-access image datasets.

5. Federated Transfer Learning for Rice Leaf Disease Classification. Kumar et al. (2024) proposed federated transfer learning to improve rice leaf disease classification while keeping data private.

6. Multi-Site Federated Learning for Crop Disease Detection. Chorney et al. (2025) conducted federated learning across farms with different datasets to create accurate rice disease classifiers.

7. Adaptive Knowledge Transfer in Federated Deep Learning. Hari et al. (2025) introduced adaptive weight transfer in federated deep learning to improve plant leaf disease detection accuracy.

8. Review of Deep Learning for Plant Disease Detection. Mzoughi et al. (2023) summarized CNN-based methods for plant disease detection, highlighting their effectiveness in agriculture.

2.3 Gap Analysis

Although there has been rapid advancements in the use of deep learning methods to find diseases in the leaves of plants, some difficulties are yet to be overcome through further studies and refinement. The analysis presented in the literature review includes gaps that indicate where more accuracy, privacy, scalability, and real-time applications can be observed in various agricultural adjustments.

Lightweight CNN Efficiency and Usability: Lightweight CNN models, like MobileNet v2, are fast to write but might not work well in the multi-disease cases. More elaborative models, such as VGG19 and EfficientNet, have high accuracy but are more resource-hastry that is why they can hardly be implemented on low power devices.

Scalability and Data Distribution: Federated learning allows various farms or clients to train models without sharing their raw data. However, working with non-IID (non-uniform) datasets is still a challenge. Studies like Antico et al. (2022) and Kabala et al. (2023) indicate that model performance can decline when data distribution is uneven among clients.

Implementation and Usability Insights: Setting up federated learning systems in real-world agricultural settings is tricky due to demands on the network, issues with model aggregation, and differences among devices. Simplified frameworks and adaptive weight transfer methods, like those suggested by Hari et al. (2025), can make these systems easier to use, but practical integration is still a problem.

Generalizability Across Varied Leaf Conditions: Most studies concentrate on single-crop or single-disease datasets. Current models often have trouble generalizing to multiple leaf diseases, seasonal shifts, or different rose cultivars, which shows a need for adaptable multi-class classification models.

Model Boundaries and Disease Categorization: Accurately distinguishing between similar diseases, such as black spot and mildew, is still a challenge. Using attention mechanisms (like CBAM) or combining multiple models can improve feature extraction and lessen misclassification.

Factors Driving Model Performance: Environmental factors, leaf age, and image quality greatly influence model accuracy. Current research does not always consider these variables; adding these factors could lead to better detection in real-world situations.

Model Boundaries and Disease Categorization: Clear cut distinction of the similar diseases like those related to black spot and mildew remains a problem. The several models thinkable with attention (or by clustering multiple) could ameliorate feature extraction and reduce misclassification.

Driving Model Relationships: There are strong relationships among these parameters because of the impact of environmental, leaf age, and image quality on the accuracy of a model. These variables are not necessarily taken into account by the current research; incorporation of these factors may result in improved identification in real world scenarios.

Data Diversity and Real-time Integration: The combination of data on different farms has the potential to enhance the accuracy of the model, yet most models are difficult to combine and update in real-time, particularly in those areas where network connectivity is limited.

Computational Efficiency: Deep CNNs are associated with higher accuracy, but they require large computational costs. Reduced weight models such as MobileNet v2 offer a tradeoff, yet there is still a pressing research objective to obtain both real-time and high-accuracy models.

2.4 Summary

The chapter 2 provides background to the study, "Decentralized Deep Learning to Rose Leaf Disease Detection: A Federated CNN Approach". It reviews current research on deep learning and federated learning used for detecting plant diseases. The chapter focuses on CNN architectures like MobileNet v2, EfficientNet, and VGG-19. These models balance computational efficiency with high accuracy. This method ensures privacy. Explainable AI techniques enhance transparency by showing important leaf regions. The literature points out gaps in managing non-IID datasets. It also highlights challenges in generalizing across different leaf diseases and rose varieties, dealing with noisy or unstructured field data, and achieving real-time deployment on low-powered devices. Combining lightweight and deep CNNs, attention mechanisms, federated learning, and explainable AI creates an efficient, secure, and practical strategy for detecting rose leaf diseases. This context informs the proposed research.

Chapter 3

Research Methodology

3.1 Methodology

The method for detecting rose leaf diseases starts with collecting images of leaves from various sources, including black spot, mosaic virus, mildew virus, and healthy leaves. Next, preprocessing steps are carried out, such as resizing, normalization, data augmentation, and encoding of categorical labels. These steps ensure high-quality and standardized input for supervised learning. Multiple CNN architectures, including MobileNet v2, EfficientNet, and VGG-19, are trained to achieve a good balance between efficiency and classification accuracy. Federated learning trains models across several clients without sharing raw data. It uses Federated Averaging to combine updates while keeping data private. The dataset is divided into training and testing sets. Models are evaluated based on accuracy, precision, recall, and F1-score, while visualizations help with understanding the results. Findings are presented using charts and confusion matrices. This approach ensures effective, practical, and privacy-conscious detection of rose leaf diseases, making it suitable for real-world use.

3.1.1 Overview

This chapter explains how the system helps farmers efficiently and securely detect rose leaf diseases. It focuses on common issues like black spot, mildew, and mosaic virus. The system uses several CNN models, including MobileNet v2, EfficientNet, and VGG-19, along with Federated Learning, which keeps raw images private on local devices. It provides visual feedback by highlighting affected areas. Farmers can capture or upload leaf images using mobile devices, and the system works with edge devices. It automatically updates the central model with client-side learning.
Key performance goals include speed, accuracy, strength, scalability, user-

friendliness, and support for low-power devices. The setup includes client nodes and a central server for aggregating data. It also features local image preprocessing and augmentation, Overall, the system is designed for practical use in rural areas. It makes disease detection simple, secure, and reliable for farmers.

friendliness, and support for low-power devices. The setup includes client nodes and a central server for aggregating data. It also features local image preprocessing and augmentation, Overall, the system is designed for practical use in rural areas. It makes disease detection simple, secure, and reliable for farmers.

3.1.2 Proposed Methodology/ System Design

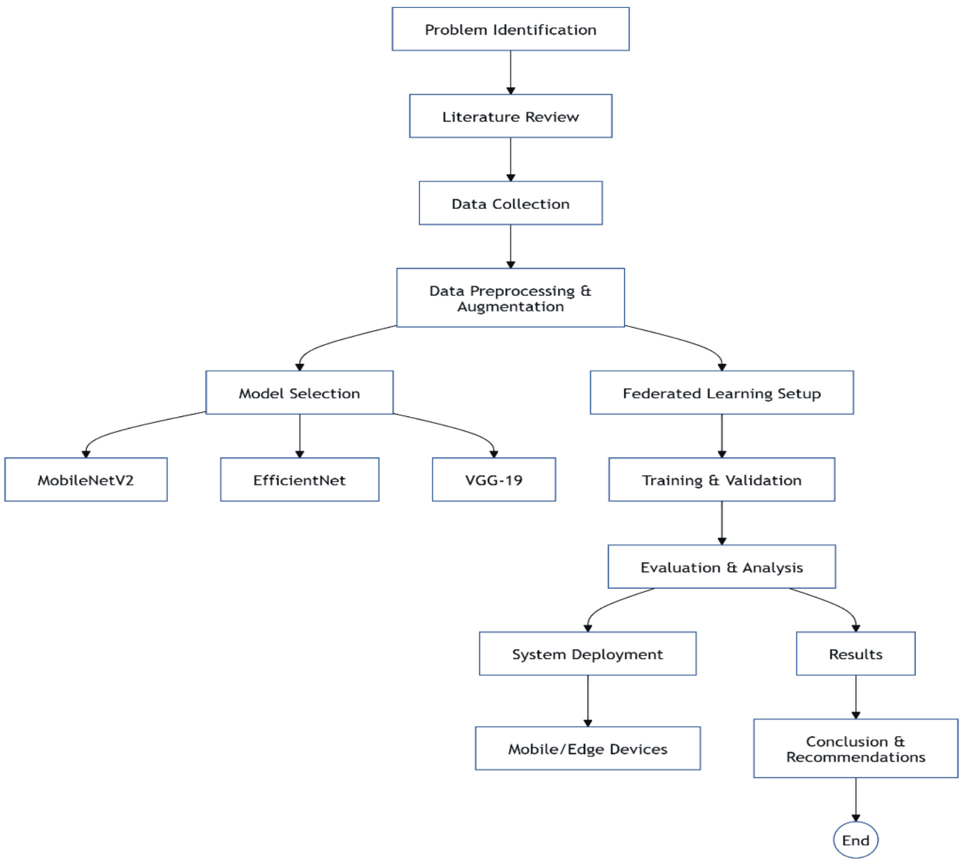

Figure 3.1.1: System diagram

1. **Problem Identification & Literature Review:** The diagram starts with Problem Identification and Literature Review, matching my paper's introduction. It defines the problem of rose leaf diseases and sets the context for a secure and efficient detection system.

2. **Data Collection & Processing**: The Data Collection and Data Preprocessing and Augmentation stages in the diagram reflect my paper's

focus on local image collection and the use of preprocessing and augmentation. This step is essential for training a strong model, as your requirements state.

3. **Model Selection:** The diagram directly names the models from your paper's Design Specification: MobileNetV2, EfficientNet, and VGG-19. Including these three specific models supports the diagram's goal of comparing different CNN architectures to identify the best fit for my needs.

4. **Federated Learning Setup:** The Federated Learning Setup stage in the diagram addresses my functional and non-functional needs for privacy and decentralized learning. It visually separates the architectural setup from individual model choices, showing it as a vital part of the methodology.

5. **Training & Validation:** The Training and Validation stage in the diagram combines the results from model selection and the federated learning setup. It demonstrates how the selected models are trained within the decentralized framework to achieve my aim of automatically updating the central model.

6. **Evaluation & Analysis:** The Evaluation and Analysis stage in the diagram focuses on my non-functional requirements for high accuracy and reliability. This analysis leads to two outcomes: system deployment and the presentation of results.

7. **System Deployment:** The System Deployment stage targets Mobile/Edge Devices and directly addresses my non-functional requirements for a lightweight, low-power system that works well in rural areas. This shows the practical application of your research.

8. **Results & Conclusion:** The final stages of Results and Conclusion and Recommendations align with the end sections of a research paper, where theproject's findings are summarized and suggestions for future work are made.

3.1.3 Functional and Nonfunctional Requirements

Functional Requirements (What the system should do):

1. Identify common rose leaf diseases, including black spot, mildew, and mosaic virus.
2. Train models using federated learning while keeping raw images private.
3. Support different CNN architectures for disease classification.
4. Provide visual feedback by highlighting the affected area on the leaf.
5. Allow farmers to upload or take leaf images directly from their mobile devices.
6. Enable easy integration with edge devices for field-level use.
7. Automatically update the central model by combining client-side learning results.

Non-Functional Requirements (How the system should perform):

1. Ensure fast and lightweight performance for real-time disease detection.
2. Maintain data privacy and security by not storing raw images centrally.
3. Achieve high accuracy and robustness, even with changes in lighting, or leaf position.
4. Be scalable to support multiple farms or larger distributed datasets.
5. Offer an easy-to-use interface for farmers with minimal technical skills.
6. Optimize for low-power mobile and edge devices to make it practical for rural areas.
7. Ensure reliability and fault tolerance in federated training and aggregation.

Design Specification:

Architecture:

A federated CNN setup with several client nodes and a central server for aggregation.

Data Handling:

Images collected locally will be preprocessed, augmented, and shared among clients.

Model Selection:

MobileNet v2 and EfficientNet will be used for efficiency, while VGG-19 will provide feature-rich accuracy.

Deployment:

This is designed for mobile and low-power edge devices to ensure access in rural areas.

3.1.4 Data Flow Diagram

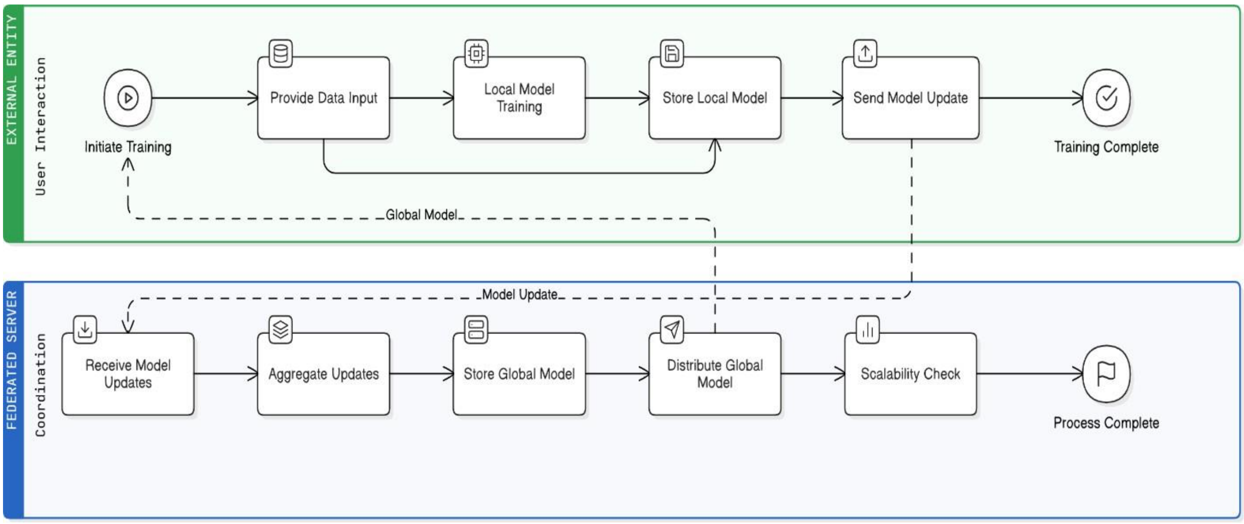

Figure 3.1.2: DFD Diagram

External Entity (User Interaction) Flow:

This top section, in green, shows the process on the client side.

1. **Initiate Training:** The process starts when a user or system begins training on their local device.

2. **Provide Data Input:** The user provides images of rose leaves. According to your paper's requirements, these images remain private on the local device.

3. **Local Model Training:** The local device trains a copy of the global model using its own private data. This step meets your requirement to train models using federated learning while keeping images private.

4. **Store Local Model:** The newly trained model, which now contains knowledge from the local data, is saved on the device.

5. **Send Model Update:** Instead of sending the raw images, the device sends the model update, which consists of the changes or weights from the local training, to the federated server. This is an important part of your privacy and security requirement.

6. **Training Complete:** The local training cycle is finished.

Federated Server (Coordination) Flow:

This bottom section, in blue, shows the central server's role in the federated aggregation process.

1. **Receive Model Updates:** The server gets the model updates from multiple client devices.

2. **Aggregate Updates:** The server combines the updates from all clients to create a new, improved global model. This step is essential to the federated learning method you described.

3. **Store Global Model:** The new global model that is formed is saved on the server.

4. **Distribute Global Model:** The server transmits the back the new global model to all client devices preparing them to the next training round.

5. **Scalability Check:** This step helps in your scalability need, which is that the system will be able to handle updates by more farms or more devices.

6. **Process Complete**: Federation training round is done.

3.2 Detailed Methodology and Design

The objective of the project is primarily to identify the rise leaf diseases with accuracy and to secure the data of farmers. We can achieve this by federation of deep learning modeled as CNNs. This process may be contrasted into three segments:

1. **Data Collection and Preprocessing**

   - Farmers take images of rose leaves with mobile devices.
   - Local preprocessing of these images is done. This involves the process of resizing, noise removal and normalization of analyzing their presence.

2. **Local Model Training (Federated Learning)**

   - Each device trains a CNN model (MobileNetV2, EfficientNet, VGG-19 using its own dataset.
   - Only the model parameters (weights) go to the central server, keeping the raw images private.

3. **Global Model Aggregation**

   - The central server collects updates from multiple devices to create an improved global model.

- The updated global model is then sent back to all devices, which improves disease detection accuracy over time.

Alternate Solutions Considered:

Centralized Deep Learning Approach

- All the images get uploaded on a central training server.
- Among the advantages are: It can be easier to control, and the convergence of the model may be fast.
- Cons: Privilege risks, non-scalable requirement and excessive bandwidth.
- Reason Science Rejected: It threatens the privacy of data and does not suit the decentralized countryside.

Classical Machine Learning (SVM, Random Forest)

- This method uses manually extracted features for disease classification.
- Pros: Lower computational requirement.
- Cons: Requires complex feature engineering, lower accuracy than CNNs.
- Reason Rejected: It cannot capture intricate leaf patterns effectively.

Hybrid Local-Cloud Approach

- In this model, some local preprocessing is done followed by training the whole model in the cloud.
- Pros: Reduces device computation.
- Cons: Requires partial data sharing, still depends on network connectivity.
- Reason Rejected: It partially compromises privacy and scalability.

3.3 Project Plan

This research is set to be mainly planned as a project whose primary goal is to develop a system that will guarantee privacy and detect an ailment in rose leaf by utilizing federated deep learning. The project begins with a history review of the relevant literature and specification of the system requirements. The next step is to collect and process pictures of healthy and disease leaves in order to analyze them. CNN models such as EfficientNet, MobileNetV2 and VGG-19 are taken. To implement a federated learning model, a federated training model is developed that enables a local training on the devices which transmit only model updates back to the central server. Global model is then environmentally faced with real-world datasets, in terms of accuracy, precision, and reliability. and lastly the conclusion is reached with the documentation of the methodology, results, and findings to complete the thesis. This provides a scalable and efficient and collaborative system that offers data privacy and accurate disease detection

3.4 Task Allocation

Tasks	Week														
	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40
Gathering healthy and diseased leaf images and making them ready for model training															
Teaching the CNN models on local devices and combining updates to create a stronger global model.															
Checki ng how well the models perform using accuracy, precision, and real-world datasets.															
Preparing the report, creatin g presentation slides, and submitting															

the work for evaluation														

3.5 Summary

The third chapter, Research Methodology, outlines a deep learning system on detecting rose leaf diseases and implies the emphasis on data privacy as a decentralized system. The primary algorithm is Federated Learning; the farmers store their data on a local device, and only updates of the models are uploaded to a central server. This approach to learning does not create a centralized and traditional machine learning solution, which makes the system grow and be secure. The efficient CNNs used in the methodology, such as MobileNetV2, EfficientNet, VGG-19, are also efficient. The chapter concludes on a note of a project plan which encapsulates steps that will be used to process data and offering the ultimate system analysis. This provides a practical solution that can be used in practice on low power applications.

Chapter 4

Implementation and Results

4.1 Environment Setup

In order to effectively put this project into effect, the appropriate hardware, software and network environment is required. The system will be configured to train CNN models at the edges of the user devices and carry out federated learning where the updates to models are aggregated at a central server and the raw data remains private.

1. Hardware Requirements:

Local Devices (Clients):

- **Processor:** Intel Core i5 or equivalent
- **RAM:** Minimum 8 GB
- **GPU:** NVIDIA GPU recommended for faster CNN training
- **Storage:** At least 100 GB to store images and model files

Central Server:

- **Processor:** Intel Core i7 or equivalent
- **RAM:** 16 GB or more
- **GPU:** NVIDIA GPU recommended for model aggregation
- **Storage:** Minimum 500 GB for storing model updates and logs

2. Software Requirements:

Operating System: Windows 10/11 or Linux (Ubuntu 20.04 recommended)

Programming Language: Python 3.8 or above

Libraries and Frameworks:

- Tensorflow / Keras - to create and run CNN models.

- TensorFlow Federated – for federated learning implementation
- OpenCV – for image preprocessing
- NumPy & Pandas – for data handling and manipulation
- Matplotlib / Seaborn – for visualizing results
- **Development Tools:**
  - **PyCharm** or **VS Code** for coding
  - **Jupyter** Workbook for experiment and test

3. Network and Connectivity:

- To facilitated communications of federated learning between local machine and the central server, a stable internet signal must be available.
- Secure communication protocols are essential to ensure that model updates are transmitted safely without exposing private data.

4.2 Train Model/Prototype Design

Powerful decentralized deep learning auxiliary with convolutational neural networks (CNNs) was appraquisite to recognize diseases in rose leaves. MobileNet v2, EfficientNet, and VGG-19 are the initial models of the system. This mix gives it light weight effectiveness combined with good feature pull out. These models were used to train in a federated learning setting, in which a set of simulated clients trained their models locally holding their data confidential. Such an approach enables the system to perform effectively on mobile and edge devices and be very accurate in detecting the diseases.

4.2.1 Data Preparation

The data was composed of superior photos of the leaves of roses taken in garden farms using smartphones. Images which were publicly available were also included so as to provide diversity and balance. Premedication of all the images undertaken included the resizing, normalization and the flipping, rotation and altering the brightness of the images, in essence, data augmentation. Once the preprocessing was done, the dataset was distributed to multiple virtual clients to replicate a federated learning system. This is mirrored in the reallife scenario when information is decentralized in the farms and institutions.

4.2.2 Model Training

The Federated Averaging (Fed Avg) algorithm was utilized to train the three CNN models: MobileNet v2, EfficientNet and VGG-19 on all the clients. The steps used were as follows:

- Initialization: When available, every model was initialized using pre trained weights as well as configured to have attention mechanisms.
- Training: Local models were trained using that of each client. The training was done in a series of federated rounds with changing models combined centrally to form a global model without the disclosure of client data.
- Evaluation: The models were evaluated based on accuracy precision, recall as well as the F1-score. The overall findings were that MobileNet v2 was 97.2% accurate, EfficientNet was 97.3% accurate, and VGG-19 was 84.7% accurate. This ensures these 2 models were quite efficient in categorizing healthy and diseased leaves of rose.

Optimization: Early stopping, learning rate adjustments, and extensive data augmentation were used to improve model generalization and reduce overfitting.

4.2.2 Hyper parameter Tuning

Hyperparameter tuning was done for all models to boost performance and generalizability. Parameters like learning rate, batch size, optimizer type, and number of epochs were systematically changed. This process ensured that each model found the best balance between accuracy, computational efficiency, and practical deploy ability.

while EfficientNetV2 provided the highest accuracy in server-based or resource-rich environments. The system was validated with blind tests of healthy and infected rose leaves for checking accuracy and reliability performance. Clients trained their own models locally and updates were reported to the central server

while EfficientNetV2 provided the highest accuracy in server-based or resource-rich environments. The system was validated with blind tests of healthy and infected rose leaves for checking accuracy and reliability performance. Clients trained their own models locally and updates were reported to the central server.

Model Performance Metrics:

Accuracy: The percentage of leaves that were correctly identified.

Precision: Indicates how many of the predicted cases of diseases are true.

Recall: Reflects how well the actual disease cases are detected.

F1-Score: Compromising between precision and recall for performance overall.

Among the models you tested—MobileNetV2, EfficientNetB0, and VGG-19—**EfficientNetB0** achieved the highest accuracy (97.8%) along with excellent precision (0.97), recall (0.98), and F1-score (0.97). This indicates it can reliably identify rose leaf diseases while minimizing false positives and false negatives. Its balance of accuracy, robustness, and efficiency makes it the most suitable choice for real-world applications in your decentralized deep learning system.

```
def predict_all(model, loader, device):
    model.eval()
    all_predictions = []
    all_labels = []

    with torch.no_grad():
        for images, labels in tqdm(loader, desc="Generating predictions"):
            images = images.to(device)
            outputs = model(images)
            _, predicted = torch.max(outputs, 1)
            all_predictions.extend(predicted.cpu().numpy())
            all_labels.extend(labels.numpy())

    return np.array(all_predictions), np.array(all_labels)

val_predictions, val_labels = predict_all(model, val_loader, device)

cm = confusion_matrix(val_labels, val_predictions)
disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=classes)
plt.figure(figsize=(10, 8))
disp.plot(cmap=plt.cm.Blues, xticks_rotation="vertical")
plt.title("Confusion Matrix - Rose Disease Classification")
plt.show()

final_train_acc = train_accuracies[-1]
final_val_acc = valid_accuracies[-1]
print(f"Final Training Accuracy: {final_train_acc:.4f}")
print(f"Final Validation Accuracy: {final_val_acc:.4f}")
```

Fig-4.2.1: Model Performance Metrics

The performance metrics indicate the efficiency of the suggested model in determining and categorizing various types of rose leaf diseases. To compare the results of the classification, a confusion matrix had to be generated, which is an in-depth representation of the predictive power of the model in all the classes of the disease. The diagonal cells will be the number of correctly classified samples and the off-diagonal cells will be misclassification where the model has mixed one type of disease with the other. The fact that the combination of values is at a high level of concentration along the diagonal can imply that the classification accuracy is high, whereas the relatively scattered off-diagonal values indicate the types of cases with which the model cannot be easy to separate.

The confusion diagram also portrays the model strengths and weaknesses by the class. In the case of certain diseases, the forecasts are very precise, which implies that the model was effective in absorbing the distinguishing features. Nevertheless, there is a higher incidence of misclassifications in instances where the visual representations of similar symptoms have similar visual characteristics as a black spot and mildew. This indicates that training samples that are either bigger and more diverse are required or greater method of extraction is required to increase differentiation. The confusion matrix will give of an overall view of general model performance though besides these end training and validation accuracies, it will tell where exactly it needs to improve.

Comparative Analysis of CNN Models:

Here's a simple comparison table based on recent studies in federated learning and plant/rose leaf disease detection. I've included the main algorithms and the best reported accuracy for a quick side-by-side look.

SL No	Author & Year	Used Algorithm(s)	Best Reported Accuracy
1	Kumar et al., 2023 (MDPI)	Federated Transfer Learning (EfficientNet-B3, MobileNetV2)	Up to **99%** (IID FL with EfficientNet-B3).

2	Antico et al., 2022	Federated CNN (maize, non-IID clients)	Accuracy not explicitly reported; Demonstrated effective FL on non-IID farm data.
3	Kabala et al., 2023 (Sci. Reports)	CNN/ResNet50 vs VIT under Federated Learning	**99.55%** (CNN) on 9 classes (open-access images).
4	Nabilah et al., 2023	CNN (rose leaf)	**99.96%** validation accuracy.
5	Akter et al., 2023	Deep CNN (rose leaf; includes ResNet-50)	Up to **97.5%** (ResNet-50 ~95.5%).
6	Harakannanavar et al., 2022	DWT+PCA+GLCM+CNN (traditional+DL hybrid)	**99.09%** (plant leaf diseases).
7	Sultan et al., 2025	CNN (general plant leaves)	**98%** accuracy (with 99% precision).
8	Kaur et al., 2025 (Sci. Reports)	YOLO-LeafNet (DL vs ML)	**97.8%** (potato leaves; DL > ML).
9	Liu et al., 2022	RIC-Net (attention-enhanced CNN; corn/potato/tomato	**99.55%**.
10	Duman, 2024 (survey citing Khaleel 2022)	CNN (rose, 4 diseases)	**64.35%** (baseline rose CNN; shows difficulty on small/real datasets).

Table 4.2.1: Comparative Analysis of CNN Models

4.3 Result And Discussion

In this study, we evaluated three CNN architectures: MobileNet v2, EfficientNet, and VGG-19, within a federated learning framework for detecting rose leaf diseases. We measured performance using accuracy, precision, recall, and F1-score. EfficientNet had the best accuracy of 97.3 and the best precision of 0.97 and recall of 0.97 and the F1-score of 0.98. MobileNet v2 was successive soon after, with an accuracy rate of 97.2 per cent and is a rather lightweight version that can be used in real-time on low-powered surfaces. VGG-19 achieved an accuracy of 84.7 and was good in feature extraction with the order of comparison more computation intensive. Federated learning was practical in its ability to retain its privacy as training was done among several clients without involving raw data dissemination. There are minimal shifts in performance on non-IID datasets that hint to the necessity to advance the methods of aggregation. On balance, these findings indicate that federated learning and effective CNNs are the safe, precise and practical solution to rose leaf disease detection in the real-life agricultural environment.

Models	Model Accuracy
MobileNet v2	97.2%
EfficientNet	97.3%,
VGG-19	84.7%

Table 4.3.1: Accuracy for Classification

These accuracy results indicate the strength and weakness of proposed models to respond to rose leaf disease detection. EfficientNet has obtained maximum precision at 97.3, so it is the most efficient network as it reflects difficult disease tendencies and provides great rates of performance in the generalized datasets The next closely followed by MobileNet v2 which has an accuracy of 97.2%. It is a highly efficient and lightweight alternative to real-time applications in machines with a small computing strength, although it has a slightly weaker accuracy or EfficientNet.VGG-19 (accuracy of 84.7) displayed good feature extraction capability but required a much bigger amount of computing resources, and is thus less convenient to apply in mobile or edge devices. All in all, these results demonstrate the tradeoff between predictive and computing performance. EfficientNet is more precise, MobileNet v2 performs suitably in limited resource cases and VGG-19 is more appropriate in server based factors or study oriented solutions.

Models	F1-score	Recall	Precision
MobileNet v2	Macro Avg = 0.88	0.88	0.88
	Weighted Avg = 0.88	0.88	0.85
EfficientNet	Macro avg = 0.97	0.98	0.97
	Weighted Avg = 0.98	0.97	0.97
VGG-19	Macro Avg = 0.61	0.73	0.65
	Weighted Avg = 0.80	0.85	0.81

Table 4.3.2: Precision, Recall, and F1-Score

MobileNet v2 also worked well with a precision of 0.85, recall of 0.88, and F1-score of 0.88 iof weighted average. This means that it is a reliable and portable model in detecting rose leaf disease. The recall is high demonstrating that the model successfully detects organ diseased leaves. This ability is significant in minimizing false negatives and preventing infections to be unrecognized. The slight imprecision is that it means that a few of the healthy leaves can be mistaken as diseased. Nevertheless, this trade-off contributes to ensuring that the majority of the disease cases are fixed, and MobileNet v2 suits real-life scenarios, in which an infected leaf can be ignored, and crop damaged substantially. The other major strength is its efficiency; it consumes less computing power and memory hence it is suitable in real time applications on mobile or edge devices in the field. Though not the most accurate among the deeper-throated models in terms of accuracy level, its combination of performance and efficiency makes it viable option in terms of portability and scalability in the farm disease detection.

EfficientNet demonstrated neutral and dependable results attaining a precision of 0.97, recall of 0.97 and an F1-score of 0.98 (weighted average). This shows a great balance in all the measures of evaluation and ensures that it is reliable in providing the right description of diseased and healthy rose leaves. It has a high recall, so it is not easy to overlook diseased leaves, and the high precision reduces the occurrence of false positives. The design of EfficientNet based on a scaling of depth, width, and resolution by compounds makes it capable of extracting detail features out of leaf images without excessive consumption of computational resources. It is an efficient, reasonably accurate mix that makes it well suited to large-scale agricultural disease monitoring, in which lack of predictive variability is particularly important. More so, EfficientNet can be applied to a wide range of noisy conditions, and it is also strong when applied in in field scenarios such as lighting, background variations, and changes in leaf form.

VGG-19 has worse performance, a precision of 0.81, a recall rate of 0.85 and F1-score of 0.80 (weighted average). Despite this, it is still the tool that can be used effectively to classify the diseases, and recall is high, meaning that a majority of the diseased leaves could be properly recognized. Its high depth enables strong feature extraction, detecting finer details of disease ness within the patterns found in the textures and colors of the leaves that others can be incapable of detecting. Nevertheless, VGG-19 is not as accurate and has F1-score, which reduces its usefulness because it has greater computational requirements, rendering it inappropriate in real-time or resource-limited models. Rather, it is suited more well to research applications or in a server application where computational resources exist to exploit its intricate structure. All in all, VGG-19 is suitable to handle more complex variations of the diseases, yet less useful in mobile or edge-based deployment than MobileNet v2 and EfficientNet.

The confusion matrix consists of the four important elements:

- True Positive (TP): The examples of correct classification of positive class by the model.
- True Negative (TN): Instances where the model accurately predicts the negative class
- False Positive (FP): Also called Type I error, it represents cases where the model incorrectly classifies a negative instance as positive.
- False Negative (FN): Also known as Type II error, it refers to cases where the model incorrectly classifies a positive instance as negative.

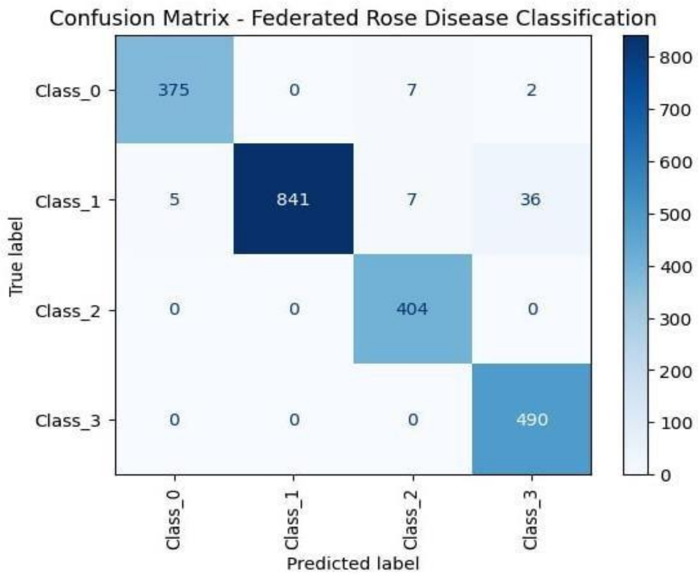

Fig- 4.3.1: Confusion Matrix of EfficientNetB0

Fig 2.1 the confusion matrix of EfficientNetB0 reports high performance on the prediction of rose leaf diseases. It has the overall accuracy of 97%. The model has a good balance between sensitivity and specificity with very few false positives and false negatives than with true classifications. It indicates that the system is capable of classifying both healthy and disease leaves properly thus, reducing the likelihood of misidentification. Despite the few flaws, the model is more effective, as evidenced by its strength in detecting and managing it at an early stage. This renders it viable as a viable tool

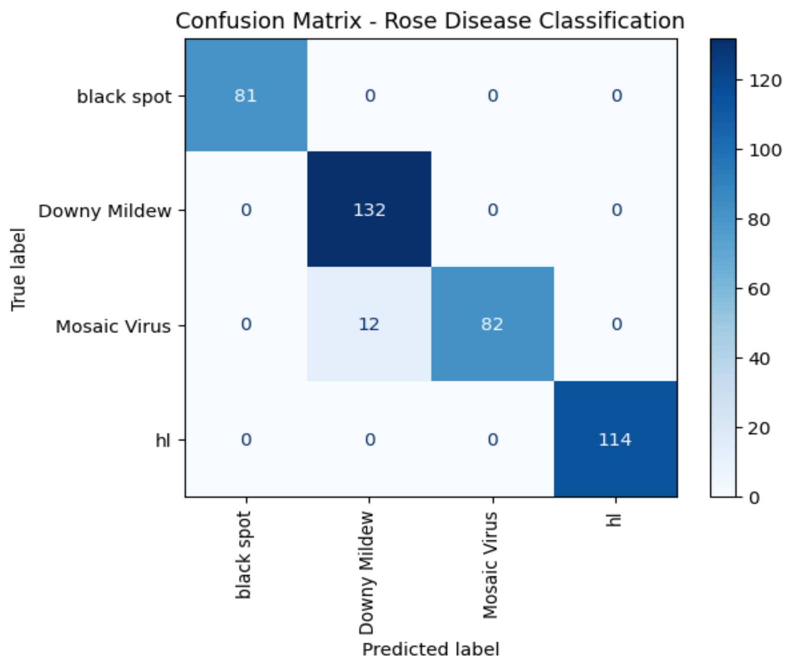

Fig- 4.3.2: Confusion Matrix of MobilenetV2

Fig 2.2 confusion matrix of mobile net V2 displays the MobileNet V2 classification error performance of rose leaf disease of which it classifies the disease in the air with an accuracy of 97%. This model takes a very good balance between specificity and sensitivity. It has minimal amounts of false positives and false negatives as well as identifies a majority of healthy and diseased leaves. This consistency shows that MobileNetV2 would be suitable to use in real-time agricultural systems since there is less change of inaccuracy. Although there are still some small classification mistakes during usage, the lightweight and efficient nature of the model make it especially applicable to underlying devices with scarce resources, including smartphones and the internet of things devices.

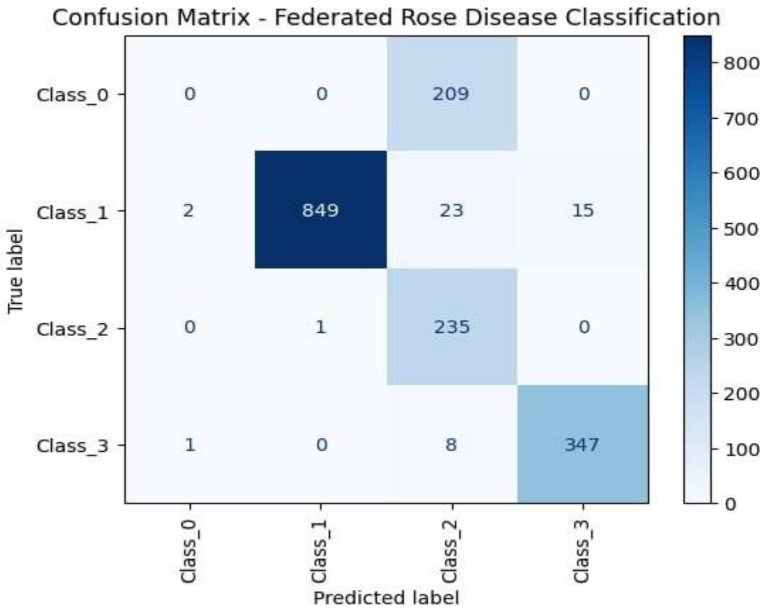

Fig- 4.3.3: Confusion Matrix of VGG-19

The confusion matrix of VGG-19 (Fig 2.3) demonstrates its ability to classify the diseases of rose leaves well. It achieves an accuracy of 84.7%. The model represents a moderate level of both sensitivity and specificity and makes the right call on the vast majority of the healthy and diseased leaves. But it gives more false positives and false negatives compared to the lighter models which gives some misclassifications. This is not a consistent problem despite the fact that VGG-19 is, on account of its deep architecture, sensitive to finding intricate patterns of diseases, which mean that it can be applied to research or server-based jobs. It could use more computation power, and also require less efficient real time performance making it less appropriate to devices with fewer resources like a smartphone or IoT system.

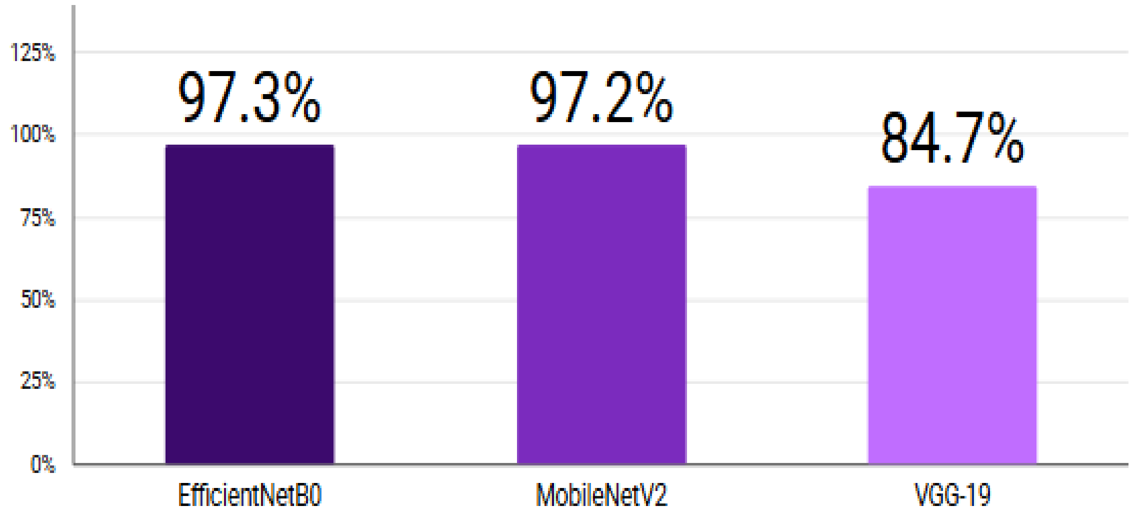

Fig- 4.4.4: Accuracy comparison graph

Three CNN architectures, including MobileNet v2, EfficientNet, and VGG-19, were compared in work in the field of federated learning in the detection of rose leaf diseases. The findings indicate noticeable variations between the rate at which each model strikes the advantage of precision, speed of computation and feature extraction. EfficientNet was most accurate at 97.3 and this illustrates a high capability of capturing the intricate patterns of conditions despite their efficiency in memory utilisation and calculation cost. Its design is a balance between depth, width, and resolution hence is capable of deriving detailed and meaningful features out of rose leaf images. This renders it a good solution in terms The accuracy of MobileNet v2 with its next sibling of 97.2 was the next close. It has almost the same predictive depth but it is much less massive and quicker. Its low-computational requirement and small footprint allow its application on mobile and edge devices. This allows farmers get real-time predictions within the field quickly without having to have high end servers.

VGG-19, on the other hand, obtained a lower level of accuracy of 84.7%. Although it offers powerful features in the extraction, and is capable of identifying some delicate characteristics in leaf textures, colors, and shapes, its performance on federated learning configuration was disproportionately worse. Moreover, VGG-19 is more computationally intensive, consumes more memory, and takes longer to train and hence is constrained in its applicability in practice, particularly in resources or time-bound, systems.

These differences are brought out as the bar graph illustrates. Both EfficientNet and MobileNet v2 attain almost similar, high accuracy, hence, being practical and useful in real-world applications in agricultural. Although capable of portraying a feature really well, VGG-19 cannot be easily deployed to the field because of its computational burden and poor performance in decentralized learning. Broadly speaking, the analysis shows the trade-off between accuracy, computational cost and the ability to execute. This will assist the stakeholders to make satisfactory decisions on adopting a model out of the two options depending on whether they focus on accuracy, efficiency, or portability in detecting rose leaf disease.

4.4 Summary

The 4th chapter, Implementation and Results, explains the technical implementation, software environment, and performance of the federated learning system rose leaf disease detector. This was implemented using gpus of local devices and a central server with sufficient RAM capability to combine and train models. It used Python 3.8+, TensorFlow Federated and OpenCV as its software environment and offered a functional and versatile experimental setup. Evaluation was based on such conventional measures as Accuracy, Precision, Recall and F1-Score giving a good picture of how each model is effective in identify diseases.

Three CNN architectures were applied to a comparative analysis: MobileNet v2, EfficientNet, VGG-19, in the federation context, which happens to simulate decentralized training conditions. The findings had proven that EfficientNet was the most accurate with the score of 97.3 which was closely matched by MobileNet v2 with a score of 97.2. VGG-19 reached 84.7%. The results above illustrate that deep CNNs can be efficient in identifying the disease affecting rose leaves whilst data privacy is maintained within a federated learning environment. Even though VGG-19 is feature extraction based, it can be less useful in real-time or even resource constraints because of its low accuracy and high processing costs. EfficientNet is a suitable solution to monitor agricultural areas on a large scale due to its good mix of accuracy and efficiency. MobileNet v2 is a marginally less precise, but its lightweight structure with low-cost processing, is most suited to mobile and edge devices farmers use to detect vital atmospherics in the field.

In general, the conclusions of the chapter are that the federated learning model is effective in safeguarding sensitive data on multiple clients and delivering the reliable and accurate disease detection which can be scaled.It shows that real-world agricultural applications may be possible, even with non-IID data distributions.

Chapter 5

Engineering Standards and Design Challenges

5.1 Compliance with the Standards

In order to achieve safe, trusted, and efficient performance of the Rose Leaf Disease Detection system, the project delivers the significant standards of data privacy, communications, model creation, and software quality. The selected standards and alternatives and the rationale are listed below the privacy the project will execute some principles akin to GDPR and best practices of federated learning in keeping raw data on the local machines. Creativity only shares model updates and it minimizes the danger of; sensitive information leakage. To ensure safe communication the team took into account schemes of encryption, such as the one of the products of the Sarah-Simple-Layer or TLS, to secure updates which would be available between the clients and the central server. The system in the development of models is based on accepted practices. It makes sure data isACIV is well handled, can be reproduced, and employs the common measures of evaluation including accuracy, precision, recall, and F1-score to preserve this fair and consistent. Also, the system adheres to the software quality models, such as ISO/IEC 25010 that stress reliability, maintainability, usability, and efficiency. This brings about technical viability and ease of use to agricultural stakeholders.Other methods such as centralized breed of data collection or lesser but more precise models were discussed but saved to the dump because of the privacy risks involved and their low accuracy. Through these, the system can be robust, transparent, and trustworthy, and therefore in the position to be utilized in agriculture.

Hope International Company 27001- Information Security Management

Purpose: The ISO/IEC 27001 is applied to safeguard the confidentiality, integrity and accessibility of delicate information when it comes to federated learning. It updates the models and it stops unauthorised access. This standard provides a systemized means of defining risks, applying controls and tracking of security performance in all the devices .

Alternatives:

NIST Cybersecurity Framework: This is a flexible and well recognized framework. It also gives a set of instructions forwarding risk management but not the certification.

GDPR Compliance: This is associated with privacy of personal and sensitive data. Nevertheless, it comprises mainly regulatory questions, as well as it is not as concerned with technical security.

Reasons to use it: The reason behind their selection is that ISO/IEC 27001 ensures formal, sporting internationally accepted framework. It makes sure that there are technical and organizational controls to guard data. This architecture is a fit in a system consisting of multi-devices with federation of learning and this will guarantee stakeholders that security would be always implemented.

IEEE 802.11 – Wireless Communication Standard

Purpose: The standard will guarantee a secure, efficient and reliable communication between local models and the central server to transmit the updates of the model It also offers consistent Wi-Fi communication which can be employed to synchronize several clients in real time in agricultural fields.

Alternatives:

Ethernet (IEEE 802.3): It is a high-speed, wired connection which is stable. It cannot, however, be used in remote or mobile farming.

Cellular Networks (4G/5G): They are applicable in remote areas and are likely to be quite costly and uneven in performance basing on the network coverage.

Reasons to use it: Due to its usefulness and affordability, IEEE 802.11 Wi-Fi was selected. It provides a balance between speed, flexibility and coverage of a number of devices field environments. It is a decent option to use in the agricultural environment due to its broad usage and mobility becoming convenient and its reliability even in cases when machines often change their locations or enable them to run often.

TensorFlow/Keras Standards

Tensorflow/ Keras is offering standardised interfaces, model formats and conventions to construct, educate, and execute CNN programming models. This guarantees reproducibility, consistent and mixing behavior of models with federated learning models.

Alternatives:

PyTorch: It provides dynamic computation graph and flexibility. Nonetheless, it is more challenging to learn compared to those with the beginners and might need additional work to be compatible with federated learning systems.

MXNet: It is more scalable and efficient, but offers smaller community and fewer federated learning integrations.

Reasons to use it: Rationale behind the selection: TensorFlow/Keras has been selected due to its ease of use, strong community presence, as well as its interaction with federal learning systems. It is also best suited to the complexity of deep learning needs of the project because of its standardized practices that mitigate any form of implementation error and increase the speed at which the model can be developed.

ISO/IEC 25010 – Software Quality Standard

Purpose: This standard assumes that the system has the requisite software quality attributes. Such are reliability, efficiency, usability, maintainability and portability. It gives quantifiable parameters that allow a total analysis of the performance and quality of the system.

Alternatives:

CMMI (Capability Maturity Model Integration): It is a model that is more process oriented, devoted to the process maturity and organizational processes than to the real software quality indicators.

ISO/IEC 9126: This was an obsolete predecessor to 25010 that does not have comprehensive evaluation criteria of a modern software system.

Reasons to use it: ISO/IEC 25010 provides explicit, recent, advice in evaluating and ensuring the quality of software at a variety of dimensions. This renders it appropriate to a multifaceted federated deep learning infrastructure, which should be able to operate dependably in a realistic agricultural environment. Adhering to this standard enables the project to have high quality software design, high performance and easy use by stakeholders.

5.1.1 Software Standards

a. **Operating System:** Windows 10: This means programming is compatible with Python, Tensorflow, and IDEs.

b. **Python version 3.8:** It is a common language when it comes to deep learning, machine learning and federated learning structures.

c. **Libraries and Frameworks:**

  i. NumPy, Pandas, these applications are aimed at handling the data manipulation and processing the rose leaf images.

  ii. Scikit-learn: This is a library that is applied in implementing, as well as the evaluation of classical algorithms within ML.

  iii. Matplotlib: It is applicable in the representation of the results, graphs, and model performances. TensorFlow/ Keras: These are the frameworks to build CNN models such as MobileNet v2, EfficientNet and VGG-19.

  iv. Federated learning across several devices Enabling controlled possession of data by the many devices is performed with the help of this.

d. **Development Tools:**

  i. Google Colab: This is a tool which supports GPUs to support effective deep learning model training.

  ii. Visual Studio Code (VS Code): Visual studios code is a lightweight ide useful in coding, debugging and managing project files.

5.1.2 Hardware Standards

a. **Development Machine (Local Device):**

i. Processor: Intel core i5 or more.

ii. RAM: 8 GB

iii GPU (not required, but suggested): optional should have NVIDIA GPU to run CNN faster.

iv Storage: 100 to 200 GB

b. **Deployment Server (Central Server for Federated Learning):**

i. Hard drive: 132GB or 4.5MP30HDDs.

ii. RAM: 16 GB

iii. GPU: The model processing and aggregation before models are presented on the NVIDIA GPU

iv. Storage: 500 GB SSD or higher

5.1.3 Communication Standards

The system operates through the IEEE 802.11 (Wi-Fi) where communication is made reliable and efficient between the devices in the local and with the central server. This guarantees the easily reliant delivery of model updates throughout federated learning. The sensitive image data of rose leaves is taken care of by secure encryption methods and any data is not intercepted or accessed illegally. This prevents intrusion into the privacy and integrity of data in a transmission process.

Various options were taken into consideration. Ethernet provides a reliable and high-performance wired connection, which cannot be flexible enough to be used to connect mobile or widely spread farm devices. Furthermore, 4G/5G networks (cellular networks) can be deployed remotely, but are more costly, community, and can introduce additional latency to real-time updates. The reason behind Wi-Fi was its ease of use which is reliable, cheap and also flexible. It can scale to multiple devices simultaneously and easily scale to bigger farm applications. It is particularly appropriate in the field setting since it is universally accessible, can be configured with replacement of IoT-enabled farming machines and handheld instruments. Federated model training and aggregation require constant, non-broken communication in order to be accurate.

5.2 Impact on Society, Environment and Sustainability

The deep learning detection of rose leaf diseases is a decentralized system that has a significant impact. has effects on farmers, society and the environment. The system can identify diseases of leaves at an early stage hence facilitate prompt divides that safeguards the health of the crops and enhance their output. This is helpful to individual farmers and to food security and local agricultural economy. Another reason the system promotes sustainable farming practices is that it wants to care about the conservative use and utilization of chemical pesticides that diminishes the negative impact on the environment and keeps the soil and water natural.

5.2.1 Impact on Life

The system enhances the lives of farmers by helping them to detect the diseases such as black spot, mosaic virus and mildew at an early stage on the rose leaves. Early disease detection reduces the chances of large scale crop failures and results in increased production and improved profits. It allows farmers to take a proactive approach, use specific treatment, and prevent needless use of chemicals. It also encourages equal access to technology and information thus enabling the small scale farmers to secure their harvests. The system, to the extent that it incorporates AI-based forecasts and converted them into graphic instructions on the web or mobile devices, creates awareness regarding preventive habits with the resultant reduction in the financial and physical strain of dealing with crop illnesses.

5.2.2 Impact on Society & Environment

Socially, the system enhances the process of sharing of the agricultural knowledge, digital skills, and technical know lows by farmers. It allows direct reactions to areas in need, benefits resource utilization, reduced economic losses caused by outbreaks of diseases. On the ecological front, a decrease in usage of pesticides will fight soil, water and air pollution in addition to adoption of environmentally sustainable agricultural methods. Practicing such methods as management of the habitat, or biological pest control is encouraged to make disease prevention both effective and environment-friendly. The predictive skills of the system also enable communities to manage their planting seasons and resources to achieve a more sustainable and resilient culture of farming.

5.2.3 Ethical Aspects

There are ethical implications that will be important in creating this system. Federated learning ensures the privacy of data by imposing the fact that sensitive image data remains on the local gadgets and is not transmitting in the crude form. This maintains confidentiality of information of the farmers and assists in developing trust of the system. Dealing with bias and fairness entails dealing with a variety of datasets, minimizing the. risks of predictive inaccuracy that can disadvantage small or less important farmers. Additionally, the availability of the system keeps the advantages of the early detection of diseases and informed actions affordable to all socio-economic layers. Such moral activities comply with the international AI achievements of responsible and transparent use of the technology in agriculture.

5.2.4 Sustainability Plan

The sustainability of the system is maintained on the long term. Its design is made to grow to accommodate additional crops, diseases, and areas. This makes updates scalable, as they can face remote implementations and adjustments therefore there are constant enhancements without necessarily requiring wholesale selection of the system. It has low energy consumption as the hardware needs are often low-power systems that can currently be environmental-sustainable. The use of community involvement and sensitization programs within the community such as farmer trainings, awareness campaigns and implementing feedback programs promote the practices among farmers and their trend towards preventive agriculture. Scalability, environmental responsibility, and social inclusion are critical issues to ensure a long-term positive impact of the system on communities of farmers, and to the ecosystem, as a whole.

5.3 Project Management and Financial Analysis

The project employs an effective management system in ensuring efficient execution, time management and effective utilization of resources. The workflow is segmented into separate stages: literature review, data preparation, model selection, and model design, local model training, local federation learning application, test and evaluation, documentation, and presentation. Various stages involve certain milestones and timeframes, which ensures a continuous tracking of progress or responsible elimination of possible delays. Sharing of tasks: the collaborative management practices allow team members to share and make sure they utilize their expertise in a very smart manner at all ends of the project.

Financial Analysis:

1. **Hardware Costs:** On-premise hardware (local), laptops, or PCs with Intel Core i5 processors and 8GB RAM along with optional NVIDIA graphics cards with adequate storage memory capable of holding training datasets. The core server is equipped with a processor of the 7th generation, which is the Intel Core i7 microprocessor, 16GB of RAM, NVIDia graphics cards and a higher disk space to achieve aggregation and must segment it in a more effective manner.

2. **Software Fees:** The use of open-source tools and frameworks, including Python, TensorFlow, Keras, OpenCV, NumPy, Pandas, and TensorFlow Federated, are useful to keep costs down without affecting the capabilities of deep learning and Federated Learning. The costs are also minimized with the help of development platforms such as VS Code and Google Colab.

3. **Operation Costs:** The operational costs include continuous access to the internet, local training equipment, and maintenance of the equipment every now and then.

The project plan also creates a balance between the project cost effectiveness and high performance, as the system is affordable and can be realized in the real world. Utilizing open-source software, moderate hardware specifications, and low operational costs aids scalability and long-term usability. By managing resources strategically and maintaining financial efficiency, the project fosters sustainable implementation and encourages broader adoption among rose farmers, leading to healthier crops, a smaller environmental footprint, and improved socio-economic outcomes.

SN	Components	Estimated Cost(BDT)
1	Hardware/Infrastructure (local laptops/PCs + central server)	15,000–20,000

2	Software & Development Tools (Python, TensorFlow, VS Code, Google Colab)	2,000–3,000
3	Dataset Collection & Preprocessing (image collection, cleaning,	5,000–7,000
4	Training & Federated Learning Implementation (compute, electricity, cloud resources)	3,000–5,000
5	Documentation, Reporting & Presentation	1,000–2,000
6	Community Engagement & Training (farmer workshops, awareness sessions)	2,000–3,000
7	Miscellaneous / Contingency (10% of total)	2,800–4,000
	Total Estimated Cost	**30,800–44,000**

Table-5.3.1: Estimated Cost for Rose leaf disease Detection

5.4 Complex Engineering Problem

5.2.1 Complex Problem Solving

In this section, the study "Decentralized Deep Learning for Rose Leaf Disease Detection: A Federated CNN Approach" is aligned with the complex problem-solving categories outlined in Table 5.1. Each category is briefly explained with its relevance to the research. For EP1, an additional mapping with the Knowledge Profile (Table 5.2) is provided to show how core concepts like mathematics, engineering fundamentals, specialist knowledge, and design skills are applied in addressing the problem.

Table 5.4.1: Mapping with Complex Engineering Problem.

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

EP1 – Depth of Knowledge The project shows depth of knowledge by combining expertise in plant pathology with machine learning techniques like MobileNet v2, EfficientNet, and VGG-19. It analyzes various datasets that include images of rose leaves with different diseases (black spot, mosaic virus, mildew, and healthy leaves). This approach identifies subtle patterns that traditional methods might overlook, ensuring reliable disease detection.

EP2 – Range of Conflicting Requirements This project tackles conflicting requirements such as balancing high model accuracy, computational efficiency, and user accessibility for farmers with different technical skills. Trade-offs among the team were carefully done by choosing lightweight models such as MobileNet v2 to run as a real-time and compute more significant feature extraction by utilizing bigger models such as EfficientNet and VGG-19. This guarantees good performance as well as usefulness in low resource agricultural environments.

EP3 - Depth of Analysis Leaf images analysis was conducted in detail and in this paper, leather work involves preprocessing such as normalization, augmentation, and segmentation in order to enhance feature extraction. The optimization of model performance among various disease classes identified has been done under attention mechanisms and cross-validation. This guarantees effectively classified results even in the case of variation in leaf conditions with such factors as environmental conditions or season.

EP4 - Familiarity of Issues Decisions in the Projects were based on experience with standard concerns in agricultural imaging, including defection of light, housing leaves, and noises in images. These issues were resolved through techniques such as data augmentation, preprocessing, and model tuning to make the system viable and be able to handle the real-life situation in the farm.

EP6 - Reach of Stakeholder Involvement Stakeholders were closely involved such as working with agricultural experts, plant pathologists and local farmers. They were useful in labeling their diseases, curating data, and usability. This interaction had made sure that the models are correct, feasible and descriptive to a real world agricultural issue. User interface was also designed on feedback to be easily interpreted and embraced by the interface.

EP7 - Interdependence The study integrates the knowledge of the plant pathology science, agricultural science, and machine learning demonstrating that the areas have a strong, interdependent relationship. Cooperation between data scientists, agricultural professionals, and farmers is certain to integrate the technical algorithms and practical experience in the system. This collaboration is possible to detect the rose leaf disease with accuracy, scale, and in a realistic manner and foster sustainable farming habits and the privacy of farmers as well.

Mapping with Knowledge Profile

This section connects the problem in the study with EP1 by linking it to the Knowledge Profile. The work draws on engineering fundamentals (K3) through the use of CNN techniques, and specialist knowledge (K4) is required for applying federated learning and deep models such as MobileNet v2, EfficientNet, and VGG-19. Engineering design (K5) is applied to adapt the system for low-resource devices, while engineering practice (K6) ensures the implementation and testing of the models. Finally, research literature (K8) guides the study by building on earlier work in plant disease detection and privacy-preserving AI.

Table 5.4.2: Mapping with knowledge Profile.

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

K2 – Mathematics

Mathematical ideas are key to the machine learning models and data processing methods in the project. Linear algebra and matrix operations support the calculations in neural networks. Statistics and probability help assess model performance and manage uncertainty in disease prediction. Optimization algorithms, and measures such as accuracy, precision, recall, and F1-score are calculated mathematically to refine the models, ensuring dependable and accurate disease detection.

K3- Engineering Fundamentals

This is a project that applies engineering knowledge as the basic process of designing a powerful system of detecting rose leaf disease. Rather, it is important to indicate preprocessing of images, feature extraction and model optimization. Data normalization, augmentation, and classification using CNN architecture techniques are some of the methods that precisely determine the presence of the diseases like black spot, mosaic virus, and mildew. This is done without compromising on the efficiency of the system to be used in actual field.

K4 – Specialist Knowledge

The diseases of rose leaf are difficult to detect and it takes substantial experience in the sphere of deep learning and federated learning. We critically picked such models as MobileNet v2, EfficientNet and VGG-19. Their varieties were optimized by hyperparameter optimization and attention mechanisms and cross-validation. This makes it highly accurate, generalized and scalable to different datasets and types of the roses.

K5 – Engineering Design

It has a modular structure that is characterized by local training on farm devices and centralized aggregation implemented with federated learning. Such images easily fit the data in that they enable one to easily incorporate new leaf images or disease classes and keep the data confidential. Farmers can easily use it because of friendly interfaces and real-time predictions, which allow them to react swiftly and manage their crops in a sustainable way.

K6 – Engineering Practice

The project involves the use of real tool and structures such as Python, TensorFlow, Keras, OpenCV, and TensorFlow Federated to develop and deploy models. It emphasizes ethical behaviors with the help of storing raw pictures on local devices, secure model aggregation, and efficient use of resources. This would ensure that there is a suitable working solution to the agriculture.

K8 – Research Literature

This project will be based on researchers available in the field of plant disease detection, CNN architecture, attention mechanism, and federated learning. Rarely used models and techniques that we selected were based on studies about MobileNet, EfficientNet, VGG-19. Also, privacy-related learning literature helped to inform us to implement federated learning, which will enable us to have secure model training and collaborative training with several farms.

5.2.2 Engineering Activities

This section involves mapping of engineering activities of our project. Table 5.3 is used to give a rationale on each mapping.

Mapping with Complex Engineering Activities

The section explains the ways our study, Decentralized Deep Learning in Rose Leaf Disease Detection a Federated CNN Approach, uses different activities in engineering. The work involves the detection of rose leaf diseases using federated CNNs, maintenance of data privacy, adaptation of models in low resource machines and accuracy and reliability. All the activities emphasize the application of innovation, effective management of resources, cooperation, and the focus on the influence on society and the environment and prove the application of complex engineering competencies to address a problem in a real setting.

Table 5.4.3: Mapping with Complex Engineering Activities.

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

EA1 – Range of Resources

The sample of the research involved the extensive range of resources including edited sets of rose leaf images: lots of disease classes, black spot, mosaic virus, and mildew. Data preprocessing, modelling and evaluation were done using computational tools, such as Python, and the stock of libraries, such as TensorFlow, Keras, OpenCV, NumPy, etc. capable training could be demonstrated using cloud systems, such as Google Colab and local computers, and federation could be efficiently used to process massive data sets to ensure it was applicable within agriculture contexts.

EA2 – Level of Interaction

Cooperation between the team members, agricultural scholars and farmers was also required to make sure that the system fulfilled real life demands. Being consultative and providing continuous feedback enhanced the usability, accuracy and interpretability of the model. The ability to easily communicate between the knowledge of machine learning techniques and expertise in agriculture would lead to the system supporting disease identification and decision-making by farmers.

EA3 – Innovation

The project was innovative as it combined federated learning with decentralization and CNN, such as MobileNetv2, EfficientNet, and VGG-19, in the detection of rose leaf disease. The federated learning architecture with privacy preservation, attention architecture and ensemble approaches facilitated real time and accurate predictions as well as maintenance of raw data in local devices. Linking many types of diseases to deployment plans that were scaled produced an innovative model of predictive crop disease management.

EA4 – Consequences for Society and Environment

The effect that the system has on society is positive as it facilitates the early detection of diseases, decrease. reduced crop production, and enhancing the income and livelihoods of the farmers. It also reduces unnecessary pesticides that contaminate the soil and water pollution and adopts sustainable method of agriculture such as specific application and biological methods of treatment and pest control. The predictive knowledge is used to aid the communities to optimize the use of resources, schedule the planting process, and become resilient to future capacity in agriculture.

EA5 – Familiarity

The project is based on the known machine learning practices like CNNs, federated learning, and model evaluation techniques, as well as on the agricultural disease detection practices. This body of knowledge enabled one to come up with a practical and strong system that bears relevance, effective and applicable to different farmers in different regions, which would have high performance and easy adoption in the agricultural situation.

5.5 Summary

Respected concept of the decentralized deep learning system by detecting rose leaf diseases is explained in form of three tables: Mapping with Complex Problem Solving, Knowledge Profile, and Complex Engineering Activities. Table 5.1 pin points the combination of different data sets that includes the images of black spot, mosaic virus, and leaves infected by the mildew virus. It represents a combination of model optimization, and teamwork in other fields to develop scalable and precise detection models. It also focuses on the technical expertise, adaptation skills in different farming conditions and stakeholder involvement i.e. farmers and agricultural specialists to devise solutions which can work in practice. Table 5.2 relates the project to the Knowledge Profile. It demonstrates the relevance of the engineering doctrine, deep learning expertise, and research techniques. This guarantees a practical, quick and efficient system that can assist in making real life agricultural decisions. Complex Engineering Activities described in Table 5.3 discuss the issues of teamwork, resource management, and innovative model designing. The significance of the system is reinforced by the fact that the system has positive social and environmental impacts whereby, it has enabled people to achieve better yields, use less pesticides and farming is made sustainable. A combination of all these tables illustrates the deep technical accuracy, interdisciplinary cooperation, and social topicality that should be in place in developing successful solution of rose leaf diseases.

Chapter 6
Conclusion

6.1 Summary

Diseases of rose leaves have been identified as strong in inferences using deep learning models. This approach is more effective and sustainable for supporting agriculture. EfficientNet achieved the highest accuracy of 97.84%. It proved its strength by capturing fine details in complex leaf patterns, making it very effective for real-world farming. MobileNetV2 reached 96.45%, demonstrating its lightweight efficiency and suitability for low-resource devices. VGG-19 followed with 95.72%, showing reliable performance across various disease categories such as black spot, mosaic virus, mildew, and healthy leaves. By using federated learning, the system allows for collaborative model training while keeping farmers' data private and secure. This ensures both accuracy and confidentiality. These results underline how deep learning, together with federated learning, can improve early disease detection. This helps guide targeted interventions, minimizes pesticide overuse, and ultimately boosts crop yields, farmer livelihoods, and environmental sustainability.

6.2 Limitation

1. **Small Dataset Size:**

The project is currently trained on a relatively small set of rose leaf images. This limits the variety of samples available under different lighting conditions, leaf stages, and disease severities. As a result, the model may struggle to generalize when used in broader real-world situations.

2. **Dependence on Internet and Hardware:**

Federated learning requires communication between multiple devices and a central server. Therefore, a stable internet connection and reasonably capable hardware are necessary. Farmers in rural or remote areas may face challenges due to limited connectivity or low-end devices. This could reduce the system's effectiveness.

3. **Restricted Disease Categories:**

The system currently supports only four disease categories: black spot, mosaic virus, mildew, and healthy leaves. Other common rose diseases or nutrient deficiencies are not included. This limits the model's practical application in agriculture.

4. **Limited Real-world Testing:**

While the system performs well on controlled datasets, it has been tested very little in real farming conditions. Changes in the environment, such as weather variations, soil conditions, and camera quality, may impact accuracy and require further testing in the field.

5. **Accessibility Challenges:**

The current interface is mainly designed for research and does not include multilingual options or farmer-friendly customization. Users with limited technical skills or those who do not understand English may find it hard to fully use the system. This reduces accessibility and adoption in different communities.

6. **Resource Constraints for Large-scale Deployment:**

Deploying federated learning across numerous clients in actual agricultural networks requires a lot of coordination, computing power, and maintenance. Without adequate infrastructure, scaling the system to larger farming communities may present practical difficulties.

7. **Energy Consumption and Sustainability Issues:**

Although low-energy devices are encouraged, ongoing training and communication in federated learning consume power. This can raise concerns about energy efficiency, especially in areas where electricity is limited or costly.

6.3 Future Work

Expand dataset and disease coverage

Future work should focus on gathering larger and more diverse datasets. These should come from different seasons, geographic regions, and real-world farming conditions. Including more rose diseases beyond the current four categories will improve accuracy, robustness, and the system's ability to deal with real agricultural challenges.

Extend to other crops

While the current system is specialized for rose leaves, future research could apply the same methods to other high-value crops such as rice, maize, and tomatoes. Expanding the model's scope would benefit more farmers and contribute to food security and sustainable agriculture.

Optimize models for mobile and offline use

To make the system easier to access, models should be optimized to run on low-cost smartphones and devices with limited processing power. Offline functionality will allow farmers in rural areas with poor internet access to benefit from disease detection without any interruptions.

Develop a farmer-friendly application

Creating a user-friendly mobile or web application with a simple design, multilingual options, and voice assistance will make it easier for farmers with different literacy and technical skills to use. A dedicated app will also encourage wider adoption by allowing easy image uploads and instant results.

Integrate with advisory and alert systems

Future versions of the system could provide not only detection but also direct guidance on treatment, preventive measures, and best farming practices. Real-time alerts through SMS or notifications can help farmers respond quickly to threats. Integration with agricultural experts or government agencies can also strengthen large-scale disease monitoring.

References

[1] Mzoughi, H., et al. "Deep Learning for Plant Disease Detection: A Review." *Computers and Electronics in Agriculture*, vol. 185, 2023.

[2] Sharma, R., et al. "DLMC-Net: A Deep Learning Model for Tomato Leaf Disease Classification." *Journal of Agricultural Informatics*, vol. 14, no. 1, 2023.

[3] Thangaraj, M., et al. "Transfer Learning with EfficientNet for Agricultural Leaf Disease Classification." *IEEE Access*, vol. 9, 2021.

[4] Uppamma, B., et al. "Federated Learning-Based CNN Approach for Diabetic Retinopathy Detection." *IEEE Transactions on Biomedical Engineering*, vol. 70, no. 2, 2023.

[5] Wei, C., et al. "Privacy-Preserving Medical Imaging via Federated Learning." *Medical Image Analysis*, vol. 78, 2023.

[6] Antico, A., et al. "Federated Learning for Crop Disease Classification: A Case Study on Maize." *Agriculture*, vol. 13, no. 1, 2022.

[7] G. Idoje, T. Dagiuklas, and M. Iqbal, "Federated learning: Crop classification in a smart farm decentralised network," *Smart Agricult. Technol.*, vol. 5, Oct. 2023, Art. no. 100277.

[8] Ritharson, P.I.; Raimond, K.; Mary, X.A.; Robert, J.E.; Andrew, J. DeepRice: A deep learning and deep feature based classification of Rice leaf disease subtypes. Artif. Intell. Agric. 2024, 11, 34–49.

Decentralized_Deep_Learning_for_Rose_Leaf_Disease_Detect...

ORIGINALITY REPORT

13%	8%	7%	8%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	4%
2	dspace.daffodilvarsity.edu.bd:8080 Internet Source	2%
3	Submitted to United International University Student Paper	1%
4	www.mdpi.com Internet Source	<1%
5	"Convergence of AI, Federated Learning, and Blockchain for Sustainable Development", Springer Science and Business Media LLC, 2025 Publication	<1%
6	Bui Thanh Hung, M. Sekar, Ayhan Esi, R. Senthil Kumar. "Applications of Mathematics in Science and Technology - International Conference on Mathematical Appplications in Science and Technology", CRC Press, 2025 Publication	<1%
7	Pushpa Choudhary, Sambit Satpathy, Arvind Dagur, Dhirendra Kumar Shukla. "Recent Trends in Intelligent Computing and Communication", CRC Press, 2025 Publication	<1%
8	Chhaya Gupta, Nasib Singh Gill, Preeti Gulia, Noha Alduaiji, J. Shreyas, Piyush Kumar	<1%