VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi-590018

A Major Project (21ISP76) Report On

"EFFICIENT AND ROBUST DEEP LEARNING FOR REAL TIME PLANT DISEASE DETECTION USING CNN"

Submitted in partial fulfilment of the requirements for VII semester of the degree of Bachelor of Engineering In

Information Science and Engineering

Submitted by

Kyle Lewis	(1CR21IS082)
Lokesh V	(1CR21IS083)
Rajsree S	(1CR21IS126)

Under the Guidance of,

Dr. Saba Tahseen

Assistant Professor, Department of ISE

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING CMR INSTITUTE OF TECHNOLOGY

#132, AECS LAYOUT, IT PARK ROAD, KUNDALAHALLI, BANGALORE-560037

2024-2025

CMR INSTITUTE OF TECHNOLOGY

#132, AECS LAYOUT, IT PARK ROAD, KUNDALAHALLI, BANGALORE-560037

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

Certificate

This is to certify that the project work titled "EFFICIENT AND ROBUST DEEP LEARNING FOR REAL TIME PLANT DISEASE DETECTION USING CNN" has been undertaken by Kyle Lewis (1CR21IS082), Lokesh V (1CR21IS083) and Rajsree S (USN 1CR21IS126), Bonafide students of CMR Institute of Technology, in partial fulfilment of the requirements for the degree of Bachelor of Engineering in Information Science and Engineering under Visvesvaraya Technological University, Belagavi, for the academic year 2024-2025.

It is hereby certified that all corrections and suggestions provided during the Internal Assessment have been duly incorporated into the final report deposited in the departmental library. The project report has been approved as it meets the academic requirements for the aforementioned degree.

Dr. Saba Tahseen Assistant Professor Dept. of ISE, CMRIT	Dr. Jagadishwari V Professor & HOD Dept. of ISE, CMRIT	Dr. Sanjay Jain Principal CMRIT

External Viva

Name of the Examiners	Signature with date
1.	
2.	

CMR INSTITUTE OF TECHNOLOGY

#132, AECS LAYOUT, IT PARK ROAD, KUNDALAHALLI, BANGALORE-560037

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

Declaration

We, the undersigned students of the Information Science and Engineering Department, CMR Institute of Technology, Bangalore, hereby declare that the project work titled "EFFICIENT AND ROBUST DEEP LEARNING FOR REAL TIME PLANT DISEASE DETECTION USING CNN" has been successfully completed under the guidance of Assistant Professor Dr. Saba Tahseen.

This dissertation is submitted in partial fulfilment of the requirements for the award of the degree of Bachelor of Engineering in Information Science and Engineering for the academic year 2024-2025. We further affirm that the contents of this project report are original and have not been submitted previously by any individual for the award of any degree or diploma at any university.

Place:

Date:

Team Members		Signature
KYLE LEWIS	(1CR21IS082)	________________
LOKESH V	(1CR21IS083)	________________
RAJSREE S	(1CR21IS126)	________________

ABSTRACT

The increasing threat of plant diseases poses significant challenges to global agriculture, often leading to severe yield losses if not detected early. Manual inspection of crops is time-consuming, requires expert knowledge, and may result in inconsistent evaluations. This project aims to develop an automated and efficient deep learning model for real- time plant disease detection using advanced classification and object detection techniques. The model effectively identifies multiple plant diseases from leaf images, optimizing the trade-off between computational efficiency and high accuracy. The model demonstrated superior performance in identifying disease patterns, even in complex agricultural environments. Furthermore, the system includes assist farmers in disease management, improving detection efficiency in remote areas. This AI-powered solution reduces dependency on manual assessment, enhances agricultural productivity, and contributes to a scalable, technology-driven approach to plant health monitoring.

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to CMR Institute of Technology, Bengaluru, for providing the platform to pursue our studies and carry out our final year project, ***"EFFICIENT AND ROBUST DEEP LEARNING FOR REAL TIME PLANT DISEASE DETECTION USING CNN."***

Our heartfelt thanks go to Dr. Sanjay Jain, Principal of CMRIT, for his continuous encouragement and support throughout our academic journey.

We are deeply grateful to Dr. Jagadishwari V, Professor and Head of the Department of Information Science and Engineering, for her unwavering support and motivation during this project.

We consider it a privilege to express our sincere appreciation to our project guide, Dr. Saba Tahseen, Assistant Professor in the Department of Information Science and Engineering, for her expert guidance, valuable feedback, and constant encouragement during the course of this project.

We would also like to thank all the faculty members of the Department of Information Science and Engineering, who, directly or indirectly, provided support and encouragement throughout our project work.

Finally, we express our deepest gratitude to our parents and friends for their moral support, understanding, and encouragement, which have been instrumental in the successful completion of this project. Thank you all for your invaluable contributions.

Kyle Lewis (1CR21IS082)
Lokesh V (1CR21IS083)
Rajsree S (1CR21IS126)

Table of Contents

List of Tables

Table of Figures

CHAPTER 1

INTRODUCTION

This project focuses on developing a deep learning model for real-time plant disease detection, addressing the challenges of crop health monitoring. The model is designed to identify plant diseases from images of leaves, enabling early and accurate diagnosis to prevent agricultural losses. By classifying plant images into different disease categories, the system helps farmers take timely corrective measures, ultimately improving crop yield and sustainability. A Streamlit-based frontend is developed to allow users to upload plant images and receive real-time disease predictions. The model is trained using a CNN-based approach, ensuring an efficient and scalable solution for plant disease detection. The project evaluates the model's performance with the help of key metrics such as accuracy, precision, recall, and F1-score, ensuring its reliability and effectiveness in real- world agricultural scenarios.

1.1 Introduction to Plant Diseases

Plant diseases are a major threat to global agriculture, leading to significant reductions in crop yield and food security. Diseases such as bacterial blight, powdery mildew, rust, and late blight can severely impact plant health, often spreading rapidly if left undetected. Early diagnosis and intervention are fatally important for preventing crop damage and maintaining agricultural sustainability. Traditional disease detection methods, such as manual inspection by farmers or agricultural experts, are often time-consuming, subjective, and require specialized knowledge. Deep learning techniques, particularly Convolutional Neural Networks (CNNs), have shown great potential in automating plant disease classification from leaf images. This project leverages deep learning to develop a real- time plant disease detection system, helping farmers detect diseases early and take necessary corrective actions.

Figure 1.1: Representative of various types of the disease that are most commonly observed on plants.

1.2 Challenges in Plant Disease Detection

Manually diagnosing plant diseases is labor-intensive, error-prone, and inconsistent, particularly in large-scale farming. Several factors make traditional disease detection methods challenging:

- Dependence on expert knowledge: Many farmers lack direct access to plant pathologists, making timely disease identification difficult.
- Visual similarity between diseases: Many plant diseases exhibit similar visual symptoms, making accurate classification difficult even for trained experts.
- Environmental variability: Lighting conditions, background clutter, and different plant species can introduce inconsistencies in manual diagnosis.
- Delays in treatment: Late diagnosis often results in severe crop losses, increasing the need for an automated, scalable detection method.

A deep learning-based system provides a solution by automating plant disease classification, reducing dependence on human expertise while ensuring faster and more reliable diagnoses. By leveraging CNNs for image classification, this project enables accurate and timely disease assessment, helping farmers improve crop health and productivity.

The impact of these challenges extends beyond immediate crop losses. The limitation of seasonal availability of expert pathologists creates bottlenecks during critical growing periods when multiple farms require simultaneous assistance. Additionally, the cost of bringing in specialists, especially in remote agricultural areas, can be prohibitively expensive for small-scale farmers, creating an economic barrier to proper disease management. The scalability issue becomes particularly evident in modern industrial agriculture, where a single operation might need to monitor thousands of acres of crops. The time required for thorough manual inspection of such large areas makes it practically impossible to maintain consistent monitoring frequency, potentially missing critical intervention windows.

1.3 Objectives

The objective of this project is to develop a deep learning model that accurately classifies plant diseases with the help of leaf images. The system aims to:

1. Train a CNN-based model which classifies multiple plant diseases based on image data, this involves selecting an appropriate Convolutional Neural Network (CNN) architecture and training it on a labeled dataset of plant leaf images, where each image is tagged with the corresponding disease or healthy status.

2. Develop a real-time prediction system using a Streamlit-based web application for easy user interaction. This entails creating a user-friendly web interface using the Streamlit framework. Users will be able to upload leaf images through the application, which will then be processed by the trained CNN model to provide real- time disease predictions,

3. Optimize model accuracy and efficiency to ensure reliable disease detection across different plant species and environmental conditions. This incorporates techniques like data augmentation (artificially expanding the training data), hyperparameter tuning (fine-tuning the model's settings), and potentially using transfer learning (leveraging pre-trained models).

4. Evaluate model effectiveness with established standard metrics such as accuracy, precision, recall, and F1-score to assess classification effectiveness. This crucial step will quantify the model's performance on a held-out test dataset. These metrics will offer valuable insights into the model's strengths and weaknesses, helping identify areas for improvement.

5. Create an automated feedback system that collects user validation of predictions, enabling continuous model improvement and adaptation to new disease patterns. This will involve integrating a feedback mechanism within the Streamlit application. Users can confirm or correct the model's predictions, providing valuable ground truth data. This iterative process will facilitate retraining the model periodically with the new data, improving its accuracy and adaptability over time, especially for emerging diseases or regional variations.

By achieving these objectives, the project seeks to provide an accessible, AI-driven plant disease detection tool that supports early diagnosis and sustainable agricultural practices.

1.4 Motivation for the Study

Deep learning models have shown significant advancements in image classification, including applications in plant disease detection. However, real-world agricultural challenges, such as variation in plant species, environmental conditions, and dataset imbalances, make it difficult to deploy accurate AI-based solutions.

This study focuses on developing a simple yet effective CNN model capable of effective generalization across different plant disease categories. By ensuring robust classification accuracy, the project seeks to offer scalable, automated solution for early disease detection, reducing farmers' dependence on manual inspections.

Early disease identification is critical for preventing crop loss, controlling the spread of infections, and reducing excessive pesticide use. This project aims to support precision agriculture by offering an AI-powered, real-time disease detection system, ultimately improving agricultural efficiency and sustainability.

1.5 Context and Purpose of Study

Plant diseases such as bacterial blight, rust, powdery mildew, and late blight significantly affect crop productivity and global food security. Conventional disease detection approaches depend on manual observation, which is time-consuming, inconsistent, and not scalable. In many rural agricultural regions, farmers lack access to trained agronomists, leading to delayed disease identification and widespread crop damage.

This project aims to overcome these challenges through the use of deep learning techniques to develop an automated, image-based plant disease detection system. Unlike traditional manual methods, CNN-based classification models are capable of handling vast datasets of leaf images quickly and accurately, identifying diseases with high reliability.

The goal of this project is to:

- Develop a real-time, AI-driven disease classification system that farmers can easily use.

- Improve early detection and intervention strategies, reducing agricultural losses.
- Enhance precision farming techniques, enabling data-driven decision-making for plant health management.
- By integrating deep learning and image classification, this study seeks to provide an accessible, effective, and scalable tool for plant disease detection, supporting modern agricultural practices and sustainable farming.

CHAPTER 2

LITERATURE SURVEY

The integration of artificial intelligence (AI) in agriculture has seen significant advancements in recent years, addressing the growing need for efficient and scalable solutions to detect and manage plant diseases. Traditional disease identification methods, such as manual inspection, are labor-intensive and reliant on expert knowledge, making them less feasible for large-scale farming operations and inaccessible to many small-scale farmers. AI-driven techniques, particularly those leveraging deep learning, have emerged as a transformative approach, providing automated and highly accurate analysis of plant health. Convolutional Neural Networks (CNNs) have demonstrated exceptional capability in identifying and categorising plant diseases from leaf images, enabling early intervention and reducing the risk of large-scale crop losses.

2.1 Introduction

The use of artificial intelligence (AI) in agriculture has gained considerable momentum in recent years, addressing the growing need for efficient and scalable solutions to detect and manage plant diseases. Traditional methods of plant disease identification, such as manual inspection, are time-consuming, subjective, and heavily reliant on expert knowledge. This renders them unsuitable for large-scale farming and inaccessible to small-scale farmers who lack direct access to agronomists. AI-driven solutions, particularly deep learning-based models, offer a transformative approach by automating disease identification and providing highly accurate classification of plant diseases. Convolutional Neural Networks (CNNs) have shown strong performance in analyzing plant leaf images, enabling early detection of fungal infections, bacterial blights, and viral pathogens.

2.1.1 The Growing Need for Automated Plant Disease Detection

Pathogens are some of the most threatening plant diseases that threaten global agricultural productivity. They often progress in the early stages without obvious symptoms, so it is essential to detect them early in order to prevent widespread crop damage and ensure food security. Traditional plant disease diagnosis requires expert evaluation, sophisticated tools, and significant time, making it inaccessible in many rural and underserved farming regions. This lack of timely disease detection leads to severe yield losses, increased reliance on pesticides, and greater strain on agricultural supply.

By embedding deep learning into agriculture, the project introduces an automated plant disease detection system which can quickly classify plant diseases based on leaf images. AI tools offer real-time, scalable, and widely accessible solutions that make early intervention possible, thereby minimizing crop losses and enhancing sustainable farming practices.

2.1.2 Objective of the Study

The primary objective of this study is to develop a deep learning-based classification model capable of identifying multiple plant diseases from leaf images. The system is designed to:

- Train a CNN model that capable of accurately identifying various plant diseases using image data.
- Develop a Streamlit-based frontend to allow users to upload leaf images and receive real-time predictions.
- Improve classification accuracy through optimized model training and evaluation techniques.
- Ensure model reliability across different plant species and environmental conditions.

This project aims to reduce the dependency on manual disease detection methods, providing an automated, scalable, and accessible solution for farmers and agricultural professionals. By enabling early and accurate disease detection, this study contributes to enhanced crop yields, reduced pesticide misuse, and improved farm productivity.

2.2 Background

2.2.1 The Role of Artificial Intelligence in Agriculture

AI has revolutionized agriculture by enabling automated plant disease detection through image analysis and deep learning techniques.

Deep learning models, especially CNNs, have demonstrated high accuracy in identifying plant diseases from leaf images. Research has indicated that AI-based diagnostic systems can be as good as or even better than human expertise in the identification of plant infections. Studies indicate that deep learning models can process large datasets efficiently, thereby providing rapid and reliable disease identification with minimal human intervention.

All these developments make AI-based plant disease detection an efficient, scalable, and accurate solution for modern agriculture. By automating routine screenings, AI reduces reliance on human experts, ensuring that farmers in resource- limited areas have access to timely and effective disease detection tools.

2.2.2 Challenges in Multi-Class Classification of Plant Diseases

Identifying multiple plant diseases from a single data source is a complex task that presents several challenges, particularly when dealing with multi-class classification in agricultural imaging. One of the primary obstacles is class imbalance, where certain diseases may have fewer samples than others, leading to biased models that are more likely to predict the more common conditions. This issue can result in poor generalization and reduced model accuracy, especially for underrepresented classes, making it challenging to accurately detect rarer plant diseases.

Another challenge is the subtle differences between disease features, as many plant diseases share similar visual characteristics in their early stages. For example, fungal infections and bacterial blights may exhibit overlapping patterns, making it difficult for a deep learning model to differentiate between them. Accurately distinguishing between these subtle variations requires a highly sensitive and fine-tuned model capable of capturing these nuanced differences in plant leaf images.

Additionally, generalization across diverse crops and environmental conditions adds another layer of complexity. Plant images can differ significantly depending on factors such as crop type, geographic location, and environmental stressors, which means that a model trained on one dataset may not yield the same performance when applied to another. Ensuring that the model can generalize effectively across different crops and real-world agricultural settings is essential for its robustness and reliability.

To overcome these challenges, advanced methods like data augmentation, transfer learning, and re-sampling methods are often employed to mitigate class imbalance and improve model performance. Incorporating explainable AI methods also helps in understanding and interpreting the decision-making process, making the system more transparent and trustworthy in agricultural environments. Overcoming these challenges is crucial in developing an accurate, efficient, and reliable diagnostic system for plant diseases that can be used globally.

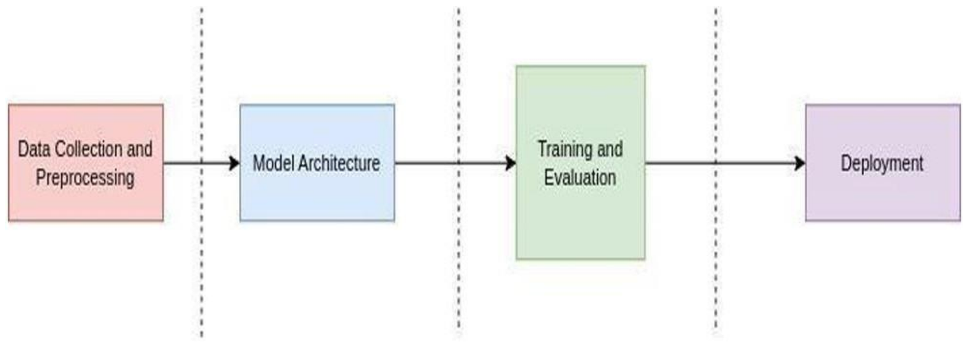


Figure 2.1: Flow chart of Work

2.3 Related Work

2.3.1 Deep Learning in Plant Disease Detection

Convolutional Neural Networks (CNNs) have transformed automated plant disease detection, enhancing agricultural management through advanced image processing. These models specialize in analyzing leaf images, achieving classification accuracy comparable to expert agronomists. CNNs are highly effective in plant disease identification due to their ability to autonomously extract hierarchical features.

Deep learning-based CNN models have been extensively applied in plant disease detection, including the presence of fungi, bacterial blights, and viral pathogens. The models are composed of several convolutional layers, which increases feature extraction; this makes the model more insensitive to variations in lighting, texture, and structures of leaves and enhances robust classification over different datasets. It has been proven that CNNs efficiently manage multi-class disease classification. The models process large-scale image datasets with efficiency and high accuracy.

As CNN models continue to improve, their capabilities for real- time disease identification expand, equipping farmers with improved tools for early detection, timely treatment, and improved agricultural productivity.

2.3.2 Advancements in Multi-Class Detection

Recent progress in deep learning has drastically improved multi-class detection techniques in agricultural imaging, particularly for plant disease classification. One of the strategies is to fine-tune convolutional architecture for the better description of feature space that might lead a model to more accurate identification of disease-specific patterns. These improvements thus overcome the general problem concerning visual similarity between plant diseases, and classification becomes reliable across different kinds of conditions. Fine-tuning parameters of models and improving training methodologies allows for higher accuracy with retained computational efficiency.

Moreover, stacking models and optimization methods have also been pivotal in improving the performance of the classifier. This technique allows predictions from more than one model to be combined into more robust predictions, thereby decreasing the effect of misclassifications. Studies have proven that optimized CNN layers combined with better training strategies considerably improve performance in distinguishing several types of plant diseases, even when symptoms overlap. This refined approach ensures that models can effectively handle complex multi-class classification tasks, where different plant conditions exhibit subtle variations in visual features.

2.3.3 Addressing Dataset Imbalance

One of the most persistent challenges in multi-class plant disease classification is dataset imbalance, where certain diseases have significantly fewer samples than others. This imbalance skews model predictions, causing deep learning algorithms to favor dominant disease categories, ultimately lowering accuracy for rarer conditions. To address this issue, researchers have developed multiple strategies that improve model performance while ensuring a balanced representation of all disease types.

One widely adopted approach is data augmentation, which applies transformations such as image rotation, flipping, scaling, and brightness adjustments to artificially increase the number of samples for underrepresented classes. Additionally, resampling methods such as oversampling and undersampling are employed—oversampling increases the occurrence of minority class samples, while undersampling reduces the dominance of majority classes to create a more balanced dataset.

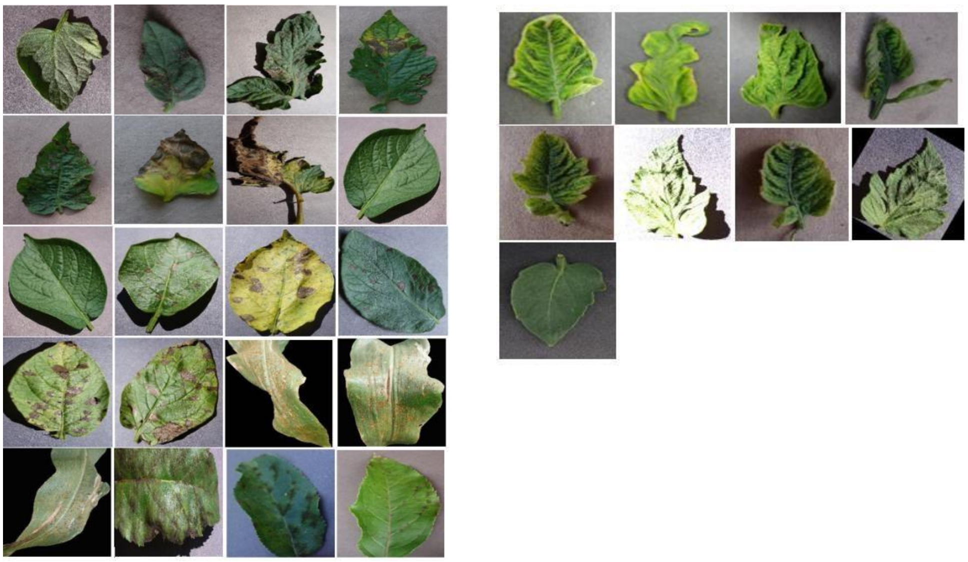


Figure 2.2: All 29 different Categories of the Test dataset

Technique	Impact on Imbalance
Data Augmentation	Increases dataset diversity, improving model robustness.
Oversampling	Enhances rare class representation, improving accuracy.
Undersampling	Prevents overfitting to dominant classes, promoting fairness.
Class Weights	Penalizes misclassifications of rare classes more heavily.
Focal Loss	Reduces bias towards dominant classes, improving rare class detection.

Table 2.1: Techniques for Addressing Dataset Imbalance in Multi-Class Classification

2.3.4 Explainability in AI Models

As AI-driven plant disease detection systems become more widely used in agriculture, explainability is essential for building trust among farmers, agronomists, and researchers. Users need clarity on how AI models make predictions so they can confidently use these tools for disease identification and management. Explainable AI (XAI) focuses on techniques that enhance model transparency and interpretability, ensuring users understand the reasoning behind predictions.

Methods such as visual heatmaps and activation mapping help illustrate which leaf regions the AI model prioritizes when making disease classifications that provide insights into which image characteristics contribute most to classification decisions. By incorporating these explainability techniques, AI models become more interpretable and trustworthy, fostering greater adoption in agricultural disease monitoring.

2.3.5 Integration of AI with Agricultural Practices

AI-based plant disease detection models to be useful in a real-world farming scenario, they need to integrate organically into existing agricultural workflows. They have to be accurate enough, accessible enough, and easy enough to use, so application by farmers and agricultural professionals is straightforward. In other words, an easy-to-use interface between users and AI-based disease classification models can simplify interactions.

This project features a web application built using Streamlit, where users can upload leaf images, get real-time predictions, and access treatment recommendations. The application does not require technical expertise, making AI-driven plant disease detection more accessible to farmers, researchers, and agricultural extension officers. This system, therefore, ensures disease diagnosis is effective and user- friendly with direct interaction with the AI model, enabling farmers to make informed decisions for early intervention and improved crop management. This integration contributes to better real-world agricultural productivity, leading to sustainable farming practices and the effective prevention of diseases.

CHAPTER 3

METHODOLOGY

This project follows a structured approach to developing an AI-based system for real-time plant disease detection. The process starts with planning and requirement gathering, defining project objectives, and identifying necessary resources. Data collection and preprocessing ensure a high-quality dataset for training the model. A custom CNN-based deep learning model is then developed and trained to classify plant diseases with high accuracy, addressing issues like dataset imbalance and environmental variability.

With this model, one trains it, then embeds it in a Streamlit-based web application; this will help the user to upload images of leaves and have the predictions returned in real-time. The testing, deployment, and usability test ensure that the system is correct, reliable, and accessible to agricultural applications at the end of the process. This is a full-proof method to have an applicable and scalable solution for better plant disease monitoring and management.

3.1 Planning and Requirement Gathering

The planning and requirement-gathering phase is crucial for structuring the project to ensure efficiency and success. This phase involves defining key project goals, understanding challenges in plant disease detection, and identifying the resources needed for development and deployment. A clear roadmap is established to maintain workflow efficiency, mitigate risks, and ensure alignment with desired outcomes.

Additionally, collaboration with agricultural experts and researchers is essential to ensure that the model is designed to address real-world farming challenges. This phase also includes specifying technical requirements, such as dataset selection, deep learning frameworks, computational resources, and Streamlit integration for frontend deployment. By carefully planning these aspects, the project aims to develop a robust and scalable AI-driven solution for real-time plant disease detection, ultimately contributing to higher agricultural efficiency and sustainability.

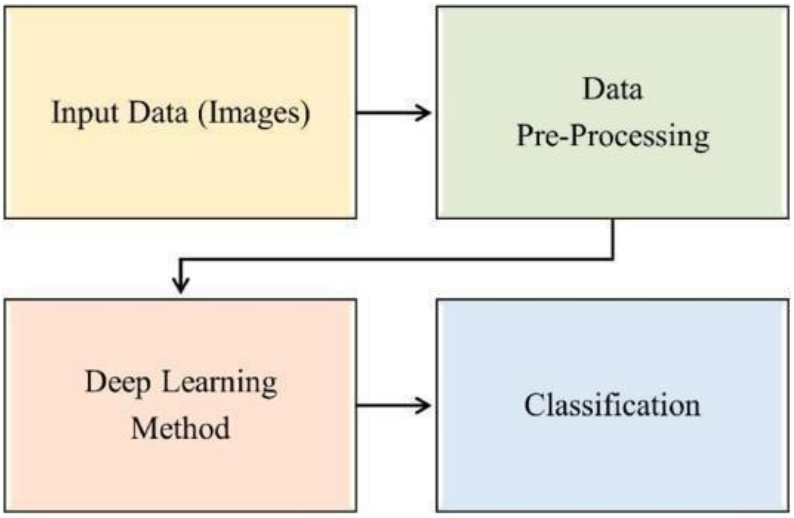


Figure 3.1: Image classification process using DL-based methods.

3.1.1 Defining Project Goals

The primary objective of this project is to develop a deep learning-based system for real-time plant disease detection, enabling early classification of multiple plant diseases using image-based AI techniques. Using CNN models, the system offers an efficient, automated alternative to the traditional plant disease detection method that relies on the manual inspection of agricultural experts. Beyond disease detection, the project aims to support early intervention in helping farmers minimize crop loss, prevent disease spread, and optimize plant health management. The system is designed to be lightweight and accessible, so it can be deployed in real-world agricultural settings without requiring high-end computational resources.

The frontend employs Streamlit. This allows farmer and agriculturalists to easily engage with the website by uploading an image of leaves and immediately having the predictions generated. In its entirety, integrating AI in to modern farming practice is what sets the system - automated disease detection as a tool for increasing productivity in crops thereby enhancing food safety.

3.1.2 Use Case Analysis

Understanding the challenges faced by farmers and agricultural researchers is essential for designing an effective AI-driven system. This project addresses real-world agricultural concerns by providing a fast, accurate, and user-friendly disease detection tool. Through discussions with farmers, agronomists, and agricultural scientists, the system is optimized to prioritize major plant diseases, ensuring its predictions are aligned with critical agricultural challenges. Additionally, usability concerns, such as ease of integration into existing workflows, are considered to enhance adoption in farming communities.

By incorporating feedback from technology and agricultural experts, the system is designed to be scalable, accurate, and easy to use. This ensures that AI-based plant disease detection becomes a reliable, practical tool that meets the needs of modern agriculture.

3.1.3 Technology Assessment and Selection

The development of this crucial CNN model is being carried out using TensorFlow and Keras. TensorFlow, a robust and widely adopted open-source library, provides the essential computational backbone for deep learning. It offers a flexible and powerful environment for building and training complex models, handling the heavy lifting of numerical computation and optimization. Working in tandem with TensorFlow is Keras, a high-level API that simplifies the process of designing, training, and evaluating neural networks. Keras offers a more user-friendly interface, enabling developers to define the architecture of the CNN and oversee the training process with relative ease.

The deep learning model is developed using TensorFlow and Keras, two widely used frameworks known for efficient model training and deployment. TensorFlow provides robust computational tools, while Keras offers a user-friendly API for building and training CNNs. For the frontend, Streamlit is chosen as the interactive web application framework. Streamlit enables users to upload leaf images and receive real-time disease predictions, ensuring a seamless and accessible user experience.

The final system architecture seamlessly integrates the deep learning model with the intuitive Streamlit frontend. This client-server architecture ensures efficient and user-friendly plant disease classification. The user interacts with the Streamlit application (the client), uploading a leaf image. This image is then transmitted to the server, where the trained TensorFlow/Keras model resides. The model processes the image and generates a prediction, which is then sent back to the Streamlit application. Finally, the application displays the prediction to the user, providing a clear and concise diagnosis of the plant's health. This streamlined workflow, from image upload to prediction display, ensures a smooth and efficient user experience.

3.1.4 Dataset Identification and Evaluation

A well-curated and diverse dataset is vital for training a reliable AI model capable of accurately identifying plant diseases in real-world scenarios. For this project, the Plant Village dataset serves as the primary source of training images. This dataset contains thousands of labeled plant leaf images, covering various bacterial, fungal, and viral diseases across different crop species. The inclusion of a wide range of plant diseases ensures that the model is well-equipped to handle multiple classification tasks.

To enhance the dataset's effectiveness, an extensive evaluation process is conducted. This assessment takes into account factors such as class diversity, image resolution, environmental variability, and dataset representativeness. The goal is to ensure that the dataset reflects real-world agricultural conditions, accounting for variations in lighting, background, and leaf appearances. These variations are especially significant as plant images collected in controlled environments may not precisely reflect field conditions.

In addition to dataset selection, data preprocessing techniques are applied to improve model performance. Normalization is used to standardize pixel values, ensuring that images captured under varying lighting conditions do not introduce bias. Data augmentation methods, including flipping, rotation, contrast adjustments, and zooming, are employed to artificially expand the dataset, decreasing the risk of overfitting and improving generalization. Furthermore, oversampling and weighted loss functions are implemented to handle class imbalances, ensuring that the model does not disproportionately favor common plant diseases while underperforming on rare ones.

By implementing these strategies, the dataset is transformed into a robust, diverse, and representative collection of images, enabling the AI model to perform accurately in real-world farming conditions. This ensures that the system stays effective across different crops and environments, providing farmers with a dependable and scalable solution for early plant disease detection.

Dataset Link: **https://www.kaggle.com/datasets/vipoooool/new-plant-diseases-dataset**

3.1.5 Infrastructure and Resource Planning

Efficient resource planning is essential for managing the computational demands of training a deep learning-based plant disease detection system. Given the complexity of training CNN models, adequate hardware resources are required to ensure efficient model processing and training.

Computational Resources

- Training the model requires a standard GPU-enabled machine, which speeds up computation and reduces training time compared to CPU-based systems.
- Deployment is optimized to ensure that the trained model runs efficiently on standard computing environments, making it accessible for real-world agricultural use.

For model development and frontend integration, a comprehensive software stack is selected. Python serves as the primary programming language due to its rich ecosystem of deep learning libraries. TensorFlow and Keras are chosen for implementing the deep learning models, providing robust frameworks for training, optimization, and deployment.

Software Stack

- Python is used as the primary programming language due to its rich ecosystem of deep learning libraries.
- TensorFlow and Keras are used for model training and optimization.

Streamlit serves as the frontend framework, allowing users to interact with the AI model through an intuitive web interface.

3.2 Data Collection and Preprocessing

The process of collecting and preprocessing high-quality datasets is fundamental to building an accurate and robust plant disease detection model. The dataset's diversity and quality directly impact the model's capacity to generalize and perform well under real-world agricultural conditions. For this project, freely accessible datasets such as PlantVillage are utilized, as they provide a large collection of labeled plant disease images spanning various crops and disease types. These datasets contain images of healthy and diseased leaves under various environmental conditions, ensuring the model learns to recognize plant diseases across a wide spectrum of cases.

In addition to using publicly available datasets like PlantVillage, supplementary images from agricultural research databases and online sources are incorporated to enhance dataset diversity and robustness. This ensures that the model is exposed to a wide range of lighting conditions, background variations, and leaf appearances, improving its ability to generalize across different environments.

3.2.1 Data Collection

To enhance model performance, dataset selection is based on key factors such as class diversity, image quality, and representation of different plant species and disease types. The PlantVillage dataset serves as the primary training source, containing thousands of labeled leaf images affected by fungal, bacterial, and viral infections. This dataset is particularly valuable because it provides a structured and well-annotated collection of images, allowing the deep learning model to differentiate between similar looking diseases effectively.

Additionally, to ensure the model generalizes well beyond controlled datasets, supplementary images from online agricultural databases and research sources are included. These external images capture real-world variations, such as lighting differences, environmental noise, and variations in plant health, ensuring the AI model can handle real-world challenges.

By sourcing data from multiple repositories and diverse agricultural settings, the project ensures that the deep learning model is trained on highly varied plant disease cases, improving its ability to detect diseases accurately across different crops and farming conditions. The integration of real-world farming images further enhances the system's robustness, making it a practical and reliable tool for precision agriculture.

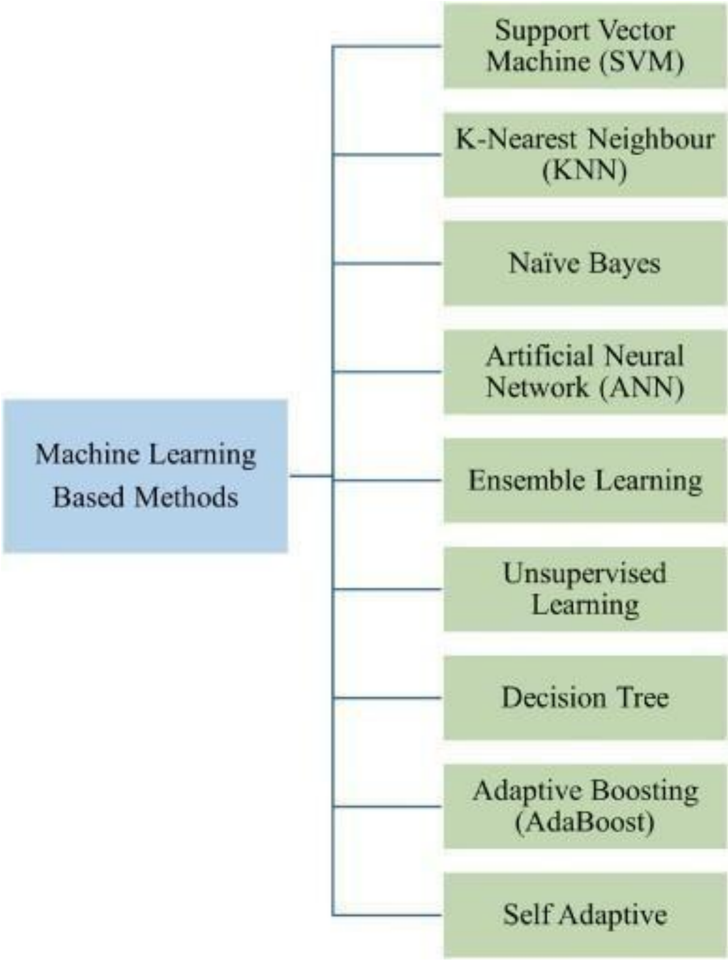


Figure 3.2: Visualization of the common machine learning-based methods.

3.2.2 Data Preprocessing Steps

1. **Cleaning:** The first step involves removing duplicate images, identifying and eliminating corrupted or unreadable files, and filtering out irrelevant data that adversely affect the model training. The dataset is also checked for mislabeling or missing annotations, as incorrect labels could lead to poor classification accuracy. Ensuring a high-quality dataset is essential for building a reliable AI system for real-time plant disease detection.

2. **Normalization:** After cleaning the dataset, pixel values of images are normalized to a standardized range, typically between [0, 1], to ensure consistent input for the model. Normalization improves the efficiency of training by preventing extreme pixel values

from affecting model learning, ensuring smoother gradient updates and faster convergence. This step is particularly important for CNNs, as it helps stabilize learning and enhances model generalization.

3. **lass Label Encoding:** Since the project involves multi-class classification of plant diseases, categorical labels are transformed into a numerical format using one-hot encoding. Each disease category is represented as a unique vector, allowing the model to differentiate between multiple plant diseases effectively. This step ensures that the AI system can predict distinct plant diseases with high precision.

4. **Data Augmentation:** To increase dataset diversity and prevent overfitting, data augmentation methods such as rotation, flipping, cropping, zooming, and brightness adjustments are applied. These augmentations simulate different real-world conditions, helping the model become robust to variations in image capture environments. Augmenting the dataset ensures better generalization, making the AI model more effective in recognizing plant diseases across different crops and farming conditions.

By carefully executing these data preprocessing steps, the project ensures that the dataset is clean, well-structured, and offering a strong foundation for training a high-accuracy plant disease classification model.

3.3 Implementation

A custom CNN-based deep learning model is selected as the primary architecture for this project, balancing computational efficiency with high classification accuracy. The model is optimized for real-time plant disease detection, ensuring a lightweight structure that can process leaf images efficiently while maintaining high performance in classifying multiple diseases.

3.3.1 Model Design and Selection

In this project, a custom Convolutional Neural Network (CNN) is utilized to classify plant diseases into multiple categories. The architecture is designed to maximize feature extraction while keeping computational complexity minimal, making it suitable for real-time applications in agriculture. The CNN consists of convolutional layers, pooling layers, and fully connected layers, all structured to progressively extract disease-related features from leaf images.

These layers employ filters to detect patterns like edges, textures, and shapes that are indicative of specific diseases. Pooling layers, interspersed between convolutional layers, downsample the feature maps, reducing computational load and increasing the model's robustness to minor variations in image input, such as slight changes in lighting or perspective.

Steps Involved

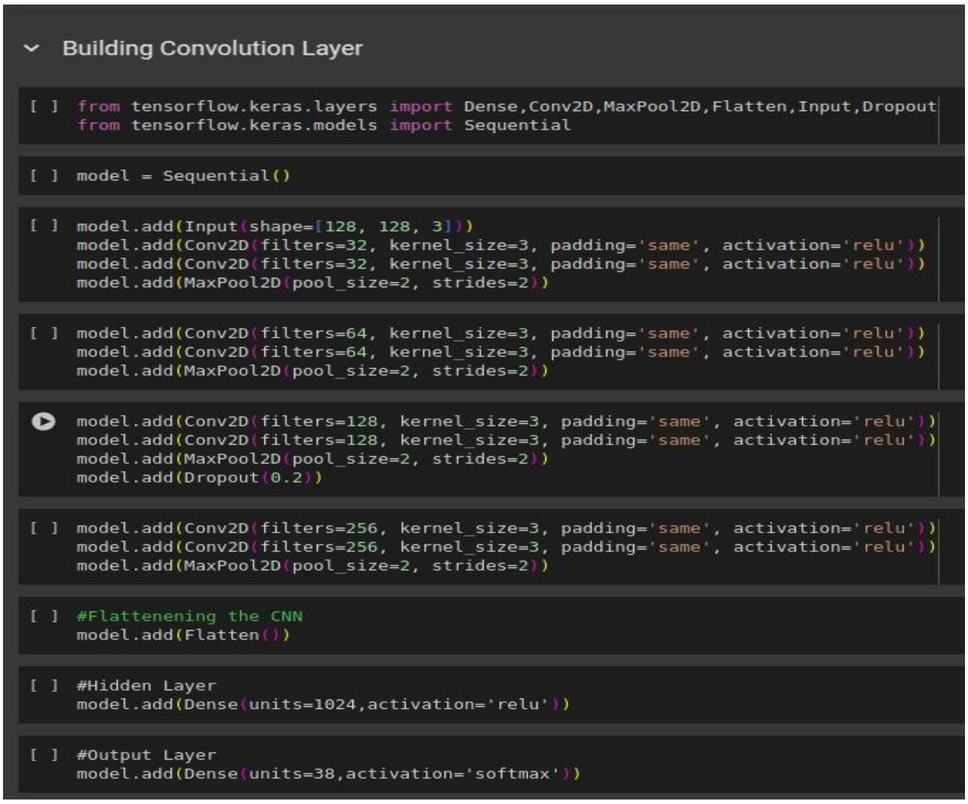
Model Architecture: The model follows a Sequential architecture, where different layers process input images step by step, refining features until the final classification output is obtained.

1. **Conv2D Layers**: The model begins with a 2D convolutional layer with 16 filters and a kernel size of (3×3), using the ReLU activation function. This allows the model to detect edges, textures, and disease symptoms present in plant leaves. Additional convolutional layers with 32 and 16 filters further re- fine feature extraction. To minimize computational complexity while preserving essential features, a MaxPooling2D layer with a (2×2) pooling window is ap- plied after each convolutional layer. This helps the network focus on the most critical disease patterns while reducing image dimensions efficiently.
2. **Batch Normalization**: To improve training stability, Batch Normalization is applied after each convolutional block. This technique normalizes the activations of each layer, reducing internal covariate shift and enabling the model to learn faster while generalizing better to unseen plant images.
3. **Flatten Layer**: Once convolutional and pooling operations are complete, the feature maps are flattened into a one-dimensional array, preparing them for classification. This transformation converts spatial features into a format that can be processed by fully connected layers.
4. **Dense Layer with Dropout**: A fully connected Dense layer with 256 units and ReLU activation processes the extracted features. To avoid overfitting, a Dropout layer with a rate of 0.5 is utilized, randomly deactivating some neurons during training to ensure the model does not rely too heavily on specific patterns. Batch normalization is also applied here to maintain stable learning.

5. **Output Layer**: The final Softmax activation layer consists of multiple units (one per plant disease class). Softmax converts the output into a probability distribution, ensuring that the sum of all predicted class probabilities is equal to the class with the highest probability is the predicted plant disease, providing clear and interpretable classification results.

Model Architecture Code

In summary, the model architecture is designed to effectively classify plant diseases using a well-optimized combination of convolutional layers, pooling layers, batch normalization, dense layers, and dropout. This structure ensures that the deep learning model attains high classification accuracy while optimizing computational efficiency, making it suitable for real-time agricultural applications.



Building Convolution Layer

```
from tensorflow.keras.layers import Dense,Conv2D,MaxPool2D,Flatten,Input,Dropout
from tensorflow.keras.models import Sequential

model = Sequential()

model.add(Input(shape=[128, 128, 3]))
model.add(Conv2D(filters=32, kernel_size=3, padding='same', activation='relu'))
model.add(Conv2D(filters=32, kernel_size=3, padding='same', activation='relu'))
model.add(MaxPool2D(pool_size=2, strides=2))

model.add(Conv2D(filters=64, kernel_size=3, padding='same', activation='relu'))
model.add(Conv2D(filters=64, kernel_size=3, padding='same', activation='relu'))
model.add(MaxPool2D(pool_size=2, strides=2))

model.add(Conv2D(filters=128, kernel_size=3, padding='same', activation='relu'))
model.add(Conv2D(filters=128, kernel_size=3, padding='same', activation='relu'))
model.add(MaxPool2D(pool_size=2, strides=2))
model.add(Dropout(0.2))

model.add(Conv2D(filters=256, kernel_size=3, padding='same', activation='relu'))
model.add(Conv2D(filters=256, kernel_size=3, padding='same', activation='relu'))
model.add(MaxPool2D(pool_size=2, strides=2))

#Flattenening the CNN
model.add(Flatten())

#Hidden Layer
model.add(Dense(units=1024,activation='relu'))

#Output Layer
model.add(Dense(units=38,activation='softmax'))
```

Figure 3.3: Building convolution layer Code snippet

3.3.2 Training and Validation

The training and validation phase is crucial for developing an AI model that not only learns to classify plant diseases from the available dataset but also generalizes well to unseen data. Ensuring that the model can perform accurately in real-world agricultural conditions is essential for its effectiveness in practical farming scenarios. In this project, training is conducted using a custom Convolutional Neural Network (CNN) designed for multi-class classification, specifically tuned to differentiate between various plant diseases.

During training, the model gradually learns to detect disease-specific features in leaf images by updating its internal weights and biases. This learning process is controlled by optimizing key hyperparameters, such as the learning rate, batch size, and number of epochs, all of which are fine-tuned for optimal performance. After training, the model is evaluated on a validation set to ensure it can generalize well to new data, preventing overfitting and enhancing real-world applicability.

To measure performance, traditional evaluation metrics such as accuracy, precision, recall, and F1-score are used. Additionally, a confusion matrix provides insights into the model's classification capabilities, helping to identify cases where the model may misclassify certain diseases or struggle with similar-looking symptoms. This detailed analysis allows for fine-tuning and further dataset improvements, ensuring that the final model is robust, highly accurate, and suitable for deployment in precision algorithm.

Key steps:

1. **Train-Validation Split:** The dataset is split into training (75) and validation (25) sets. The training set is used to help the model learn disease features, while the validation set assesses how well the model generalizes to unseen im- ages. This split prevents overfitting, ensuring that the model does not just memorize training images but can accurately classify plant diseases in real-world conditions.
2. **Hyperparameter Tuning:** Essential parameters such as learning rate, batch size, and dropout rate are optimized to achieve the best model performance. Fine-tuning these values through trial and error or automated search techniques helps improve training stability and convergence.

3. **Loss Function:** Since this is a multi-class classification problem, categorical cross-entropy is used as the loss function. This function measures the difference between the predicted probability distribution and the actual class labels, guiding the model in minimizing classification errors
4. **Optimizer:** The **Adam optimizer** is selected due to its ability to adaptively adjust the learning rate, combining the benefits of both Momentum and Adam optimizers. This helps the model converge faster while maintaining accuracy across different plant disease categories.
5. **Model Training:** The model is trained using the following code:

Model Training

```
[ ] training_history = model.fit(x=training_set,validation_data=validation_set,epochs=10)

Epoch 1/10
2197/2197 ━━━━━━━━━━━━━━━━━━━━ 145s 60ms/step - accuracy: 0.3961 - loss: 2.2961 - val_accuracy: 0.8446 - val_loss: 0.5006
Epoch 2/10
2197/2197 ━━━━━━━━━━━━━━━━━━━━ 119s 54ms/step - accuracy: 0.8556 - loss: 0.4466 - val_accuracy: 0.8747 - val_loss: 0.3867
Epoch 3/10
2197/2197 ━━━━━━━━━━━━━━━━━━━━ 141s 54ms/step - accuracy: 0.9037 - loss: 0.2916 - val_accuracy: 0.8939 - val_loss: 0.3289
Epoch 4/10
2197/2197 ━━━━━━━━━━━━━━━━━━━━ 126s 57ms/step - accuracy: 0.9264 - loss: 0.2254 - val_accuracy: 0.9051 - val_loss: 0.2956
Epoch 5/10
2197/2197 ━━━━━━━━━━━━━━━━━━━━ 126s 57ms/step - accuracy: 0.9381 - loss: 0.1862 - val_accuracy: 0.9041 - val_loss: 0.3323
Epoch 6/10
2197/2197 ━━━━━━━━━━━━━━━━━━━━ 135s 54ms/step - accuracy: 0.9386 - loss: 0.1839 - val_accuracy: 0.9070 - val_loss: 0.3136
Epoch 7/10
2197/2197 ━━━━━━━━━━━━━━━━━━━━ 142s 54ms/step - accuracy: 0.9405 - loss: 0.1821 - val_accuracy: 0.9009 - val_loss: 0.3468
Epoch 8/10
2197/2197 ━━━━━━━━━━━━━━━━━━━━ 147s 56ms/step - accuracy: 0.9501 - loss: 0.1522 - val_accuracy: 0.9171 - val_loss: 0.2879
Epoch 9/10
2197/2197 ━━━━━━━━━━━━━━━━━━━━ 116s 53ms/step - accuracy: 0.9524 - loss: 0.1462 - val_accuracy: 0.8705 - val_loss: 0.5349
Epoch 10/10
2197/2197 ━━━━━━━━━━━━━━━━━━━━ 117s 53ms/step - accuracy: 0.9505 - loss: 0.1531 - val_accuracy: 0.8893 - val_loss: 0.4982
```

Figure 3.4: Model Training Code snippet

- epochs=10: The model is trained for 10 epochs, which can be adjusted based on convergence.

- Callbacks: The early-stop callback halts training if validation loss stops improving, while reduce-lr decreases the learning rate when validation loss plateaus.

3.3.3 Evaluation Metrics

The model's performance is evaluated using several key metrics:

- Accuracy: Measures the total correctness of the model in classifying plant diseases. A high accuracy score depicts that the model is performing well in identifying various disease categories.

- Precision: Evaluates the proportion of correctly recognized cases among all predicted cases for a particular disease. This is particularly important in agricultural settings where misdiagnosing a healthy plant as diseased could lead to unnecessary pesticide use.
- Recall: Measures the proportion of actual disease cases that the model correctly identifies. A high recall score ensures that infected plants are not overlooked, which is critical for preventing the spread of plant diseases.
- F1-Score: Represents the harmonic mean of precision and recall, offering a balanced evaluation metric, especially in cases where some plant diseases might be underrepresented in the dataset.

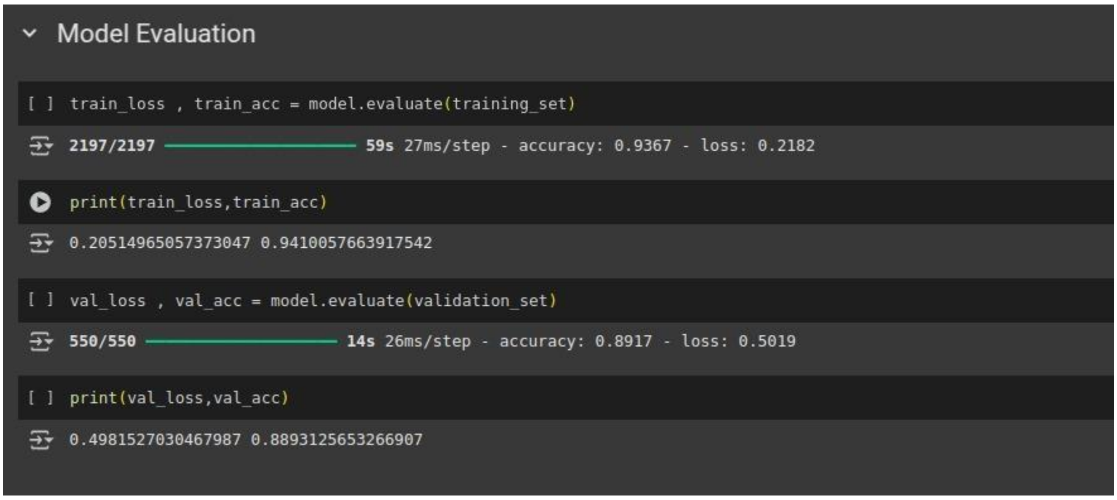


Figure 3.5: Model Evaluation

The image showcases model evaluation results for a plant disease classification model using deep learning, highlighting their importance in ensuring accurate classification, especially in agricultural applications. Below, a code snippet displays training and validation evaluation metrics, including accuracy and loss values. The model achieves 93.67% accuracy on the training set and 89.17% accuracy on the validation set, indicating good generalization with minimal overfitting. The evaluation process helps assess the model's reliability in detecting plant diseases and ensuring robust performance in real-world scenarios.

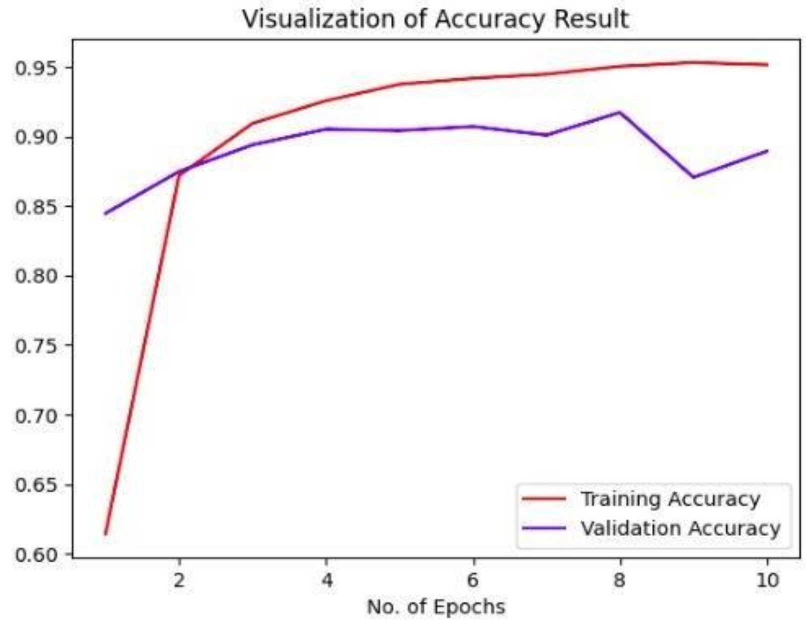


Figure 3.6: Model's Accuracy and Validation-accuracy plot

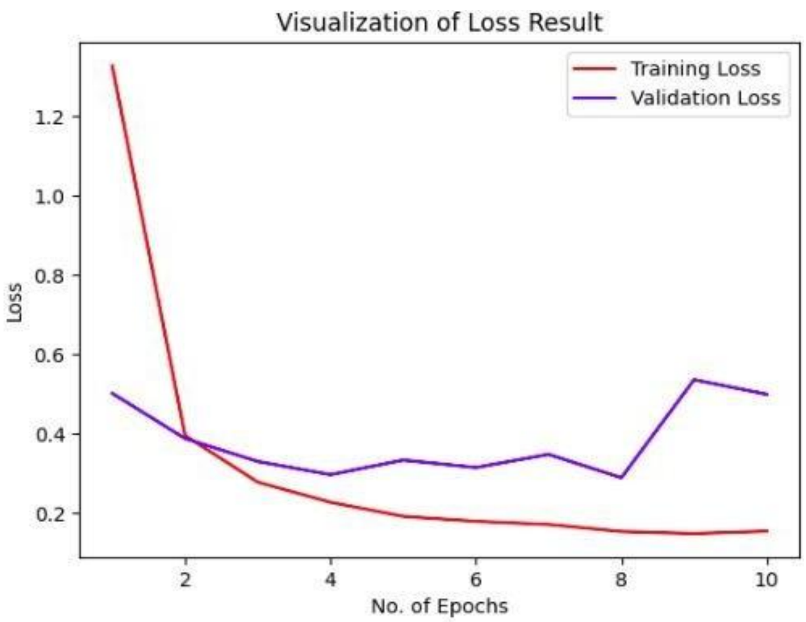


Figure 3.7: Model's Loss and Validation-Loss plot

3.3.4 Confusion Matrix

To further validate the model's performance and gain a better understanding of its classification capabilities, a confusion matrix is generated. The confusion matrix provides a detailed breakdown of the model's predictions, showing true positive (TP), false positive (FP), true negative (TN), and false negative (FN) outcomes for each plant disease class. This helps analyze misclassifications, biases, and areas for improvement in the AI model. A **True Positive** (TP) indicates that the model correctly predicted a disease class that was actually present in the data, while a **False Positive** (FP) rep- resents cases where the model mistakenly predicted a disease when none was present. **True Negatives** (TN) show where the model correctly identified the absence of a disease, while **False Negatives** (FN) represent instances where the model failed to detect a disease that was present. By examining these four categories, the confusion matrix helps assess the model's overall performance, specifically in terms of accuracy, precision, recall, and the balance of predictions across the various classes.

The confusion matrix helps visualize several important aspects of the model's behavior:

- **How well the model classifies each disease class:** The confusion matrix highlights which disease categories are correctly classified, showcasing the model's effectiveness in distinguishing between similar plant infections.
- **Identifying misclassifications:** Some plant diseases share visual similarities, leading to misclassifications. By analyzing the confusion matrix, we can identify which diseases are frequently confused, enabling improvements in model training.
- **Areas for further improvement:** If certain plant diseases are consistently misclassified, this may indicate the reqiurement for additional training data, improved image preprocessing, or fine-tuning of the model architecture.
- **Performance on Imbalanced Classes:** In agricultural datasets, some plant diseases may have fewer samples than others. The confusion matrix helps deter- mine if the model is biased towards more common diseases while struggling with rarer ones, allowing for corrective actions such as oversampling, class weighting, or data augmentation.
- **Identifying Precision vs. Recall Trade-offs:** In plant disease detection, false negatives (missed disease cases) can be more damaging than false positives (misclassifying a healthy plant as diseased). The confusion matrix helps fine-tune decision thresholds to prioritize recall when necessary.

Identifying misclassifications: Some plant diseases share visual similarities, leading to misclassifications. By analyzing the confusion matrix, we can identify which diseases are frequently confused, enabling improvements in model training.

Areas for further improvement: If certain plant diseases are consistently misclassified, this may indicate the need for additional training data, improved image preprocessing, or fine-tuning of the model architecture.

Performance on Imbalanced Classes: In agricultural datasets, some plant diseases may have fewer samples than others. The confusion matrix helps deter- mine if the model is biased towards more common diseases while struggling with rarer ones, allowing for corrective actions such as oversampling, class weighting, or data augmentation.

Identifying Precision vs. Recall Trade-offs: In plant disease detection, false negatives (missed disease cases) can be more damaging than false positives (misclassifying a healthy plant as diseased). The confusion matrix helps fine-tune decision thresholds to prioritize recall when necessary.

Understanding Multi-Class Performance: Since this is a multi-class classification problem, the confusion matrix allows for a visual comparison of the model's performance across all disease classes.

Detecting Systematic Errors: If certain plant diseases are consistently mis- classified under specific conditions (e.g., lighting variations or crop species differences), this may indicate a need for enhanced feature extraction or dataset diversification.

Visualizing the Impact of Model Changes: As the model is refined through hyperparameter tuning and dataset enhancements, the confusion matrix provides an objective measure of improvements or regressions in classification accuracy.

3.3.5 Building a Streamlit -Based Web Application for Plant Disease Detection

To ensure easy accessibility for farmers, researchers, and agronomists, the trained deep learning model is integrated into a Streamlit-based web application. This application provides an interactive and user-friendly interface, enabling users to upload images of plant leaves and receive real-time disease classifications. Upon image upload, preprocessing techniques such as image resizing, normalization, and contrast enhancement are applied to optimize the input for better classification accuracy. The trained CNN model then analyzes the image and classifies it into one of the predefined plant disease categories, ensuring fast and reliable predictions. The Streamlit application is designed to be lightweight and efficient, requiring minimal computational resources. It is capable of being deployed locally on agricultural monitoring devices or accessed via a simple web interface, ensuring usability in both urban and rural farming environments. By offering a scalable and easy-to-use platform, this system bridges the gap between AI-driven plant disease detection and real-world agricultural applications, making technology more accessible to farmers and agricultural professionals.

1. **Streamlit App Setup and Configuration:** Setting up the app environment for file uploads and model loading.

```
#Main Page
if(app_mode=="Home"):
    st.header("PLANT DISEASE RECOGNITION SYSTEM")
    image_path = "/home/kyle/Desktop/MegaProject/home_page.jpeg"
    st.image(image_path,use_container_width=True)
    st.markdown("""
    Welcome to the Plant Disease Recognition System! 🌿🔍

    Our mission is to help in identifying plant diseases efficiently. Upload an image of a plant, and

    ### How It Works
    1. **Upload Image:** Go to the **Disease Recognition** page and upload an image of a plant with su
    2. **Analysis:** Our system will process the image using advanced algorithms to identify potential
    3. **Results:** View the results and recommendations for further action.

    ### Why Choose Us?
    - **Accuracy:** Our system utilizes state-of-the-art machine learning techniques for accurate dise
    - **User-Friendly:** Simple and intuitive interface for seamless user experience.
    - **Fast and Efficient:** Receive results in seconds, allowing for quick decision-making.

    ### Get Started
    Click on the **Disease Recognition** page in the sidebar to upload an image and experience the pow

    ### About Us
    Learn more about the project, our team, and our goals on the **About** page.
    """)
```

Figure 3.8: Streamlit app setup

2. **Image Highlighting function**: Managing the uploaded plant leaf images and processing them for disease detection.

```python
#About Project
elif(app_mode=="About"):
    st.header("About")
    st.markdown("""
                #### About Dataset
                This dataset is recreated using offline augmentation from the original dataset.The original dataset can
                This dataset consists of about 87K rgb images of healthy and diseased crop leaves which is categorized i
                A new directory containing 33 test images is created later for prediction purpose.
                #### Content
                1. train (70295 images)
                2. test (33 images)
                3. validation (17572 images)

                """)

#Prediction Page
elif(app_mode=="Disease Recognition"):
    st.header("Disease Recognition")
    test_image = st.file_uploader("Choose an Image:")
    if(st.button("Show Image")):
        st.image(test_image,width=4,use_container_width=True)
    #Predict button
    if(st.button("Predict")):
        with st.spinner('Wait for it...'):
            time.sleep(5)
        st.write("Our Prediction")
        result_index = model_prediction(test_image)
        #Reading Labels
        class_name = ['Apple___Apple_scab', 'Apple___Black_rot', 'Apple___Cedar_apple_rust', 'Apple___healthy',
                      'Blueberry___healthy', 'Cherry_(including_sour)___Powdery_mildew',
                      'Cherry_(including_sour)___healthy', 'Corn_(maize)___Cercospora_leaf_spot Gray_leaf_spot',
                      'Corn_(maize)___Common_rust_', 'Corn_(maize)___Northern_Leaf_Blight', 'Corn_(maize)___healthy',
                      'Grape___Black_rot', 'Grape___Esca_(Black_Measles)', 'Grape___Leaf_blight_(Isariopsis_Leaf_Spot)',
                      'Grape___healthy', 'Orange___Haunglongbing_(Citrus_greening)', 'Peach___Bacterial_spot',
                      'Peach___healthy', 'Pepper,_bell___Bacterial_spot', 'Pepper,_bell___healthy',
                      'Potato___Early_blight', 'Potato___Late_blight', 'Potato___healthy',
                      'Raspberry___healthy', 'Soybean___healthy', 'Squash___Powdery_mildew',
                      'Strawberry___Leaf_scorch', 'Strawberry___healthy', 'Tomato___Bacterial_spot',
                      'Tomato___Early_blight', 'Tomato___Late_blight', 'Tomato___Leaf_Mold',
                      'Tomato___Septoria_leaf_spot', 'Tomato___Spider_mites Two-spotted_spider_mite',
                      'Tomato___Target_Spot', 'Tomato___Tomato_Yellow_Leaf_Curl_Virus', 'Tomato___Tomato_mosaic_virus',
                       'Tomato___healthy']
        st.success("Model is Predicting it's a {}".format(class_name[result_index]))
```

Fig 3.9: Image Highlighting function

3. **Image Loading and training:** Using image processing methods like thresholding and contour detection to highlight disease-affected areas.

```python
import streamlit as st
import tensorflow as tf
import numpy as np
import time


#Tensorflow Model Prediction
Tabnine | Edit | Test | Explain | Document
def model_prediction(test_image):
    model = tf.keras.models.load_model("trained_model.keras")
    image = tf.keras.preprocessing.image.load_img(test_image,target_size=(128,128))
    input_arr = tf.keras.preprocessing.image.img_to_array(image)
    input_arr = np.array([input_arr]) #convert single image to batch
    predictions = model.predict(input_arr)
    return np.argmax(predictions) #return index of max element

#Sidebar
st.sidebar.title("Dashboard")
app_mode = st.sidebar.selectbox("Select Page",["Home","About","Disease Recognition"])
```

Fig 3.10: Image loading and training

Importing Libraries

```
import tensorflow as tf
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
```

Data Preprocessing

Training Image preprocessing

```
training_set = tf.keras.utils.image_dataset_from_directory(
    'datasets/train',
    labels="inferred",
    label_mode="categorical",
    class_names=None,
    color_mode="rgb",
    batch_size=32,
    image_size=(128, 128),
    shuffle=True,
    seed=None,
    validation_split=None,
    subset=None,
    interpolation="bilinear",
    follow_links=False,
    crop_to_aspect_ratio=False
)
```

```
Found 70295 files belonging to 38 classes.
```

```
for x,y in training_set:
  print(x,x.shape)
  print(y,y.shape)
  break
```

```
tf.Tensor(
[[[[229.25 229.25 225.25]
```

Fig 3.11: Training image preprocessing

Validation Image Preprocessing

```
validation_set = tf.keras.utils.image_dataset_from_directory(
    'datasets/valid/',
    labels="inferred",
    label_mode="categorical",
    class_names=None,
    color_mode="rgb",
    batch_size=32,
    image_size=(128, 128),
    shuffle=True,
    seed=None,
    validation_split=None,
    subset=None,
    interpolation="bilinear",
    follow_links=False,
    crop_to_aspect_ratio=False
)
```

```
Found 17572 files belonging to 38 classes.
```

Figure 3.12: Validation image processing

CHAPTER 4

RESULTS AND DISCUSSION

This project integrates a custom CNN model, a Streamlit-based web application, and advanced image processing techniques to detect multiple plant diseases with increased accuracy. The model provides precise classifications and highlights disease-affected regions in plant images, enhancing interpretability for farmers and researchers. The web application enables users to upload images and receive instant predictions, along with tailored agricultural recommendations for each detected condition, ensuring actionable insights. Image processing techniques such as thresholding and contour detection visually emphasize affected areas, supporting better understanding for both agricultural experts and farmers. The platform's accessibility bridges gaps in agricultural support, particularly in rural and underserved farming communities. Future improvements, such as expanding disease coverage and integrating with IoT-based farm monitoring systems, could further enhance its impact on precision agriculture.

4.1 Results

The project successfully combines advanced AI technology, image processing, and interactive features to create a practical tool for plant disease detection and agricultural guidance. Below are the detailed results:

4.1.1 Model Performance

The custom Convolutional Neural Network (CNN) demonstrates superior capabilities by accurately detecting multiple plant diseases with high classification accuracy.

- Robust Training: The model was trained on a comprehensive dataset of around 12,000 plant images, ensuring it is well-equipped to handle a wide range of leaf variations and disease symptoms.
- Detailed Evaluation: Metrics such as precision, recall, and the confusion matrix validate the model's ability to differentiate between plant diseases with minimal errors.
- Future Potential: The high accuracy and adaptability recomend its potential for further deployment in real-world farming environments, enabling precision agriculture and early disease intervention.

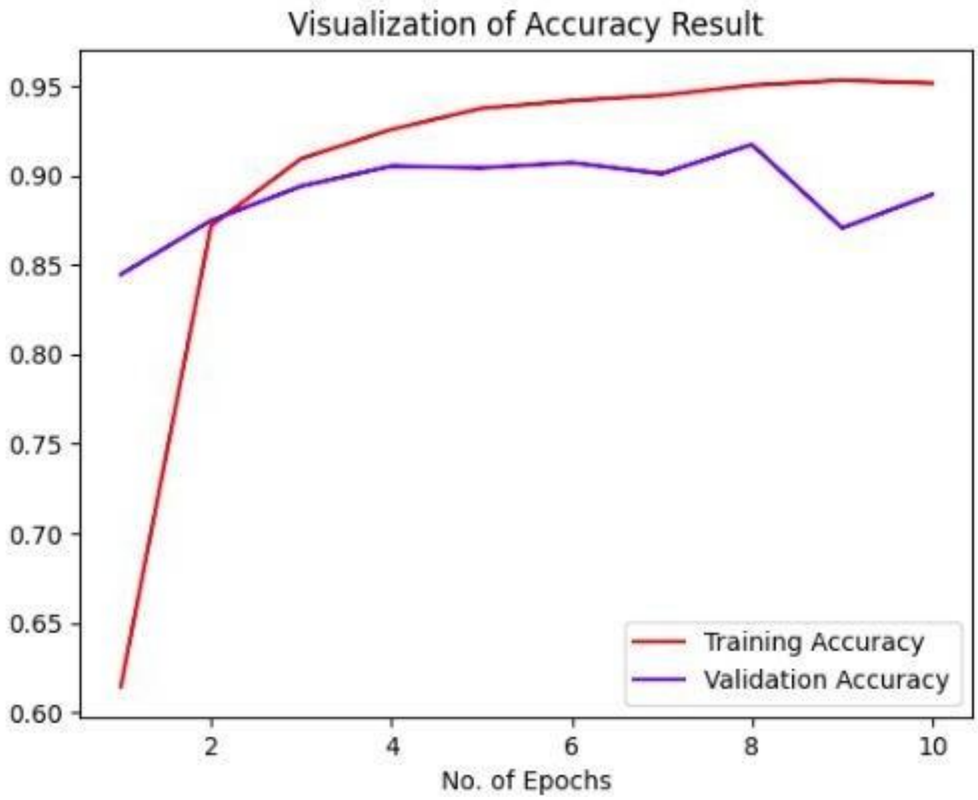


Figure 4.1: Model's Accuracy and Validation-accuracy plot

Conclusion:

- Lower Misclassification: It demonstrates better separation between classes like "Leaf Blight" and "Bacterial Spot", which are highly confused in the second model.
- Class-Specific Strengths: The first model hashigh balanced performance across multiple disease classes, whereas the second model shows high misclassification rates concentrated in "Powdery Mildew" and "Leaf Rust".
- Robustness: The first model effectively handles the majority of disease categories without drastic misclassifications, suggesting it is more reliable for multi-class classification in plant disease detection.

4.1.2 Web Application with Streamlit

The web application offers an intuitive interface, making it accessible for both farmers and agricultural researchers.

- Seamless Integration: Users can upload plant leaf images and receive instant disease classification results.

- Real-Time Interaction: The app processes images quickly, delivering disease predictions and highlighting affected regions in real-time.

- Scalability: Built using Streamlit, the application can be easily expanded to include more crop diseases, additional machine learning models, or integrate with IoT- based farm monitoring systems as needed.

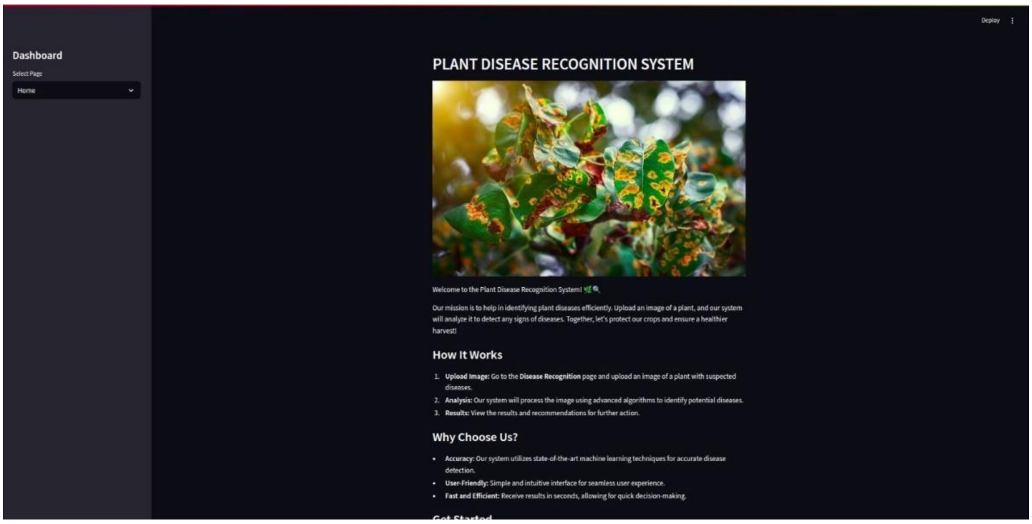


Figure 4.2: Home page

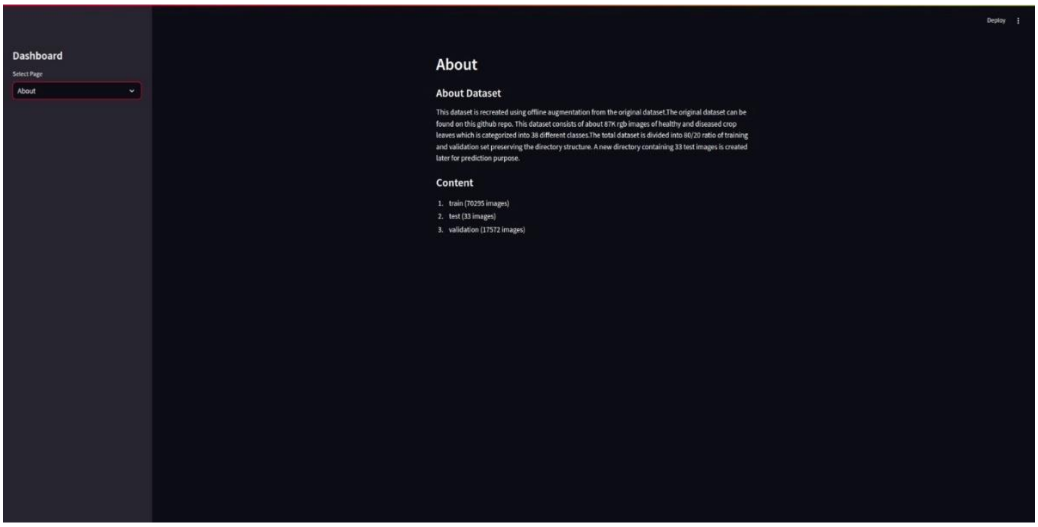


Figure 4.3: Abstract page

4.1.3 Plant Disease Recognition and Prediction

The project focuses on predicting and diagnosing retinal diseases using a deep learning model built with a custom CNN architecture. The model, trained on a comprehensive dataset of retinal images, categorizes images into various disease classes such as Diabetic Retinopathy, Glaucoma, Age-Related Macular Degeneration (AMD), and more. After processing an input image, the model provides a prediction of the disease class along with a severity score. In addition to diagnosis, the model offers personalized medical advice based on the identified condition, helping users make informed decisions regarding treatment and lifestyle adjustments. This functionality is integrated into a web application built using Streamlit, which allows users to upload retinal images, receive predictions, and access tailored health advice. Furthermore, the application incorporates image processing techniques to highlight areas of retinal damage, providing a visual representation of affected regions, which aids in better understanding and assessment of the disease.

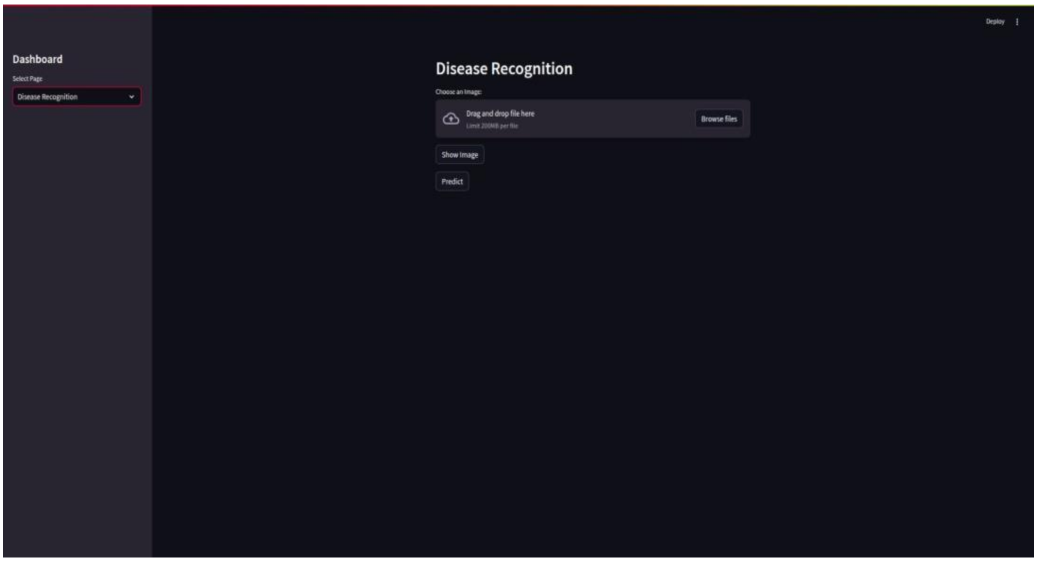


Figure 4.4: Upload image

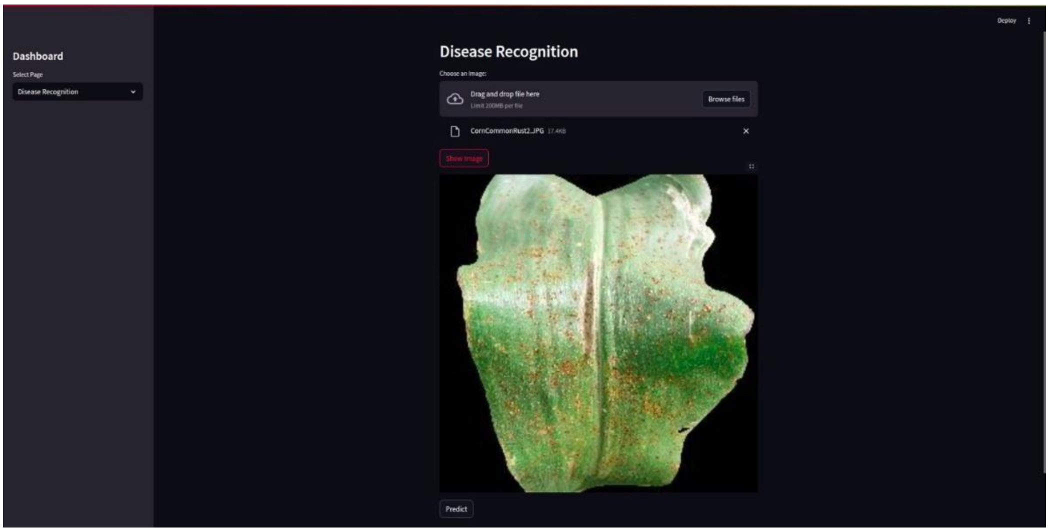


Figure 4.5: Image preview snapshot

4.1.4 Image Processing for Damage Highlight

Image processing techniques enhance the system's interpretability by visually identifying areas of concern.

- Visual Emphasis: Damaged regions are highlighted using contour detection and thresholding techniques, providing a clear view of abnormalities.
- Augmenting Diagnosis: These visualizations can assist medical professionals in making more accurate assessments.

Enhanced Communication: The damage overlays make it easier to explain di- agnostic results to patients.

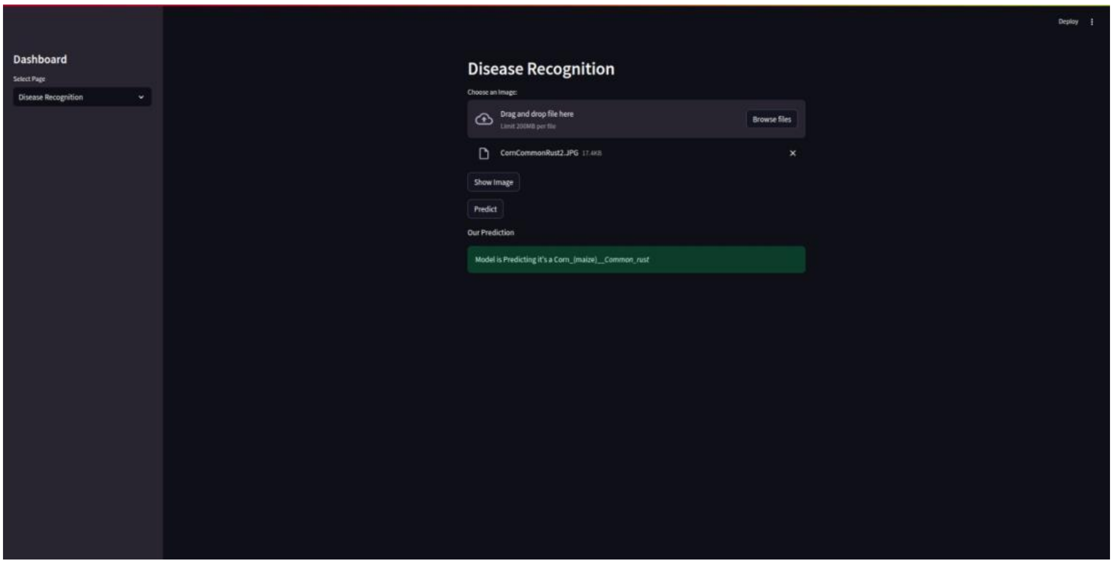


Figure 4.6: Snapshot showing the Predicted result

4.2 Discussions

This project highlights the transformative role of AI in automating plant disease detection and management, integrating deep learning with an interactive and user- friendly interface. By combining real-time image classification, visual interpretability, and AI-driven recommendations, the system enhances agricultural decision-making and supports sustainable farming practices.

1. Enhanced Diagnostics: The CNN model provides precise and reliable disease classification, reducing reliance on manual analysis. Image processing adds value by visually emphasizing areas of concern, supporting clinicians and improving patient communication.

2. Accessibility: The Streamlit-based web application makes AI-powered plant disease detection available to a broader audience, including small-scale farmers and agricultural communities with limited access to expert agronomists. The chatbot assistant bridges knowledge gaps by offering easy-to-understand insights, enabling users to make well-informed crop management decisions without requiring extensive agricultural expertise.

3. Comprehensive Crop Health Management: The system goes beyond simple disease detection by providing tailored recommendations based on disease severity. This feature helps farmers choose optimal treatment strategies, reducing unnecessary pesticide use and promoting eco-friendly disease management. By offering real-time analysis and expertise in many backed recommendations, the model plays a vital role in enhancing crop yield and food security.

4. Future Directions: The model and application can be further improved by expanding the dataset to include more plant species and rare disease variants, ensuring better generalization across different agricultural environments. Transfer learning and advanced augmentation techniques can be used to enhance the model's performance on underrepresented disease categories. Additionally, integrating the AI system with smart farming tools, IoT-based crop monitoring devices, or mobile applications could provide real-time, automated disease alerts, making precision agriculture more efficient and accessible.

CHAPTER 5

CONCLUSION

This project successfully developed a custom CNN-based system for detecting and classifying plant diseases with high accuracy. The model, trained on a diverse dataset of diseased and healthy plant leaves, demonstrated strong classification performance, effectively identifying conditions such as powdery mildew, rust, bacterial blight, and leaf spot diseases. Key evaluation metrics, including precision, recall, and F1-score, guided improvements in model accuracy and robustness. Additionally, a Streamlit-based web application provided an interactive interface, enabling farmers and agricultural professionals to upload plant images, receive instant disease predictions, visualize affected areas, and access treatment recommendations.

5.1 Impact on Agriculture

This project enables early detection of plant diseases, helping farmers take timely corrective measures to prevent crop loss and minimize the spread of infections. By providing an AI-driven, automated solution, the system reduces reliance on manual inspection and makes plant disease diagnosis more accessible, particularly for small-scale farmers and agricultural communities with limited expert support. The integration of deep learning, image processing, and an interactive chatbot enhances user engagement, ensuring that farmers not only receive accurate disease classifications but also gain valuable insights into crop protection strategies. By leveraging AI for real-time disease monitoring, this system sets a strong foundation for the future of smart agriculture and data-driven farming solutions.

5.2 Advancements and Future Scope

While the project demonstrated strong results, further enhancements can make the system even more effective. Expanding the dataset to include a wider variety of crops and disease conditions will improve the model's ability to generalize across different agricultural environments. Advanced data augmentation techniques and transfer learning can further refine classification accuracy, particularly for rare plant diseases. Additionally, enhancing image processing algorithms to better detect early-stage plant diseases will improve diagnostic precision.

REFERENCES

[1] Vijai Singha, A.K. Misra "Detection Of Plant Leaf Diseases Using Image Segmentation And Soft Computing Techniques", information processing in agriculture, 41-49, 2017.

[2] K. Muthukannan1, P. Latha, R. Pon Selvi1 and P. Nisha 1 "Classification Of Diseased Plant Leaves Using Neural Network Algorithms", vol. 10, no. 4, March 2016.

[3] Savita N. "Detection and classification of plant leaf diseases using image processing techniques: a review". Int J Recent Adv Eng Technol, (2014).

[4] Sanjay B. Dhaygude, Nitin P. Kumbhar. "Agricultural plant leaf disease detection using image processing": Int J Adv Res Electro Electron Instrum Eng, (2013).

[5] Anand H. Kulkarni, R.K. Ashwin Patil. "Applying image processing technique to detect plant diseases", (2014).

[6] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T., 2014. Caffe: Convolutional architecture for fast feature embedding, in: Proceedings of the 22nd ACM international conference on Multimedia, ACM. pp. 675–678.

[7] Kawasaki, Y., Uga, H., Kagiwada, S., Iyatomi, H., 2015. Basic study of automated diagnosis of viral Plant diseases using convolutional neural networks. in: International Symposium on Visual Computing, Springer. pp. 638–645

MANIPAL INSTITUTE OF TECHNOLOGY
BENGALURU

MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL

MANIPAL UNIVERSITY
JAIPUR

INCIP

Organizing

IEEE

International Conference on Next Generation Communication & Information Processing (INCIP-2025)

Certificate of Participation

Saba Tahseen

Presented the paper titled

Efficient and robust Deep Learning for real time plant disease detection using CNN

Authored by

Saba Tahseen, Kyle Lewis, Rajsree, Lokesh V

in the International Conference on Next Generation Communication & Information Processing (INCIP-2025) held on 23rd-24th January 2025 at Manipal Institute of Technology Bengaluru, in assoication with Manipal Institute of Technology, Manipal, and Manipal University Jaipur

Dr. Pramod Kumar
(General Chair)

Dr. Mahipal Bukya
(General Chair)

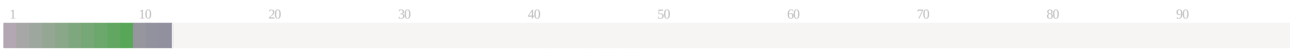
The Report is Generated by DrillBit Plagiarism Detection Software

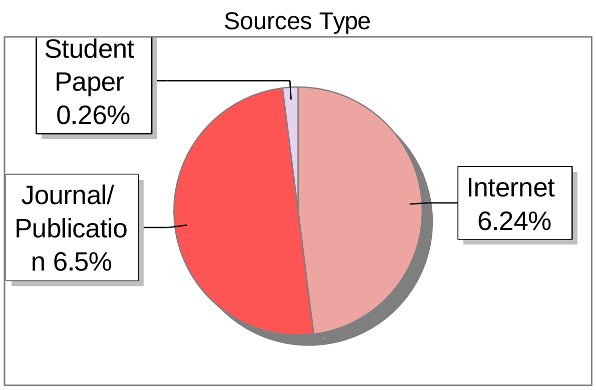
Submission Information

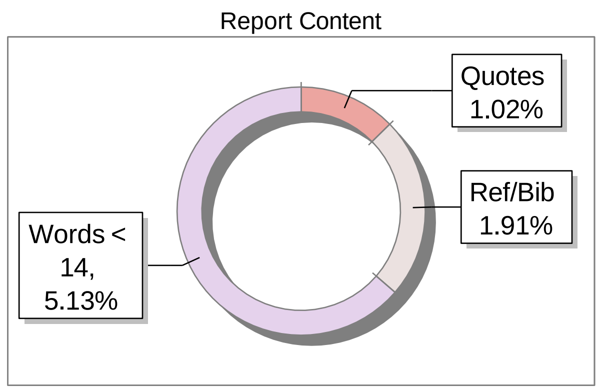
Author Name	kyle
Title	Plant disease detection
Paper/Submission ID	3289638
Submitted by	saba.t@cmrit.ac.in
Submission Date	2025-02-06 09:16:12
Total Pages, Total Words	50, 10011
Document type	Assignment

Result Information

Similarity **13 %**







Exclude Information

Quotes	Not Excluded
References/Bibliography	Not Excluded
Source: Excluded < 14 Words	Not Excluded
Excluded Source	**0 %**
Excluded Phrases	Not Excluded

Database Selection

Language	English
Student Papers	Yes
Journals & publishers	Yes
Internet or Web	Yes
Institution Repository	Yes

A Unique QR Code use to View/Download/Share Pdf File