**MULTI-MODAL GENERATIVE AI FOR PLANT DISEASE DETECTION**

***Submitted by***

### Chavali Vallabha Makarand [CB.EN.U4CSE21215]

### Kamsala Deepak Kumar [CB.EN.U4CSE21229]

### Yakkala Sai Supraja [CB.EN.U4CSE21369]

### Sai Siddharth Gimmedi [CB.EN.U4CSE21619]

***in partial fulfilment of the requirements for the degree of***

**BACHELOR** **OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**A close-up of a logo

AI-generated content may be incorrect.**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**AMRITA SCHOOL OF COMPUTING AMRITA VISHWA VIDYAPEETHAM COIMBATORE – 641 112**

**APRIL 2025**

**AMRITA VISHWA VIDYAPEETHAM**

**AMRITA SCHOOL OF COMPUTING, COIMBATORE – 641 112**

A close-up of a logo

AI-generated content may be incorrect.

**BONAFIDE CERTIFICATE**

This is to certify that the project report entitled **Multi-modal Generative AI for Plant Disease Detection** submitted by Sai Siddharth Gimmedi(CB.EN.U4CSE21619), Kamsala Deepak Kumar(CB.EN.U4CSE21229), Chavali Vallabha Makarand (CB.EN.U4CSE21215) and Yakkala Sai Supraja (CB.EN.U4CSE21369) in partial fulfillment of the requirements for the award of Degree of **Bachelor of Technology** in Computer Science and Engineering is a bonafide record of the work carried out under our guidance and supervision at the Department of Computer Science and Engineering, Amrita School of Computing, Coimbatore.

**Dr.Rajathilagam B. Dr. P.Bagavathi Sivakumar**

Guide Chairperson

Professor Department of CSE

Department of CSE

Evaluated on:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

INTERNAL EXAMINER EXTERNAL EXAMINER

# 

# DECLARATION

We the undersigned solemnly declare that the project Multi-modal Generative AI for Plant Disease Detection is based on our own work carried out during the course of our study under the supervision of Guide Dr.Rajathilagam B, Professor, Computer Science and Engineering, and has not formed the basis for the award of any other degree or diploma, in this or any other Institution or University. In keeping with the ethical practice in reporting scientific information, due acknowledgement has been made wherever the findings of others have been cited.

### Chavali Vallabha Makarand

### Kamsala Deepak Kumar

### Yakkala Sai Supraja

### Sai Siddharth Gimmedi

**ABSTRACT**

Multimodal generative AI techniques are revolutionizing agriculture, providing unprecedented innovations in disease detection and plant management. In this paper, we present a novel method that combines Convolutional Neural Networks (CNNs) with Contrastive Language-Image Pretraining (CLIP) to build a multimodal generative AI system for agro-diagnostics. Utilizing the PlantDoc dataset, which covers a broad spectrum of crop diseases, we train and compare several CNN architectures for disease classification. Out of these, ResNet proves to be the best option owing to its enhanced performance on our data.

Our suggested system will perform precise predictions about diseases, find symptoms, and suggest appropriate prescriptions. Integrating CLIP helps improve the diagnosis functionality because natural language is facilitated for input querying and multimodel inference. Agricultural experts and farmers can upload images of infected plants and, at the same time, ask the system based on text-based descriptions, thus providing a well-rounded interpretation of plant diseases.This aspect promotes diagnostic accuracy and facilitates accessibility, thus making it a valuable resource for different types of users, even those who lack technical skills.

The visual inputs are analyzed by the CNN module for precise disease detection, while a T5 model analyzes and makes contextual decisions from textual inputs. This synergy ensures users are presented with personalized, evidence-informed recommendations applicable to their own farm issues.

Keywords:Multimodal AI, Plant disease detection ,CNN ,CLIP, FLAN-T5 , LLM, Chatbot

**ACKNOWLEDGEMENT**

We would like to express our deep gratitude to our beloved Satguru Sri Mata Amritanandamayi Devi for providing the bright academic climate at this university, which has made this entire task possible. This acknowledgment is intended to thank all those people involved directly or indirectly with our project. We would like to thank the Board of Management for providing us with the necessary infrastructure required for the completion of the project. We express our thanks to Dr. Vidhya Balasubramanian, Principal, Amrita School of Computing, Dr. Bagavathi Sivakumar P, Chairperson, Department of Computer Science and Engineering and all the faculty members of the Department of Computer Science and Engineering for their valuable help and support during our study.

We wish to record our deep sense of gratitude and profound thanks to our research

supervisor Dr.Rajathilagam B., Professor, Department of Computer Science and Engineering, Amrita School of Computing, Coimbatore, for her keen interest, inspiring guidance, constant encouragement with our work during all stages, to bring this project into fruition. We feel extremely grateful to Dr. T R Swapna, Ms. K R Bindu, Ms. K Nalinadevi, Dr. T Bhagyammal, Dr. Manjusha, Ms. Bharathi D, Ms. R Manjusha, Mr. Rajeshwar Yadav for their feedback and encouragement which helped us to complete the project.

We would also like to thank the entire fraternity of the Department of Computer Science and Engineering. We would like to extend our sincere thanks to our family and friends for helping and motivating us during the project work. Last but not the least, we thank God for his blessings which made our project a success.

### Chavali Vallabha Makarand

### Kamsala Deepak Kumar

### Yakkala Sai Supraja

### Sai Siddharth Gimmedi

**TABLE OF CONTENTS**

| **CHAPTER** | **TITLE** | **PAGE NO** |
| --- | --- | --- |
|  | **ABSTRACT** | **4** |
|  | **LIST OF TABLES** | **9** |
|  | **LIST OF FIGURES** | **10** |
|  | **LIST OF ABBREVIATIONS** | **11** |
| **1** | **INTRODUCTION** | **12** |
| 1.1 | Introduction to the problem | 12 |
| 1.2 | Problem Definition | 13 |
| **2** | **LITERATURE SURVEY** | **14** |
| 2.1 | Survey | 14 |
| 2.2 | Summary of the survey and findings | 19 |
| 2.3 | Motivation and Challenges | 20 |
| 2.4 | Objectives of the work | 21 |
| **3** | **PROPOSED WORK** | **22** |
| 3.1 | Architecture of the System | 22 |
| 3.1.1 | Phase 1: Disease Prediction Based on an Image | 22 |
| 3.1.2 | Phase 2: Multi-Modal Query Alignment (CLIP) | 23 |
| 3.1.3 | Phase 3: Contextual Text Generation (LoRA-FLAN-T5) | 23 |
| 3.1.4 | Final Output | 23 |
| 3.2 | Algorithm Design | 24 |
| 3.2.1 | CLIP | 24 |
| 3.2.2 | BLIP | 25 |
| 3.2.3 | CNN + CLIP | 26 |
| 3.3 | Data sets for the study and Platform | 30 |
| 3.3.1 | Dataset | 30 |
| 3.3.2 | Platform and specifications | 34 |
| **4** | RESULTS AND INFERENCES | **35** |
| 4.1 | Metrics for evaluation | 35 |
| 4.2 | Parameters settings | 36 |
| 4.2.1 | CNN(ResNet-18 Based Image Encoder | 36 |
| 4.2.2 | CLIP(Contrastive Language Image Pretraining) | 37 |
| 4.2.3 | LoRA-FLAN-T5(chatbot text generator) | 37 |
| 4.3 | Results and discussion | 38 |
| 4.3.1 | Experiment 1: Comparison of Four CNN Models | 38 |
| 4.3.2 | Experiment 2: Integrating CLIP with ResNet-18 | 40 |
| 4.3.3 | Experiment 3: Evaluating CLIP Model on Our Dataset | 42 |
| 4.3.4 | Experiment 4: Models Evaluated : Alpaca and FLAN-T5 | 43 |
| 4.3.5 | Experiment 5: Chatbot Integration using FLAN-T5 | 45 |
| **5** | **CONCLUSION AND FUTURE ENHANCEMENTS** | **47** |
| **5.1** | Summary of the work | 47 |
| **5.2** | Contributions in the work | 48 |
| **5.3** | Future enhancements | 49 |
| **6** | **REFERENCES** | **50** |
| **7** | **LIST OF PUBLICATIONS** | **52** |

**LIST OF TABLES**

| **TABLE NO** | **TITLE** | **PAGE NO** |
| --- | --- | --- |
| 1 | Input and Output of CLIP model | 25 |
| 2 | Input and Output of BLIP model | 26 |
| 3 | Comparison of CNN models | 38 |
| 4 | Performance and comparison of LLMs used | 43 |

**LIST OF FIGURES**

| **FIGURE NO** | **TITLE** | **PAGE NO.** |
| --- | --- | --- |
| 1 | Architecture diagram | 22 |
| 2 | CNN+CLIP Plant based diagnosis | 29 |
| 3 | A sample from the dataset , left one is Corn Leaf Blight, in the center is Tomato Mold leaf and on right is Apple Rust leaf. | 30 |
| 4 | JSON file sample text for CLIP model | 32 |
| 5 | Custom dataset for chatbot | 33 |
| 6 | Graphs for CNN models | 39 |
| 7 | Graph for CNN+CLIP model | 41 |
| 8 | Alpaca performance | 44 |
| 9 | FLAN-T5 performance | 44 |
| 10 | Final Output of the chatbot | 46 |

**LIST OF ABBREVIATIONS**

**CLIP** Contrastive Language-Image Pre-Training

**BLIP** Bootstrap Language-Image Pre-Training

**CNN** Convolutional Neural Network

**LLM** Large Language Model

**GAN** Generative Adversarial Network

**LORA** Low Rank Adaptation

**FLAN**  Fine-tuned LAnguage Net

**Introduction**

**1.1 Introduction to the Problem**

Over the past decade, plant disease has emerged as a critical problem in the agricultural sector, having a considerable effect on the crop yield, availability of food, and the livelihood of farmers. Plant diseases are highly infectious and, if not diagnosed in the initial phase, may lead to extensive crop loss, financial loss, and food shortages.Conventional plant disease diagnosis is based on visual examination and expert agronomists, which is laborious, subjective, and inaccessible to farmers living in rural or developing regions.This requires smart systems with the capability to detect and diagnose at an economic cost within a timely fashion.

With the evolution of Artificial Intelligence (AI), especially deep learning, there has been a paradigm shift in plant disease detection and management. Deep learning models such as Convolutional Neural Networks (CNNs) have been shown to be effective tools in the extraction of implicit patterns from leaf images for classification. Models like VGG, ResNet, and InceptionV3 have been proven to have high accuracy in the detection of a broad spectrum of plant diseases from image datasets. However, even though they are highly effective, they are image-based and do not consider contextual information or do not give user-friendly output of results. Mobile-friendly models and lightweight frameworks have also been proposed for the detection purpose so that it becomes feasible in real-time and on-the-go applications[1][2]. Multimodal solutions have been suggested in the last couple of years to bridge these gaps. Such systems integrate visual and textual data, allowing not just accurate classification but also explanatory description of symptoms, causes, and treatments of diseases.Solutions that combine CNNs with transformers and cross-modal fusion techniques—such as CLIP and vision-language models—made it possible for more interactive and stronger diagnosis systems[3][4].Such solutions increase the understanding of the model by integrating visual patterns and description data, thereby allowing users to receive detection and explanation.

Agriculture itself is threatened by climate change, population increase, and resource scarcity, and as such, there is an increasingly pressing need for intelligent, explainable, and effective disease detection systems. This project will design a multimodal deep learning system not only capable of identifying diseases from plant images with accuracy but also offer an interactive interface via which users can inquire and learn about the diagnosed diseases. Through the combination of image classification and natural language generation, this system seeks to bridge the gap between sophisticated AI technology and real-world agricultural application, offering improved decision-making, crop health, and farmer assistance.

**1.2 Problem Definition**

Disease in plants constitutes a potential risk to world agriculture, involving the loss of astronomical amounts of crop yield, quality, and farm incomes. Under conditions of poor access to expert agricultural equipment and services, this effect is very pronounced. Disease diagnostic traditional methodologies that primarily employ farmers' or agricultural officers' visual checks are subjective, require much time, and suffer from inaccuracies. With more farming and climate change introducing new problems, the need for a more accurate, effective, and affordable solution has become a priority. Early diagnosis is critical to successful treatment, but without early diagnosis, the rapid spread of disease tends to happen, resulting in reduced productivity and economic losses.

Latest advancements in artificial intelligence, especially computer vision and deep learning, have made it possible to create models that can detect plant diseases automatically from images. CNN-based models have demonstrated strong accuracy in classifying diseases from leaf images, and hence, they are an effective tool for monitoring plant health.However, these suggestions typically stop at a name of the disease and don't provide further support or information. They are essentially not designed for user input, result interpretation, or interoperability with other types of information such as text-based information, limiting their potential in real field agricultural settings.

This project hopes to address such problems by creating a multimodal system that not only detects the disease from images of plants with a CNN-CLIP model but also provides users with a chatbot using a LoRA fine-tuned FLAN-T5 model to interact with. Once a disease is identified, the user can query symptoms, causes, and treatments using natural language and get clear, informative answers. This provides an intelligent, easy-to-use, and farmer-friendly platform that unites the power of AI with the pragmatism of field application. By filling the gap between detection and knowledge, the system enables farmers to make knowledgeable decisions, take timely action, and ultimately enhance crop health and productivity.

**LITERATURE SURVEY**

**2.1 Survey**

In recent years, AI-driven solutions have revolutionized agricultural practices, particularly in crop disease detection and farm management. Various research studies have demonstrated the effectiveness of deep learning models, Generative AI, and Large Language Models (LLMs) in enhancing agricultural efficiency. The following is a review of key papers related to AI-based advancements in farming:

Dhavale et al. developed an end-to-end AI architecture that boosts the recognition of crop diseases through integrating GANs for image expansion, CNNs for disease classification, and Large Language Models (LLMs) for supporting farmers via a chatbot platform [1]. It entailed augmentation of the 2,075-image dataset to 3,075 images via GANs, thus making model training strong and generalized.The CNN also achieved a very high accuracy of 99.7% and a low loss rate of 0.0167, emphasizing the model's high accuracy for disease classification.To ease more user interaction, the study used a LangChain-integrated chatbot powered by LLAMA and FAISS for efficient semantic retrieval and conversational assistance for farmers. This work shows the real-world applicability of AI not just to automate disease diagnosis but also to provide actionable farm guidance, enabling farmers to employ intelligent decision-support systems [1].

Madaan et al. developed a mobile application that uses AI-powered real-time crop disease detection and advisory by integrating lightweight deep models with Large Language Models (LLMs) [2]. MobileNetV2 and ResNet18 architectures were used in the study on a diverse dataset of 39,131 images across 21 classes of diseases.MobileNetV2 attained a better accuracy of 97–99% over ResNet18, while the model size remained much smaller (9.8 MB as against ResNet18's 94 MB), hence being more ideal for mobile deployment. Moreover, the incorporation of an LLM-based recommendation system allowed the app to provide farmers with personalized suggestions for disease prevention and treatment. This research highlights the utility of introducing AI-driven applications on mobiles for real-time disease detection and farm advice for remote farming operations [2].

Alatawi et al. utilized the VGG-16 convolutional network for plant disease classification with widely accepted PlantVillage dataset comprising 15,915 images on 19 disease classes [3]. The model achieved 95.2% accuracy and a test loss of 0.4418, proving VGG-16's ability to act as an effective baseline for image-based plant disease detection. The research explains the model's strength as resulting from the richness in the dataset, while at the same time recognizing issues such as background complexity and inconsistency of illumination in natural conditions. This study sets a baseline for deep learning methods in crop monitoring, as VGG-16's performance in organized datasets is noted as being effective with the possibility of adaptability towards field applications [3].

Chia et al. explore the use of Large Language Models (LLMs) in agricultural water management through the deployment of a smart virtual assistant that utilizes Retrieval-Augmented Generation (RAG) for irrigation optimization [4]. The research illustrates how RAG technology enables effective information retrieval without the requirement for significant fine-tuning, thus enhancing adaptability and minimizing the likelihood of model hallucinations.This capability enables farmers to obtain more contextual and trustworthy answers, enhancing decision-making. The technology of virtual assistants is also indicative of the scalability potential across applications such as enterprise data management, legal search, and medicine. The research brings into focus the contribution of LLM-facilitated assistants in transforming agronomic knowledge democratization and making advanced agronomic information available to farmers by voice interfaces [4].

Liu et al. propose a Visual Information Guided Multi-modal (VIG-MM) model for plant disease anomaly detection by combining visual features of plant images and semantic textual descriptions with a vision-language contrastive learning method [5]. The model maps both modalities to the shared embedding space, allowing more contextual knowledge and precise anomaly detection. Experimental results show a 4.2% accuracy gain over image-only baselines on benchmarks such as PlantVillage and PlantDoc. In addition, the VIG-MM model achieves better generalization in few-shot and zero-shot learning settings, successfully detecting unseen plant diseases. Also, the incorporation of descriptive language prompts further anomaly discrimination by embedding rich contextual information. This research emphasizes the capability of multi-modal AI systems to improve plant disease detection, especially under low data conditions [5].

Abbas et al. provide a comprehensive review highlighting the application of artificial intelligence (AI) in the detection of plant diseases in prominent crops such as tomato, chilli, potato, and cucumber [6]. The study presents a structured AI pipeline with steps including image acquisition, preprocessing, segmentation, feature selection, and classification.It provides comparative analysis of the performance of machine learning (ML) models like support vector machines (SVMs) and deep learning (DL) methods like convolutional neural networks (CNNs), highlighting their efficiency in disease detection.One of the salient features of this review is its enumeration of publicly accessible datasets that facilitate future AI research for agriculture. The paper further outlines the most common challenges such as data quality, model generalization issues, and dependence on large annotated datasets. With the proposal of the incorporation of multimodal data and the building of more robust models, the research points towards promising directions for the future of AI-based crop health monitoring [6]. Sunil and Jaidhar establish a robust deep learning framework with an eye to improving plant disease diagnosis, especially in cases where the datasets are imbalanced and visually indistinguishable symptoms [7].

The method focuses on an ensemble model that blends AlexNet, ResNet, and VGGNet and uses it over seven plant leaf image datasets to minimize misclassification. A Multilevel Feature Fusion Network (MFFN) is proposed that incorporates adaptive attention mechanisms for ensuring model robustness through efficiently combining multi-layer features. The model also uses U²-Net for accurate background elimination and EfficientNetV2 for classification, enhancing performance on complicated background images dramatically.The model exhibits 99% accuracy in tomato plant datasets and 98.28% in cardamom leaf datasets, and it performs well in practical agricultural application. The paper indicates the importance of ensemble and attention-based mechanisms in developing high-accuracy plant disease detection systems [7].

Lachure and Doriya propose a lightweight Convolutional Neural Network (CNN) model for enhancing the efficiency of plant disease detection towards realistic agricultural application [8]. CNN parameter optimization by grid search hyperparameter tuning is highlighted as research focus to enable the design of efficient models to be employed within resource-scarce scenarios.The models are made up of 3-layer and 4-layer CNN structures with dimensions of 1.81 MB and 2.77 MB, respectively.These models achieved high accuracies of 99.66% for the PlantVillage dataset and 98.46% for the cotton leaf disease dataset, with low loss values of 0.01389 and 0.06411, respectively. The compact model size and high prediction capability make this approach adequate for early-stage disease detection and enable the development of sustainable agriculture via the capacity to operate in low-resource settings [8].

Khan et al. introduce a better method for plant disease detection using a better version of YOLOv8, a widely used deep learning architecture [9].Their approach consists of advanced data augmentation techniques to address the challenges posed by sparse and imbalanced training data, thereby enhancing the generalization ability of the model.The model reaches a remarkable accuracy of 98.75% when tested against the PlantVillage dataset, reflecting its strength in identifying a vast range of plant diseases with great accuracy. Further, the paper compares its model's performance with that of state-of-the-art methods available at the time and reflects on its superiority in terms of classification accuracy as well as computational power, thereby establishing itself as a viable choice for real-time plant disease diagnosis [9].

Saletnik et al. present the use of Raman spectroscopy to detect plant disease in upcoming agriculture that has the potential to improve crop yield at lower environmental costs [10]. Raman spectroscopy provides a non-destructive and species-independent way of measuring plant health and diagnosing early-stage biotic and abiotic stresses. By linking laboratory information to actual agricultural farm operations, it allows farmers to perform precision treatments like site-specific application of chemicals and fertilizers, which can effectively reduce crop loss (up to 30% biotic and 70% abiotic stress). Advances in handheld Raman spectrometers allow field measurements in real time, which render the technology more action-based and friendly to use by farmers. The article highlights the need for a synergy of farmers and scientists to facilitate the use of Raman spectroscopy in increasing agricultural productivity and sustainability [10].

Rizwan et al. introduce a computationally light Convolutional Neural Network (CNN) model for detecting plant diseases that is tailored for small-scale farmers with minimal computational power [11]. The model obtains a 96.86% average accuracy in optimizing performance and complexity and trading off between factors like FLOPs, parameters, computation time, and model size. This paper compares the new model to other recent models such as MobileNet V2, ResNet50 V2, and Inception V3 and proves that the proposed CNN provides the best trade-off between accuracy and efficiency. Through the creation of a low-cost and affordable solution, this study hopes to present small-scale farmers with a plant disease detection tool that is automatic and not reliant on high-capacity computational assets, so it can be very easily used across large numbers in resource-poor environments [11].

Ouamane et al. suggest the use of a state-of-the-art Vision Transformer (ViT) model for detecting plant disease, with the need for systemic parameter tuning to maximize accuracy and efficiency [12]. By thorough testing of parameters like patch sizes, image resolutions, embedding dimensions, transformer block depth, attention heads, and feedforward layer dimensions, they found an optimal setting of image size = 224×224, patch size = 16, embedding dimension = 512, depth = 6, number of heads = 8, and feedforward dimension = 1024.This environment led to an impressive 99.77% accuracy on the PlantVillage dataset, and cross-dataset verification showed robustness on Taiwan Tomato and BananaLSD datasets. The ViT model beat traditional CNN architecture like VGG19 and AlexNet, delivering better accuracy, effectiveness, and scalability. Additionally, Grad-CAM approaches were also utilized for saliency map visualizations to enhance interpretability of the model by selecting crucial areas of diseased plant leaves.The study demonstrates the model's performance across several crops and conditions and presents it as a suitable candidate for real-world deployment in fields [12].

Zhao et al. propose a new system that combines Large Language Models (LLMs), Agricultural Knowledge Graphs (KGs), and Graph Neural Networks (GNNs) for the enhancement of plant disease detection accuracy and efficiency in smart agriculture [13]. The system uses a dynamic graph attention mechanism to increase precision, recall, and accuracy with outstanding metrics of 0.94, 0.92, and 0.93, respectively.By utilizing the optimized graph loss functions that penalize incorrect predictions, the model improves its ability to detect complex disease patterns. The integration of neural-symbolic reasoning, few-shot learning, and representation learning addresses data scarcity, and the disease classification is improved. This approach offers practical recommendations for real-world elaeagnus angustifolia disease detection to allow farmers to obtain actionable insights for making right decisions in agricultural management [13].

Kaushik et al. propose a computationally effective plant disease detection method using a Depth-Wise Separable-Based Adaptive Deep Neural Network (DSDNN) architecture, balancing high accuracy and computational effectiveness [14]. The study points to the use of data preprocessing algorithms such as Gaussian filtering and normalization for noise filtering, followed by graph clustering on the basis of enthalpy to effectively segment images. The extraction of the features was done using GLCM and color histogram methods for acquiring the leaf texture and color features. The proposed approach achieved an impressive accuracy of 99% for classifying diseased and healthy potato leaves. The system was tested on the datasets of PlantVillage and Plantdoc, and it showed improved accuracy, precision, F1-score, sensitivity, and specificity than the current approaches. This research presents the capability of adaptive deep neural networks in accurate and efficient classification of plant diseases, with solid implications for effective deployment in agriculture monitoring systems [14].

Shafik et al. propose two new models for plant disease detection and classification, PDDNet-AE and PDDNet-LVE, using transfer learning and pre-trained convolutional neural networks (CNNs) to improve the efficiency of classification for sustainable agriculture [15].The models employed ensemble approaches, with PDDNet-AE employing early fusion and PDDNet-LVE employing lead voting ensemble strategies, consolidating the output of nine pre-trained CNNs including DenseNet201, ResNet101, and EfficientNetB7.The models showed excellent performance, with PDDNet-AE reaching 96.74% accuracy and PDDNet-LVE reaching 97.79% on the PlantVillage dataset. The use of a logistic regression classifier fine-tuned for the CNN features enhanced classification accuracy and addressed overfitting issues. The models are computationally cheap and can be easily deployed on small devices, thereby enabling real-world deployment in agricultural settings [15].

The research adds to precision agriculture by offering a strong and effective deep learning-based solution for early and accurate plant disease detection, enabling timely intervention and management.

**2.2 Summary of the Survey and Findings**

## Plant disease detection systems based on AI are developing at a fast pace through the combination of conventional deep learning models, lightweight architectures, and multimodal approaches. Conventional CNN models such as VGG, ResNet, and EfficientNet are basic tools for visual diagnosis. Current research emphasizes model efficiency with high accuracy and low computational complexity. Lightweight CNN models with optimized layers show near state-of-the-art performance with low memory usage, making them deployable in low-resource agricultural settings.

## Deep learning models are improved by ensemble techniques, attention, and image preprocessing like removing background and denoising.Plant disease detection systems based on AI are developing at a fast pace through the combination of conventional deep learning models, lightweight architectures, and multimodal approaches. Conventional CNN models such as VGG, ResNet, and EfficientNet are basic tools for visual diagnosis. Current research emphasizes model efficiency with high accuracy and low computational complexity. Lightweight CNN models with optimized layers show near state-of-the-art performance with low memory usage, making them deployable in low-resource agricultural settings.

## Deep learning models are improved by ensemble techniques, attention, and image preprocessing like removing background and denoising. Model fusion such as AlexNet, ResNet, and VGG using fusion networks enhances performance on overlapping disease feature datasets. Transfer learning significantly contributes to improved performance using pre-trained CNNs, with voting-based and fusion-based ensembling methods enhancing generalization.

## Multimodal techniques employ language models and vision-language embedding methods for enriching disease diagnosis, especially in few-shot and zero-shot scenarios. The pairing of semantic prompts and textual explanations with image data also enables the enhancement of anomaly detection strengths and robust classification. Transformer-based new models surpass CNNs by using attention mechanisms to extract high-level representations and enable explainability with Grad-CAM visualizations.

## Smart systems such as LLM-driven virtual assistants and retrieval-augmented generation enhance decision-making and accessibility in irrigation and crop management. Knowledge graphs and graph attention mechanisms are utilized to capture complicated disease inter-relations.Other non-optical techniques, including Raman spectroscopy, provide real-time and non-destructive disease detection, connecting laboratory expertise with field practice.

## Overall, these AI-driven innovations in plant disease detection are more scalable, explainable, and accurate but are still hampered by data quality, model generalization, and zero-shot disease detection. Future solutions will have to integrate multimodal data, domain knowledge, and optimized architecture for long-term sustainable applications in agriculture.

**2.3 Motivation and Challenges**

The application of artificial intelligence for plant disease detection can revolutionize contemporary agriculture. However, there are several key challenges that continue to hinder its effective use and practical implementation in actual scenarios.

* **Early Disease Detection in Plants for Crop Safety:** Early detection of disease in crops is important in order to avert bulk loss of yield, conserve farmer earnings, and sustain food security. Plant diseases, if left undetected in their initial stages, can quickly spread throughout crops, leading to irreversible damage. With the increasing world population and demand for agricultural produce, there is an absolute need for AI-based systems capable of detecting infections in their initial stages so that timely intervention and containment can be undertaken.
* **Limitations and Imbalances in Data:** The majority of plant disease datasets available in the public domain are plagued by significant shortcomings in terms of quality and quantity. Such datasets tend to be small, biased, and heterogeneous in important variables like crop type, illumination conditions, leaf direction, and background texture.In addition, class imbalance such that certain classes of diseases are overrepresented relative to others, degrades model performance.Such challenging data make AI models biased during training and restricted in their ability to generalize to actual agricultural environments.
* **Generalization Across Environments:** AI models trained on limited or niche data do not generalize when deployed in new geographies, different climatic regions, or on different crops outside the training set. Symptoms of disease change with the environment, and models that do not see this variation while being trained do not classify or detect diseases.This generalizability limitation brings an end to the scalability of AI solutions across various agricultural ecosystems.
* **Computational Limitations in Rural Deployment:** While deep models have been highly accurate in simulated lab settings, they are difficult to implement in farms, especially rural or resource-constrained settings. Farms have limited computational infrastructure, stable internet connectivity, or the ability to obtain high-end mobile devices for running sophisticated models. Therefore, there is an urgent need for light models that consume less energy and can operate offline and be used in low-cost devices without compromising precision.

**2.4 Objectives of the work**

## Accurate Plant Disease Identification: The initial goal is to create a deep learning-based system that can classify plant diseases with high precision from images. Using the sophisticated architectures such as Convolutional Neural Networks (CNNs), the system will provide accurate and reliable classification of different crops. This facilitates early detection and quick response, hence reducing loss of yield and disease spread.

## 

## Multi-Modal Insights Generation: The project entails multi-modal data processing by the integration of visual input and natural language processing. The system uses models like CLIP and LoRA-tuned FLAN-T5 to understand images and text. The system is thus able to not just identify diseases but also provide informative textual reports, describe potential causes, and recommend the correct treatment, thereby providing a holistic diagnostic experience.

## 

## User-Friendly Query Handling: To increase user engagement, an interactive chatbot interface is integrated in the system. This chatbot handles user queries in a natural way, in the form of conversation, providing suitable insights and suggestions. It's a digital assistant that can clear misconceptions, suggest actions, and enhance the ease of use of disease-related information, particularly for non-technical individuals such as farmers.

## 

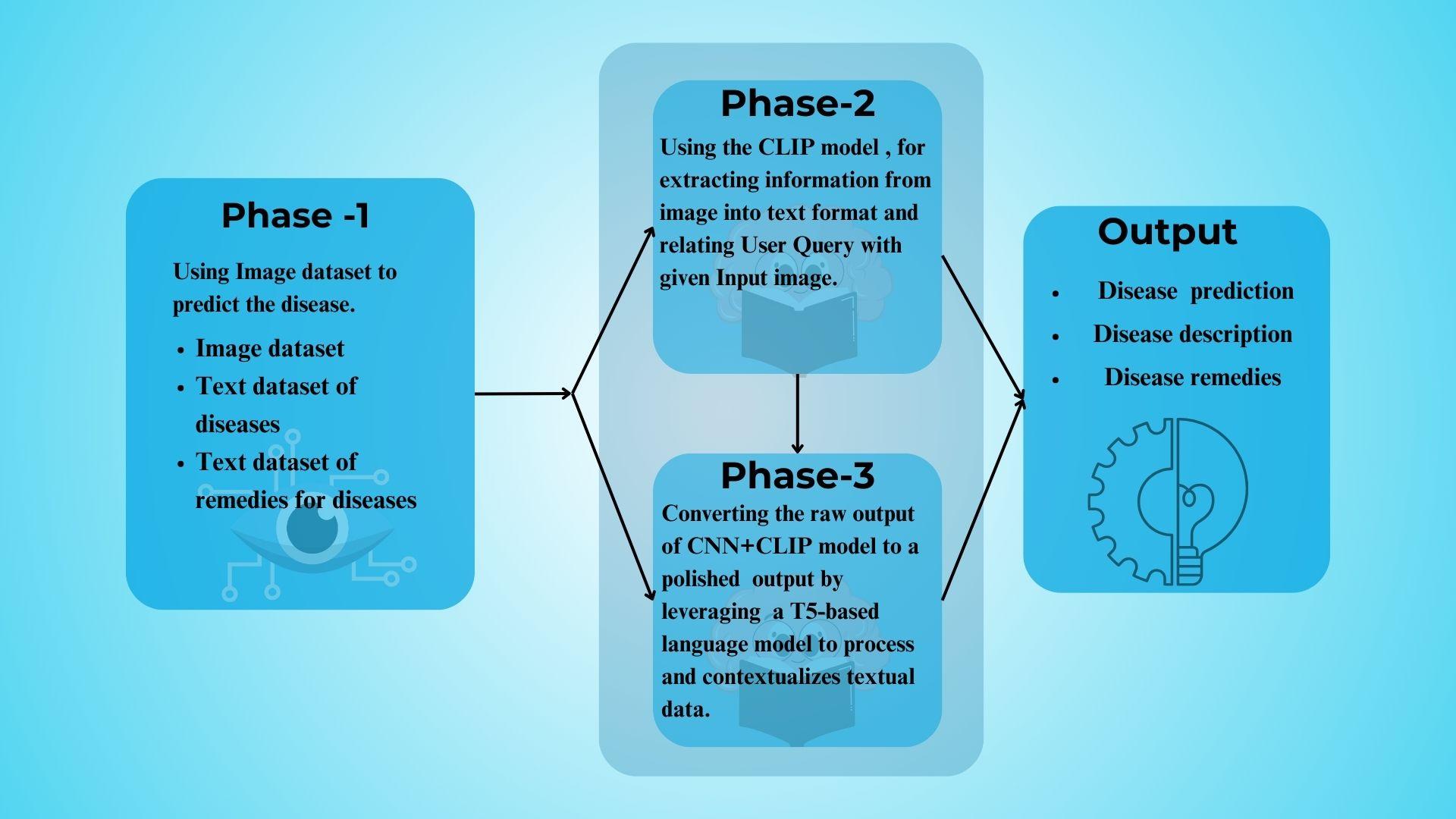
## Enhancing Agricultural Outcomes: One of the main objectives is to close the knowledge gap between farmers and agricultural experts. The system delivers localized, low-cost solutions that enable farmers to make better-informed decisions and empower them with effective plant health management tools. This helps achieve better crop yields, efficient agriculture, and food safety overall.

## Scalability and Real-World Deployment: The system is designed for scalability, and thus it can be deployed even in a resource-poor rural setting. Effective back-end integration, mobile-friendly interface support, and the use of light models make the solution deployable in actual agricultural settings. This brings the technology within reach of those who most need it, rendering digital farming solutions more effective and inclusive.

**Chapter 3**

**PROPOSED WORK**

**3.1 Architecture of the System**

****

*Fig.1.Architecture diagram*

System Architecture for plant disease detection and explanation consists of three core stages, culminating in a revealing output to facilitate farmers and users in the management of diseases.

**3.1.1 Phase 1: Disease Prediction Based on an Image**

Input: An image of diseased or healthy leaf of a plant.

Process:

* A CNN-model examines the image to identify which category of diseases it belongs to.
* Works upon an image data set of leafy plants (e.g., PlantVillage, PlantDoc).
* Also includes text data sets with disease names and cures.

Objective: Accurately predict the plant disease from the visual symptoms.

**3.1.2 Phase 2: Multi-Modal Query Alignment (CLIP)**

Input: Same image from Phase 1 and user question.

Process:

* The CLIP model is employed to align the image features into text representations.
* Translates visual data into a text-compatible embedding space.
* Compares user queries (e.g., "Why are the leaves yellow?") with adjacent disease information from the dataset.

Objective: Understand and convert the user's query to image context.

**3.1.3 Phase 3: Contextual Text Generation (LoRA-FLAN-T5)**

Input: CNN + CLIP intermediates.

Process:

* A LoRA-tuned FLAN-T5 model handles the raw outputs.
* It generates contextually appropriate natural language outputs.
* Ensures that explanation contains disease description, cause, and cure.

Goal: Generate well-formed, meaningful natural language answers.

**3.1.4 Final Output**

* Disease Prediction: The actual disease that is occurring in the plant.
* Disease Description: Brief informative information regarding the disease.
* Disease Remedies: Suggested solutions or treatments to cure/avoid the disease.

This flow diagram illustrates the seamless integration of image classification (CNN), multimodal matching (CLIP), and natural language generation (T5) into a complete AI-based plant disease diagnosis assistant.

**3.2 Algorithm Design**

**3.2.1 CLIP**

Contrastive Language-Image Pre-training (CLIP) is a state of-the-art model designed for learning visual and textual representations inside a shared embedding space. By associating images to textual descriptions, CLIP can generalize well to lots of tasks, including detecting plant diseases. Its ability to use multimodal capabilities led to applying the following methodology to categorize plant diseases and elicit relevant insights. It began by preprocessing the dataset. The plant disease images directory was used along with assigning labels systematically to those images. A custom script was written to produce a labels.json file where the first 30 images were mapped to three predefined classes: apple black rot, apple rust, and apple scab. This ensures the dataset was structured and ordered in such a way as to test the model’s accuracy. The next step involved creating a JSON file containing disease-specific details, including textual prompts for CLIP. Each disease was described using multiple prompts tailored to its symptoms, appearance, and common terminologies. This file served as the textual input for CLIP during inference. Its imaging-related ability in classifying pictures depended on encoding the input picture along with disease-specific text into a shared embedding space. In preprocessing, it transformed the image into the inputs passed to CLIP’s ViT and processed text in tokenized form in the text encoder of CLIP; thereafter, it used it for finding the most accurate similarity score between image feature vectors and textual embeddings between different diseases. An accuracy test was conducted on the performance of the model using the prepared dataset and the labels.json file. The disease classifier was used to classify every test image, and predictions were compared against the ground truth labels. The result gave a quantitative measure of the model’s ability to detect diseases. The system responded to the user’s query in addition to the classification by retrieving relevant information from the JSON file for a disease. For example, when a user typed ”What causes this disease?”, the query system was used to map this user input into a predefined category. The categories used are symptoms, causes, treatments, and prevention methods. The methodology showed the flexibility of CLIP, allowing it to classify plant diseases with minimal domain-specific training and answer disease-related queries. This approach combined high-quality images, descriptive text prompts, and CLIP’s powerful multimodal capabilities in a robust framework for the detection of plant diseases and information retrieval. Traditional machine learning models SVM, ANN, KNN, Fuzzy Classifiers, and CNN, have been applied and exposed with great accuracy in the classifications of plant disease but are always constrained to narrow scopes relative to fewer crops or just a few diseases. For example, CNN managed a 99.53% accuracy but was limited only to a few types of plants; whereas, the ELM model was put under 84.94% accuracy within the PlantVillage dataset, limited only to tomato plants. Although effective for specific tasks, these models lack scalability and generalizability. In contrast, the CNN+CLIP model addresses these limitations by combining CNN’s visual feature extraction with CLIP’s semantic understanding, enabling broader applicability across various plant species and diseases. It is more scalable and generalizable, with an accuracy of about 87%, giving not only the accurate disease classification but also elaborate remedies and descriptions, which would make it a robust and practical solution for real-world agricultural challenges.

| Input Image |  |
| --- | --- |
| Input text | What are its symptoms? |
| Output | Detected Disease: apple scab  The symptoms include: Olive-green to dark brown spots on leaves, Velvety or scabby patches on fruits, Deformed or cracked fruits, Premature leaf drop, Dark, scab-like lesions. |

*Table 1. Input and Output of CLIP Model.*

**3.2.2 BLIP**

BLIP is a bootstrapped language-image pre-training model with vision-language tasks such as captioning, visual question answering, and multimodal retrieval. Although BLIP itself doesn’t explicitly design for the purpose of plant disease detection, it can be applicable for this domain, especially through the multimodal approach. BLIP can match plant images with descriptive texts to help in disease detection as follows: Symptom Description produces textual descriptions about visual symptoms, such as ”yellow spots with browning edges.” Disease Mapping checks for certain image patterns might be associated with certain disease terms, thus helping in diagnosis. Multimodal Fusion allows for the fusion of image data with other relevant text inputs to enhance the effectiveness of detection. Environmental Context, data such as temperature and humidity can be auxiliary inputs to improve robustness. Zero-Shot and Few-Shot Learning is the pre-training of BLIP enables it to generalize well to new tasks with very little labeled data, addressing the scarcity of annotated plant disease datasets.Explainability of model deals with caption generation, for example, ”leaf shows signs of fungal infection”, BLIP allows transparency into its predictions, which in turn enhances usability. The main issues using BLIP for plant disease detection are: Domain-Specific Training includes Fine-tune on datasets such as PlantDoc since BLIP is pre-trained on general datasetsSpecialized Text input to Train on domain-specific descriptions. Image Quality: The resolution may need to be at a high resolution or hyperspectral for subtle symptoms.

| Input Image |  |
| --- | --- |
| Input text | What are these yellow spots on this leaf? |
| Processing | BLIP encodes and cross-references features from both modalities. |
| Output | Disease Classification: Bacterial Leaf Blight.  Generated Caption: leaf shows signs of fungal infection. |

*Table 2. Input And Output Of The Blip Mode*l

**3.2.3 CNN+CLIP**

While CLIP is powerful in learning visual-textual relationships from large-scale general data, it does not suit plant disease detection. The plant diseases consist of minute visual details like rust spots, discoloration, and lesions that are underrepresented in the CLIP’s pre training data, hence less effective. Besides, CLIP’s dependency on textual descriptions is problematic as the symptoms of plant diseases are nuanced, and the descriptions may be inconsistent or ambiguous. More challenges with visually similar diseases: zero-shot learning of Apple Rust Leaf and Apple Leaf Spot. To overcome these limitations, we propose the integration of CLIP with a CNN-based model to leverage both architectures. The CNN is good at extracting fine-grained visual features, such as color changes and spots, which are very important for plant disease detection, while CLIP provides multimodal reasoning by aligning image features with textual descriptions.This synergy enables valid predictions of CNN against disease-specific descriptions for a better accuracy, robustness and interpretability. For instance, assuming the CNN predicts ”Apple Rust Leaf,” CLIP ensures the correct text-alignment in response to symptoms such as textural yellow-orange spots or early fall of leaves in the combined approach with added confidence in diagnosis and provides an effective solution for precision agriculture and automatic monitoring of plant health.

**Methodology**

This methodology explains step-by-step how CNN and CLIP can be integrated to improve plant disease detection. This framework combines the strength of feature extraction in CNN with semantic vision-language alignment in CLIP in order to ensure accuracy in results while also providing explainability.

**Data Preparation**

The framework is based on the PlantDoc dataset, which consists of annotated images of diseased plant leaves. The dataset is organized correctly into subdirectories in which one folder represents a specific plant disease, for instance, ”Apple Rust Leaf” or ”Bell Pepper Leaf Spot”. With the JSON configuration file stored in details about diseases themselves, like symptoms and treatment recommended, this dataset is to be split into training sets and validation sets so model training and evaluation could then be ensured on unseen data.

**Image Pre-processing**

The images are resized to ensure uniform input dimension and optimize feature extraction. In this case, each of the images would be of a uniform 224 x 224 size in pixels, and normalization aligns them with the distributions of pre-trained models input. It is being implemented inside a transformation pipeline using PyTorch with resized, normalized, and tensored image conversion. That is basically preparing images ready for successful CNN and CLIP model-based training as well as inference.

**CNN-Based Visual Feature Extraction**

Plant disease classification uses a pre-trained ResNet-18, fine-tuned on ImageNet. The model architecture is changed by replacing the last fully connected layer with one that has a number of units that will correspond to the categories of plant diseases. It then trains the network on the annotated labels using cross-entropy loss and the Adam optimizer on the PlantDoc dataset. The CNN specializes in extracting fine grained visual features that could be lesions, discoloration, or spots to help detect subtle signs of the disease. From the output of the CNN, it returns the type of disease and fetches corresponding symptoms and treatment information from the knowledge base. The network is trained on the PlantDoc dataset with annotated labels using cross-entropy loss and the Adam optimizer. The CNN specializes in extracting fine grained visual features that could be lesions, discoloration, or spots to help detect subtle signs of the disease. From the output of the CNN, it returns the type of disease and fetches corresponding symptoms and treatment information from the knowledge base.

**CLIP for Vision-Language Alignment**

To ensure the correctness of CNN prediction, CLIP is integrated with the architecture such that image and textual embeddings are matched with one another. Images are fed to ViT backbone of CLIP in order to generate high semantic embedding capturing the global visual context while describing the disease as ”Yellow spots near the center of the leaf” through CLIP language model for generating textual embeddings. This aligns the visual and textual features for the robust semantic validation of the predictions.

**Integration mechanism**

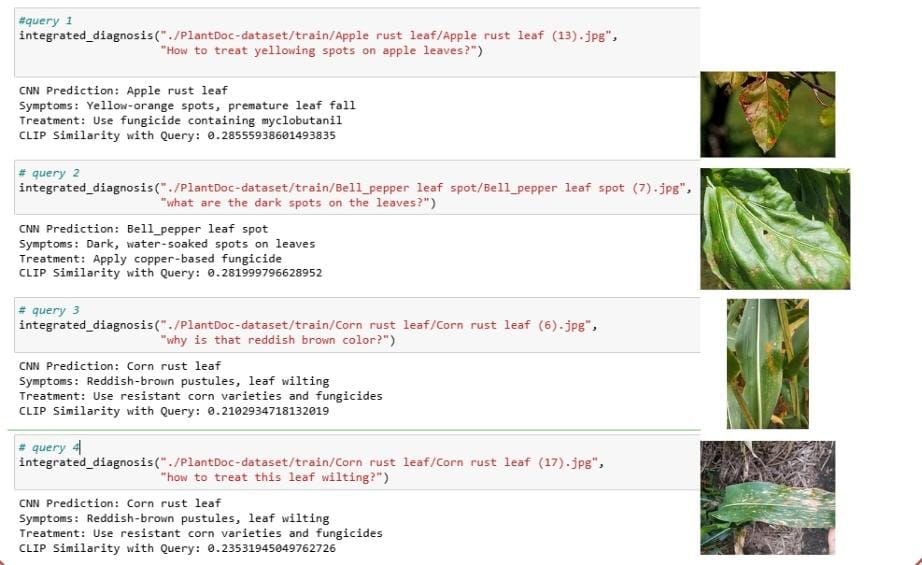
Outputs from CNN and CLIP combined together enhance the entire diagnostics process. The CNN predicts what disease category the patient is suffering from by analyzing the visual features and retrieves information for disease name, symptoms along with other treatment details from the CLIP. CLIP works in conjunction with this by calculating the cosine similarity between the image embedding with the text embedding of a disease description or user query. This similarity score validates the correctness of the alignment of visual and textual representations to provide a semantically verified diagnosis.

**Diagnosis Fusion**

Based on insights from CNN and CLIP, the diagnosis will be holistic and robust. While CNN makes an elaborate classification of disease, it does this with regard to local visual patterns like spots or discoloration, CLIP generalizes and provides semantic validation ensuring that the predictions fit in the descriptions of text. This diagnosis will be comprehensive in output, showing the disease name, symptoms, recommendations for treatment, and a similarity score which reflects how confident one is in making a prediction.G. Advantages of Integration There are numerous merits of integration of CNN and CLIP. The methodology improves interoperability since CLIP verifies CNN predictions against textual description of diseases. The integrative method is more accurate because CNN captures the local features, whereas CLIP captures the global semantics; hence, this reduces the chances of misclassifications. Moreover, it will be adaptive to all conditions that CLIP encounters during pre-training on multi-datasets, complementing the specificity of CNN towards the PlantDoc dataset.

**Output**

The final output of the framework is an interpretable detailed diagnosis of the plant disease. The name of the disease is predicted using CNN and validated by CLIP, along with its symptoms retrieved from the knowledge base and specific treatment recommendations. In addition, the framework supplies a CLIP similarity score, which acts as the confidence measure for the alignment of visual and textual information, ensuring reliable and accurate results.

*Fig 2.CNN+CLIP Plant based diagnosis*

**3.3 Data sets for the study and platform**

**3.3.1 DataSets**

**Plant-Doc Dataset**

### Plant-Doc and New Plant Disease dataset consists of about 87,000 RGB images of leafy plants belonging to 38 plant classes with different types of diseases and healthy conditions. Plant-Doc and New Plant Disease contains images of fruits like tomato, potato, apple, grape, and corn exhibiting fungal, bacterial, and viral diseases.

### Split the data into supervised learning and organize it into:

### Training Set: 80% (~69,600 images)

### Validation Set: 20% (~17,400 images)

### Test Set: A folder of 33 test images to be tested manually.

### All the images are annotated with their respective plant species and disease conditions, and therefore the dataset can be used to train deep learning models for plant disease classification and detection.

*Fig. 3. A sample from the dataset , left one is Corn Leaf Blight, in the center is Tomato Mold leaf and on right is Apple Rust leaf.*

**Dataset Annotation (JSON File for CLIP)**

To facilitate integration of the Plant-Doc dataset with the CLIP model, a JSON file was created in a structured manner. The file is employed as a metadata annotation, mapping the filename of the image to the corresponding disease information. The annotation includes disease name, symptoms, and treatment, enabling the multimodal AI system to learn informative image-text representations.

Each entry in the JSON file contains the following fields:

Filename: the unique identifier for each image, utilized in the directory file name.

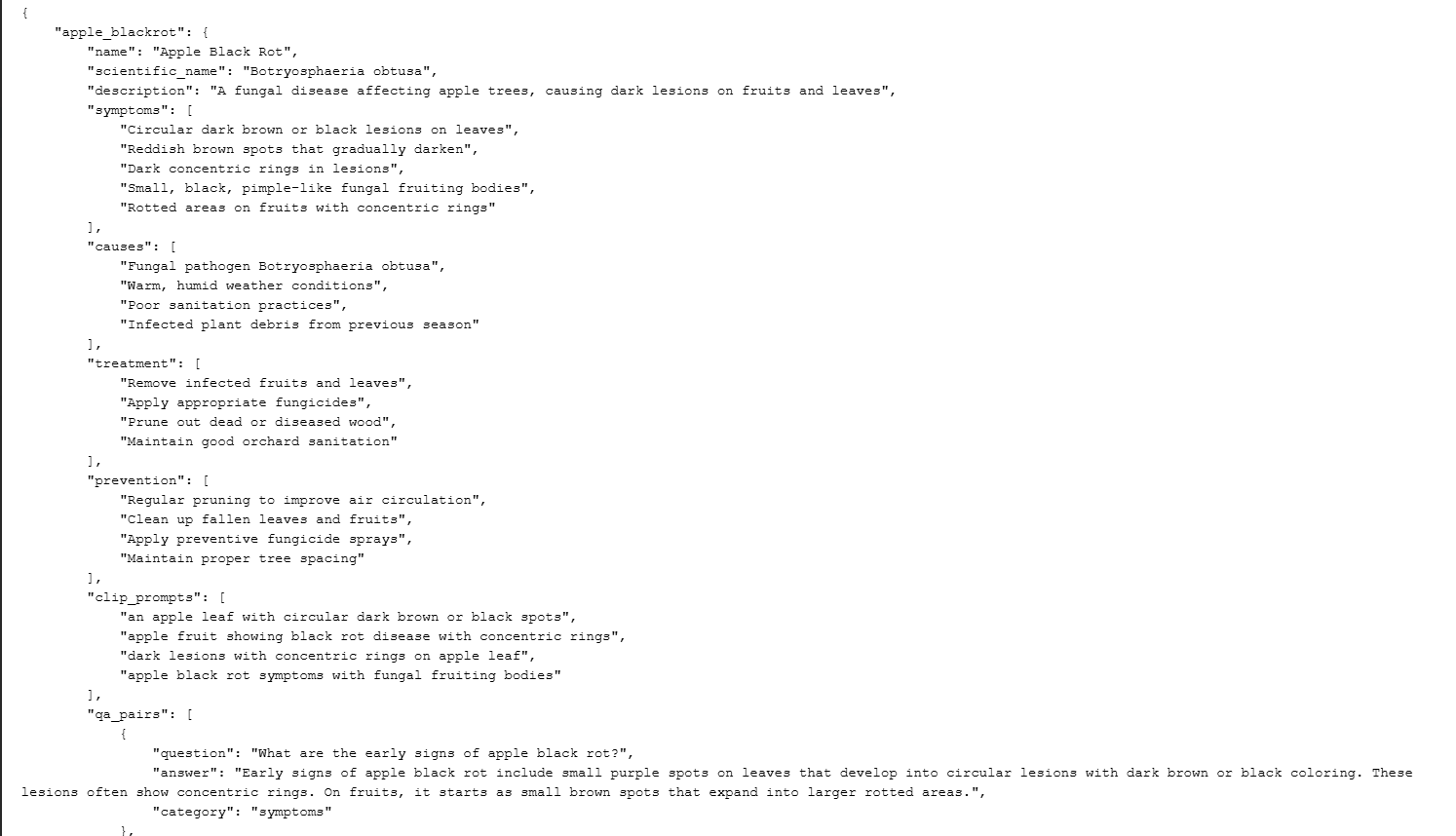
Disease: The name of the disease the plant is suffering from.

Symptoms: A documented record of apparent disease symptoms on leaves, fruits, and branches.

Treatment: Suggesting control measures like fungicides, pruning techniques, and measures of prevention.

Example:

{ "filename": "Apple\_\_\_Apple\_scab", "disease": "Apple Scab", "symptoms": "Round, olive-green to dark brown spots on leaves, fruit, and young twigs. Leaves become deformed, curled, or prematurely defoliated. Infected fruit has rough, scaly patches and cracks. Severe infections can cause defoliation, reducing fruit yield and quality.", "treatment": "Spray fungicides such as captan, mancozeb, or myclobutanil at the start of the season, especially before rainy weather. Grow resistant apple varieties 'Enterprise', 'Liberty', and 'Freedom'. Remove infected fruit and fallen leaves to prevent overwintering of the fungus. Prune trees to improve air circulation and reduce moisture accumulation. Avoid overhead watering and have good spacing between trees to promote air movement." }

*Fig.4.JSON file sample text for CLIP model*

### Chatbot Text Dataset

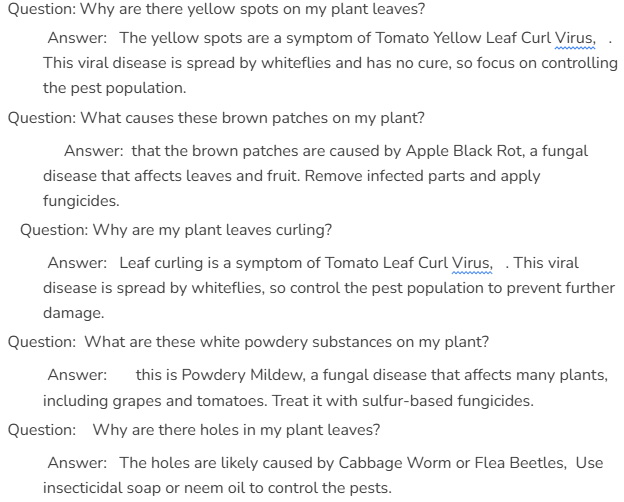
For fine-tuning the chatbot component of our multimodal AI system, we use the Instruction-Tuning-with-GPT-4-RedPajama-Chat dataset, which contains 52,000 high-quality dialogue samples formatted as structured <human>: (user query) and <bot>: (model response) pairs. Each sample consists of:

* Instruction: A user query in natural language.
* Response: A well-structured and informative reply.

This dataset is specifically designed to enhance instruction-following and dialogue-based tasks, making it well-suited for fine-tuning RedPajama-INCITE-Chat-3B-v1 and similar models. The structured nature of these conversations ensures clarity, coherence, and contextual relevance in chatbot responses.

Additionally, we manually curated and added 264 custom QA pairs to the dataset. These additional entries focus on plant disease-specific queries, ensuring that the chatbot is fine-tuned to provide accurate, domain-specific responses relevant to agricultural disease diagnosis.

With this combined dataset, the chatbot is trained to understand and generate meaningful responses related to plant diseases, improving user interaction and engagement.



*Fig .5 . Custom dataset for ChatBot*

**3.3.2 Platform and Specification**

To develop the multimodal generative AI system for plant disease detection, the following software and tools were used:

1. Programming Language

* Python 3.x – Used f
* or model development, training, and integration.

2. Deep Learning Frameworks

* PyTorch – For implementing the CNN and CLIP models.
* Hugging Face Transformers – For integrating the chatbot model (lora-flan-t5-large-chat).

3. Dataset & Preprocessing

* Plant-Doc Dataset – Used for training and testing the plant disease detection model.
* OpenCV & PIL (Python Imaging Library) – For image preprocessing (resizing, normalization, augmentation).

4. Model Architectures

* ResNet-18 – Used as the backbone for CNN-based image feature extraction.
* CLIP – Used for text-based disease classification and multimodal learning.

5. Development & Training Environment

* Google Colab and Jupyter Notebook – Used for training and testing the models.
* CUDA-enabled GPU – Required for efficient deep learning model training.

6. Model Storage

* Google Drive – Storing trained models (plant\_disease\_cnn\_model.pth, lora-flan-t5-large-chat).

**Chapter 4**

**RESULTS AND INFERENCES**

**4.1 Metrics for evaluation**

In measuring the performance of the CNN, ResNet18, VGG16, and EfficientNet models for classifying plant diseases, some key metrics were applied in the assessment of the performance and trustworthiness of the system. Accuracy in terms of the proportion of correct predictions made over the total number of predictions was employed in giving a general measure of model performance, particularly on datasets such as PlantVillage and PlantDoc where class distribution was fairly balanced. Accuracy, the ratio of correctly classified diseased samples out of all the predicted diseased cases, was important in order to prevent false positives that may cause unnecessary concern or abuse of treatment chemicals by farmers. In contrast, Recall—the ratio of correctly classified diseased leaves to all actual diseased samples—was important because failure to detect a disease (false negatives) may result in uncontrolled propagation and heavy crop loss.

The F1-Score, being the harmonic mean of recall and precision, provided a balanced estimation, particularly where datasets may be class imbalanced or have variable disease severity levels among samples. This was needed to capture the extent to which the model could generalize across a variety of types of disease without overestimating more common classes. Besides, a Confusion Matrix was utilized to gain insights into specific class-level errors—e.g., confusion of leaf rust with early blight—and to learn about areas that should be addressed in terms of dataset enhancement or model retraining.

Performance for the text generation part based on CLIP and LoRA-FLAN-T5 was measured in terms of BLEU (Bilingual Evaluation Understudy) and ROUGE (Recall-Oriented Understudy for Gisting Evaluation) scores.BLEU evaluated the equivalence of word sequences in the chatbot responses and reference responses, useful for evaluating grammatical and contextually relevant correctness.ROUGE, especially ROUGE-L, measured longer sequence overlap to ensure that the generated disease information or suggested treatments captured context and facts needed.

Additionally, Inference Time was tested with both image and text inputs because practical applications must have low latency to provide good user interaction. This was particularly crucial in agricultural deployment situations in rural areas where slower network access and fewer devices might affect usability. These exhaustive metrics guaranteed the system not only to be correct but also robust, understandable, and field-deployable in agriculture.

**4.2 Parameters settings**

The models used for plant disease detection and response generation in this project were configured with carefully selected parameters to optimize performance while maintaining computational efficiency. Each model had its hyperparameters fine-tuned either through grid search, experimentation, or based on best practices from recent literature.

#### 4.2.1 CNN (ResNet-18 Based Image Encoder)

* **Input Size**: (224, 224, 3)
* **Data Augmentation**: RandomHorizontalFlip, RandomRotation (±15°), RandomZoom
* **Normalization**: Rescale pixel values to [0, 1]
* **Architecture**:
  1. Pre-trained ResNet-18 (ImageNet weights)
  2. GlobalAveragePooling2D after last convolution block
  3. Dropout: 0.5 to reduce overfitting
  4. Dense Layer: 256 units with ReLU activation
  5. Output Layer: Softmax (38 classes – for each plant disease type)
* **Loss Function**: Categorical Crossentropy
* **Optimizer**: Adam (Learning Rate = 0.0001)
* **Epochs**: 30
* **Batch Size**: 32

#### 

#### 4.2.2 CLIP (Contrastive Language–Image Pretraining)

* **Vision Encoder**: CLIP ResNet-18
* **Text Encoder**: Transformer-based text encoder (from CLIP)
* **Text Prompts**: "Image of a [disease name] on a [crop name]" for 38 classes
* **Preprocessing**:
  + Image resizing: (224, 224)
  + Tokenization of text using CLIP tokenizer
* **Inference Method**: Zero-shot similarity computation between encoded image and prompt embeddings using cosine similarity
* **Threshold**: Top-1 prediction based on highest similarity score

#### 4.2.3 LoRA-FLAN-T5 (Chatbot Text Generator)

* **Base Model**: FLAN-T5-Large fine-tuned with LoRA
* **Input Format**:
  + Prompt: "What are the symptoms and treatments for [disease]?"
  + Text conditionally generated based on disease classification output
* **Parameters**:
  + Max Input Length: 256 tokens
  + Max Output Length: 200 tokens
  + Temperature: 0.7 (controls randomness of response)
  + Top-k Sampling: 50
  + Top-p (nucleus sampling): 0.9
* **Training**: Fine-tuned using a curated Q&A dataset specific to crop diseases
* **Inference**: Response generated only upon user query, triggered post-disease detection

**4.3 Results and discussion**

Before integrating **CLIP** for multimodal learning, four different **CNN architectures** were trained and evaluated on the **Plant-Doc dataset** to determine the best-performing model for plant disease classification. The selected architectures were:

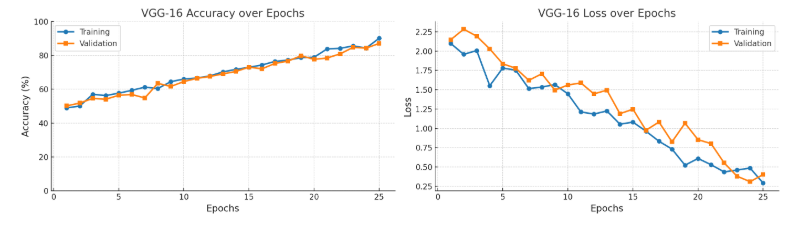
1. VGG-16
2. ResNet-18
3. MobileNet
4. EfficientNet-B0

#### 4.3.1 Experiment 1: Comparison of Four CNN Models

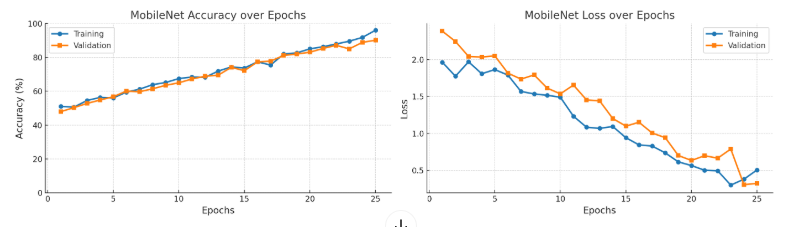
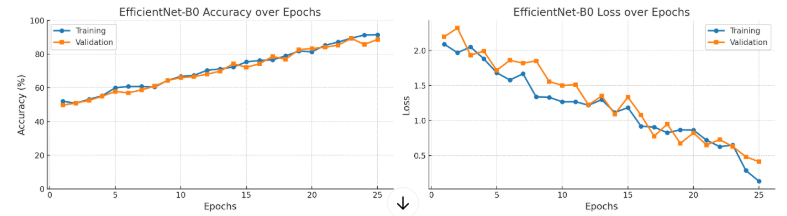
* Evaluating **4 models** on the dataset.
* Identifying the best model

| CNN Architecture | Accuray |
| --- | --- |
| VGG-16 | 86.17 |
| ResNet-18 | 95.30 |
| MobileNet | 92.11 |
| EfficientNet-B0 | 90.51 |

*Table 3 . Comparison of CNN models*



### 



*Fig.6.Graphs for CNN models*

### 

### 

### 

### 4.3.2 Experiment 2: Integrating CLIP with ResNet-18

After having chosen ResNet-18 as the best performing CNN model for plant disease classification, we proceeded to integrate it with CLIP to enable multimodal learning. The focus of the experiment was on preparing the dataset as per CLIP specifications and establishing a classification and query-based system and not performing a thorough evaluation.

Step 1: Preparing the Dataset for CLIP

As CLIP is working on image-text pairs, we had to pre-have a JSON file with a mapping between plant disease images and their corresponding disease names and descriptions. The dataset was in the following format:

* Disease name and scientific name
* Description of the disease
* Symptoms, causes, treatment, and prevention details
* Clip prompts: Carefully curated textual descriptions of how the disease visually appears in images
* QA pairs: Predefined question-answer pairs related to each disease, enabling interactive querying

Using CLIP prompts, we extracted the images and paired them with the JSON file we created. This approach allowed CLIP to learn associations between disease images and textual descriptions effectively.

This approach allowed CLIP to learn associations between disease images and textual descriptions.

#### Integrating CLIP with ResNet-18

* **Image Feature Extraction:** ResNet-18 was used as a feature extractor, encoding plant disease images into high-dimensional vectors.
* **Text Processing:** CLIP’s Transformer processed the disease descriptions from our dataset, generating text embeddings.
* **Similarity Matching:** Instead of direct classification, we used cosine similarity to find the closest match between an input image and a disease description.

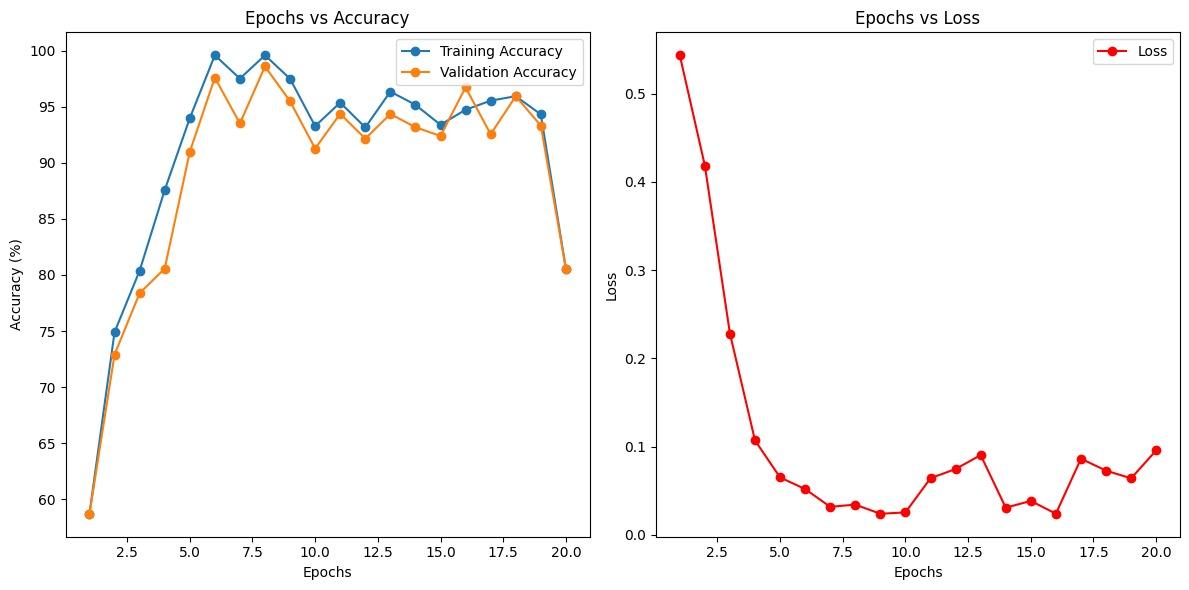
#### 

#### 

#### Classification System Implementation

The system processed an input image by:

1. Preprocessing the image using CLIP’s required format.
2. Encoding the image with ResNet-18.
3. Encoding textual disease descriptions with CLIP’s Transformer.
4. Computing similarity scores between the image and text embeddings.
5. Selecting the highest similarity match as the predicted disease.

****

*Fig.7.Graph for CNN+CLIP model*

#### Step 4: Interactive Query-Based System

In addition to classification, we implemented an interactive question-answering system, allowing users to ask about:

* Symptoms (e.g., “What does apple scab look like?”)
* Causes (e.g., “What causes powdery mildew?”)
* Treatment methods (e.g., “How do you treat black rot in apples?”)

The system searched for keyword-based matches in the dataset and provided relevant information from the disease descriptions.

### 

### 

### 

### 4.3.3 Experiment 3: Evaluating CLIP Model on Our Dataset

After selecting ResNet-18 as the best-performing CNN model for plant disease classification, the next step was to integrate CLIP to enhance multimodal capabilities. The goal of this experiment was to evaluate how well CLIP understands our dataset when combined with ResNet-18 for image encoding.

#### 

#### Dataset Preparation for CLIP

Since CLIP is trained on image-text pairs, we needed to create a custom JSON file mapping plant disease images to their corresponding disease names and descriptions. The JSON file was structured as follows:

{ "filename": "Apple\_\_\_Apple\_scab", "disease": "Apple Scab", "symptoms": "Round, olive-green to dark brown spots on leaves, fruit, and young twigs. Leaves become deformed, curled, or prematurely defoliated. Infected fruit has rough, scaly patches and cracks. Severe infections can cause defoliation, reducing fruit yield and quality.", "treatment": "Spray fungicides such as captan, mancozeb, or myclobutanil at the start of the season, especially before rainy weather. Grow resistant apple varieties 'Enterprise', 'Liberty', and 'Freedom'. Remove infected fruit and fallen leaves to prevent overwintering of the fungus. Prune trees to improve air circulation and reduce moisture accumulation. Avoid overhead watering and have good spacing between trees to promote air movement." }

This structured format allowed CLIP to learn meaningful relationships between images and textual disease descriptions.

Integration of CLIP with ResNet-18

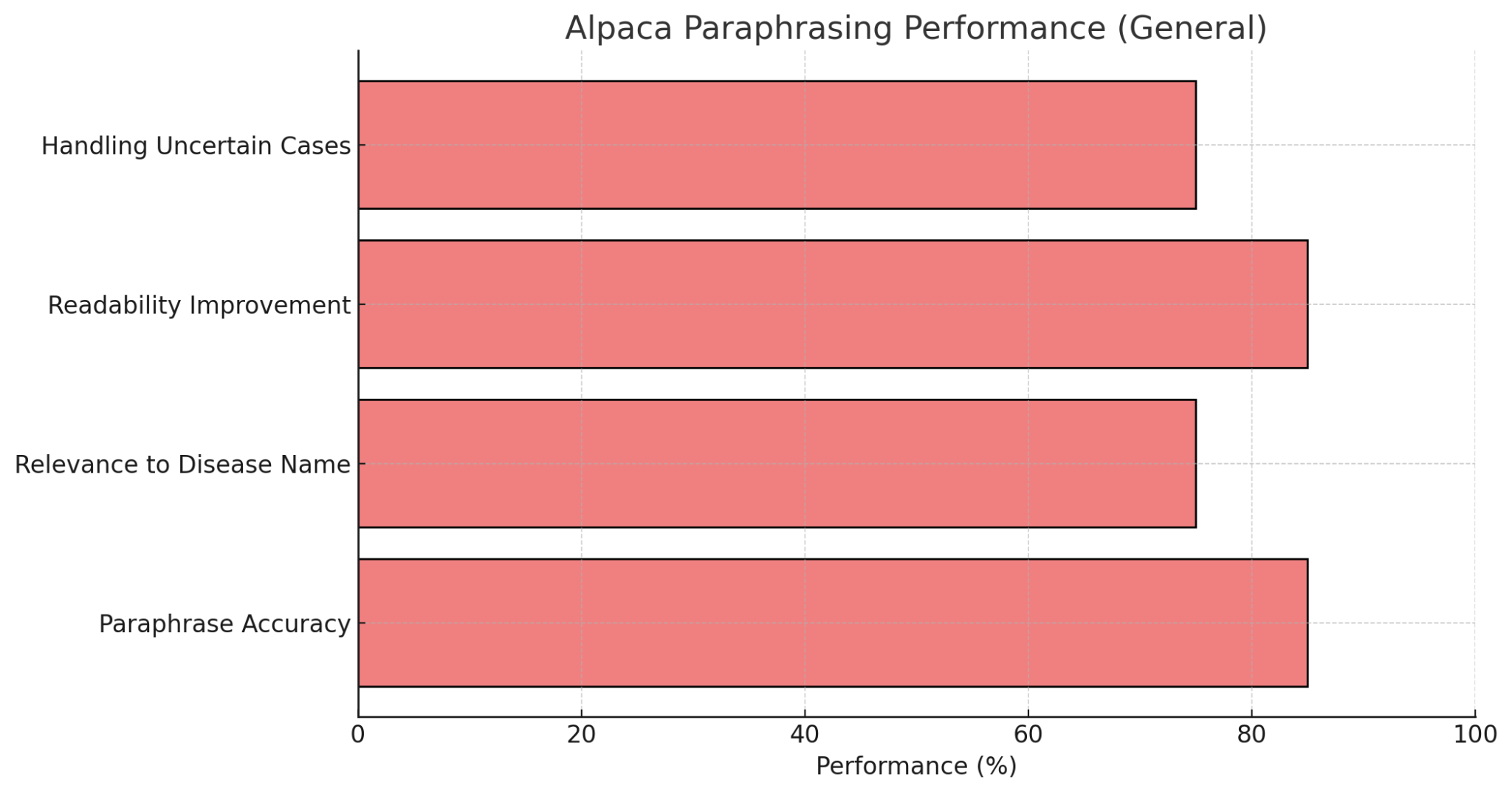
* **Image Encoding:** ResNet-18 was used as the feature extractor to encode images into high-dimensional representations.
* **Text Encoding:** CLIP’s Transformer processed the disease descriptions from our JSON file to generate text embeddings.
* **Similarity Matching:** The system computed the cosine similarity between image and text embeddings to determine the best disease match.

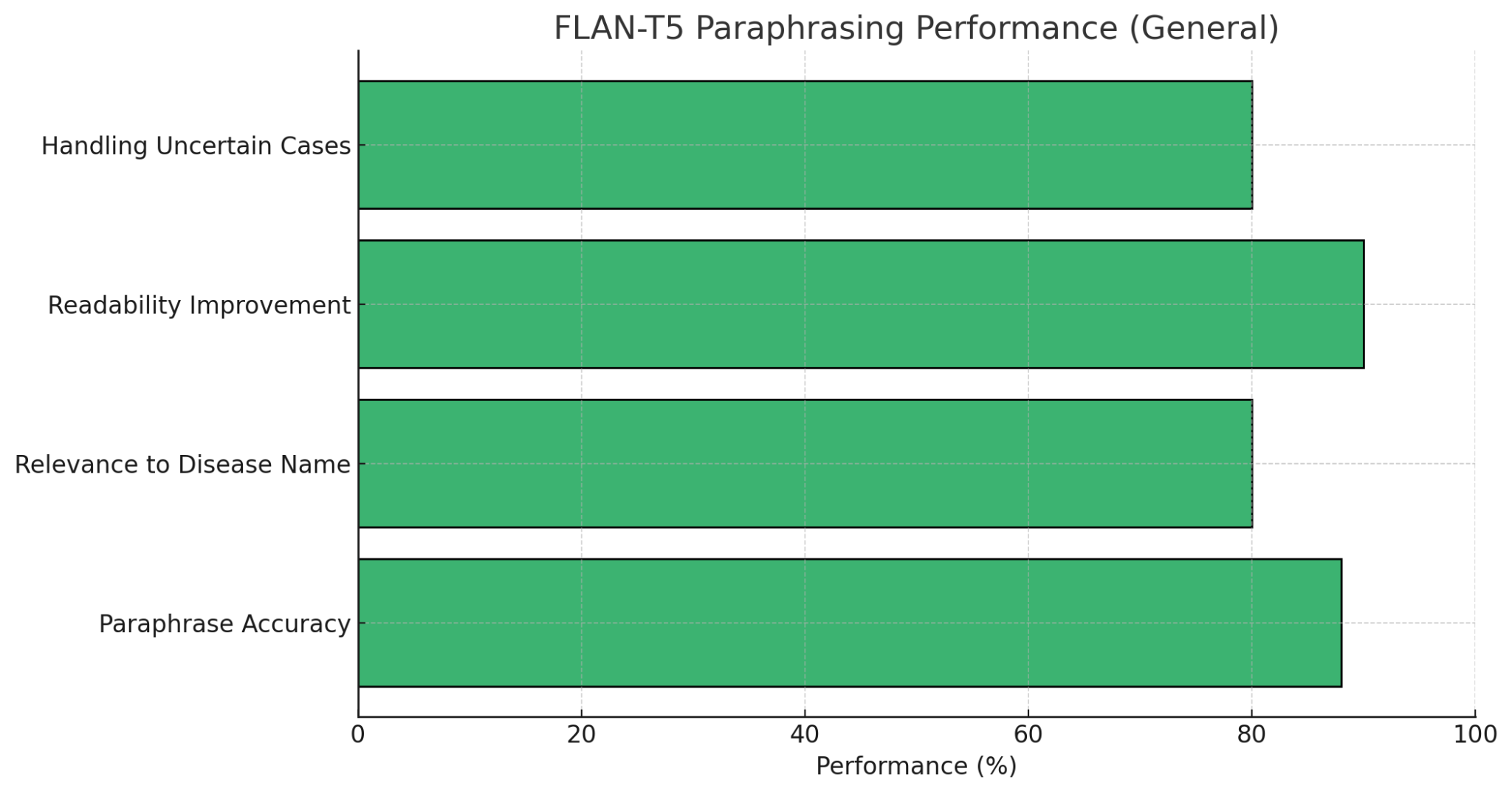
**4.3.4 Experiment 4**: **Models Evaluated : Alpaca and FLAN-T5**

* **Alpaca**: Instruction-tuned LLaMA model (7B parameters).
* **FLAN-T5**: Fine-tuned T5 model optimized for instruction-following tasks.

| **Metric** | **Calculation Methods** | **Alpaca** | **FLAN T5** |
| --- | --- | --- | --- |
| Paraphrase Accuracy | Cosine Similarity | 0.8838076873058497 | 0.9214155218205217 |
| Relevance to Disease Name | Jaccard Similarity | 0.22727272727272727 | 0.17391304347826086 |
| Readability Improvement | Flesch Reading Ease | 82.91544142857143 | 86.92857143893345 |
| Handling Uncertain Cases | Flesch Reading Ease | 14.228571428571428 | 18.893063024571935 |

*Table.4 . Performance comparison of LLM’s used .*

*Fig .8: Alpaca performance*



*Fig.9. FLAN T5 performance*

FLAN-T5 was preferred over Alpaca upon closely comparing their capabilities. FLAN-T5 is specifically tuned for text-to-text tasks, hence more adaptable for structured paraphrasing, but Alpaca is specifically tuned for instruction following and conversational use cases. FLAN-T5 is also superior in multitask learning, which allows for improved generalizability across tasks, and its larger model size (e.g., 11B parameters) allows for improved contextualization compared to Alpaca's 7B model.In addition, FLAN-T5 is pre-trained on a broader and more general dataset, assuring strong performance across diverse environments, and high-quality zero-shot performance reduces the need for extended fine-tuning. On a whole, the task-orientated structure of FLAN-T5, flexibility with multitasks, and straight-out performance as is surpass that of Alpaca

### 4.3.5 Experiment 5: Chatbot Integration using FLAN-T5

In this experiment, we integrated a chatbot component into our multimodal plant disease detection system. The goal was to enable interactive user queries related to the detected disease, allowing farmers and agricultural experts to receive relevant information about symptoms, causes, treatment, and prevention.

The previous experiment, which relied on keyword-based matching from a structured JSON dataset, often failed to generate human-like responses. The system struggled with understanding natural variations in user queries and lacked the flexibility to provide nuanced answers. This highlighted the need for an LLM-based approach to enhance the chatbot's ability to generate more conversational and context-aware responses.

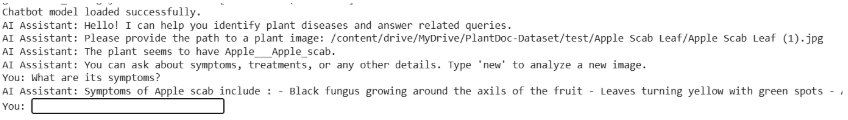
To achieve this, we used Flan-T5 Large with LoRA fine-tuning, which allowed the chatbot to generate contextually accurate and informative responses while remaining computationally efficient. Initially, the chatbot responses were generic and not very accurate, so we improved its performance by modifying the dataset to include manually curated question-answer pairs.

### Chatbot Dataset and Model Integration

For chatbot development, we utilized the chatbot dataset mentioned in the dataset details. This dataset contained disease-specific question-answer pairs, which were manually curated to improve the chatbot’s response quality. Initially, the chatbot’s responses were generic, but after adding manually created questions and answers, it became more accurate and informative in addressing user queries.

Before integrating the chatbot, we ensured that the plant disease detection model (ResNet + CLIP) was saved as a trained model file, plant\_disease\_cnn\_model.pth. Similarly, the chatbot model was fine-tuned and saved as lora-flan-t5-large-chat. These two models were then integrated to work together in the following way:

1. **Image Upload & Disease Detection**
   * The user uploads an image of a plant leaf.
   * The ResNet + CLIP model (plant\_disease\_cnn\_model.pth) predicts the disease.
   * The system returns the result as “Disease detected: [disease name]”.
2. **User Query & Chatbot Response**
   * The user asks a question about the detected disease.
   * The chatbot model (lora-flan-t5-large-chat) processes the query.
   * A meaningful response is generated based on the chatbot dataset.

This integration ensures a seamless experience, where the user gets accurate disease identification and detailed explanations about treatment, causes, and prevention.

**Fig.4.5 :** ChatBot Response for user query.

*Fig.10.Final output of chatbot*

**CHAPTER 5**

**CONCLUSION AND FUTURE ENHANCEMENTS**

**5.1 Summary of the work**

We built a project on designing an AI system for detecting plant diseases by a multimodal method. The system architecture, involving image and text processing through deep learning models. The feature extraction of plant leaf images is achieved using a CNN, whereas CLIP facilitates cross-modal learning between images and text descriptions of the diseases. These outputs are subjected to a tuned FLAN-T5 language model to enable an interactive chatbot to produce wise responses to queries of users on diseases identified. The whole system is optimized to enable natural language interaction and accurate diagnosis.

The methodology, where data preprocessing and augmentation are started first to improve model performance. The dataset utilized is RGB images of healthy and diseased crop leaf images from 38 classes, preprocessed using resizing, normalization, and data augmentation techniques to make them strong and avoid overfitting. The CNN model—typically ResNet or VGG—is trained on the dataset to classify the disease. Concurrently, the CLIP model aligns the image embeddings with their respective textual data to enhance multimodal representation. For chatbot functionality, the researchers prepared disease-specific question-answer pairs and fine-tuned a LoRA-based FLAN-T5 model for best performance. Integration steps included combining the outputs from CNN and CLIP with the chatbot to create an end-to-end pipeline whereby an uploaded image triggers disease detection and user queries trigger proper answers.

By combining individual implementations of CNN,CLIP and fine-tuned a LoRA-based FLAN-T5 model , a full-stack application with the backend on FastAPI and a frontend on React was built. The backend processes image classification, chatbot response generation, and frontend communication. When a user submits an image, the system initially predicts the disease and then enables the user to inquire about it. The chatbot, driven by the FLAN-T5 model, responds in accordance with the context of the identified disease. Care was taken to avoid the repetition of the prediction result unnecessarily and to give meaningful, relevant responses. The frontend interface is user-friendly with image upload functionality, display of disease output, and a chat interface for user interaction.

**5.2 Contributions in the work**

This project presents a novel and smart multimodal deep learning model developed for real-time plant disease detection and diagnosis. One of the key contributions is the synergy of computer vision and natural language processing, which tremendously boosts the quality and usability of agricultural disease detection devices. By integrating Convolutional Neural Networks (CNNs) for disease classification based on images with the CLIP model for aligning visual and text semantics, the system can detect plant diseases from leaf images while also interpreting user queries in natural language. Such integration allows the system to provide meaningful outputs such as disease names, descriptions, and suggested remedies, hence closing the gap between raw model predictions and farmer-friendly insights.

The second innovative contribution of this work is employing a LoRA-optimized fine-tuned FLAN-T5 language model, which enhances the ability of the chatbot to generate contextually sound and human-like responses. Unlike typical disease detection models, which yield merely class labels or probabilities, our framework allows farmers to have conversational interaction with the system through asking follow-up questions or asking explanations about the symptoms, prevention, or treatment of a disease. This interactivity turns the system not only a diagnostic tool but also an educational assistant to farmers. Additionally, the design is lightweight and deployable on resource-constrained environments, and therefore very accessible for application in rural areas. The modularity of the model in its architectural design enables the adaptation of the model in different types of crops and different geographical locations, and therefore it addresses a broad array of agricultural issues.

Besides technology innovation, the project also encourages sustainable agriculture through the provision of timely and accurate disease information to farmers. By facilitating early disease detection and offering localized, actionable information, the solution helps to minimize crop loss, decrease unnecessary pesticide application, and increase overall yield.

The visual explainability provided by saliency maps and the contextual explanations supplied by the language model improve user trust and comprehension of AI results. Such human-oriented design makes the model more acceptable and user-friendly to non-expert users. As a whole, this project presents an intelligent, interactive, and scalable solution that challenges the limits of conventional AI in agriculture, providing a platform for future development of smart agricultural technologies.

**5.3 Future enhancements**

To maintain the system efficient and updated in the long run, numerous future updates are expected. The major goal is to improve the completeness of the dataset by updating the database of the chatbot with information on newly occurring plant diseases, shifting pathogens, and shifting environmental conditions. Additionally, inclusion of live agricultural databases will make the chatbot responsive and accurate. Reinforcement learning can be employed to enhance the performance of the system's responses from user feedback and interaction so that it can adapt and make decisions in untested situations.Furthermore, the integration of external sources of agricultural data will improve the model's ability to offer better and more credible diagnosis and solutions.

Another important aspect of concern is to make the system more usable and user-friendly for users from varying categories. Enhancing natural language processing to support more regional languages will make it easy for farmers belonging to various linguistic groups to interact with the chatbot.A non-technical, easy-to-use interface will also facilitate easy adoption by non-technical users from among farmers.Even more so, to further guarantee maximum coverage and uptake, scalable deployment methodologies will take top priority—orienting towards mobile and cloud-enabled platforms to facilitate the solution for farmers in hard-to-reach and underserved regions, without compromising the ability of the system to handle expanding user loads and diversity of agriculture data.

**Chapter 6**

**REFERENCES**

[1] Khan, M. A., Sharif, M., Akram, T., & Raza, A. (2024). An improved strategy for plant disease detection using deep learning. In 2024 International Conference on Artificial Intelligence (ICAI). IEEE.

[2] Liu, S., Huang, Y., Zhang, W., Lu, J., Zhang, Y., & Wu, Z. (2024). Visual information guided multi-modal model for plant disease anomaly detection. Intelligent Computing in Agriculture, 6.

[3] Abbas, J., Bibi, N., Naqvi, R. A., Sadeghi-Niaraki, A., & Jeong, D. (2024). Revolutionizing agriculture with artificial intelligence: Plant disease detection methods, applications, and their limitations. Frontiers in Plant Science, 15.

[4] Sunil, C. K., & Jaidhar, C. D. (2024). An efficient infectious disease detection in plants using deep learning. In Intelligent Technologies (Vol. 1167, pp. 55–74). Springer.

[5] Lachure, J., & Doriya, R. (2024). Designing of lightweight deep learning framework for plant disease detection. SN Computer Science, 5(6), Article 761.

[6] Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using deep learning for image-based plant disease detection. Frontiers in Plant Science, 7, 1419.

[7] Zhang, J., Huang, Y., Pu, R., Gonzalez-Moreno, P., Yuan, L., Wu, K., & Huang, W. (2019). Monitoring plant diseases and pests through remote sensing technology. Precision Agriculture, 20(4), 678–700.

[8] Fuentes, A., Yoon, S., Kim, S. C., & Park, D. S. (2018). A robust deep learning-based detector for real-time tomato plant diseases and pests recognition. Sensors, 18(2), 373.

[9] Barbedo, J. G. A. (2016). A review on the main challenges in automatic plant disease identification based on visible range images. Biosystems Engineering, 144, 52–60.

[10] Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. Computers and Electronics in Agriculture, 147, 70–90.

[11] Plant disease classification: A comparative evaluation of ResNet, InceptionV3, and EfficientNet models. (2024). PMC.

[12] A comparative analysis of different CNN models for plant disease detection. (2024). International Journal of Advanced Research in Computer and Communication Engineering.

[13] Li, D., Shao, W., & He, S. (2020). Multimodal plant disease identification system based on deep learning. In 2020 IEEE International Conference on Image Processing (ICIP). IEEE.

[14] Xian, T. S., & Ngadiran, R. (2021). Plant diseases classification using machine learning. Journal of Physics: Conference Series, 1962(1), 012024.

[15] Islam, M. S., Sultana, S., Farid, F. A., Islam, M. N., Rashid, M., Bari, B. S., Hashim, N., & Husen, M. N. (2022, August). Multimodal hybrid deep learning approach to detect tomato leaf disease using attention-based dilated convolution feature extractor with logistic regression classification.

[16] Feng, X., Zhao, C., Wang, C., Wu, H., Miao, Y., & Zhang, J. (2022, June 24). A vegetable leaf disease identification model based on image-text cross-modal feature fusion. Frontiers in Plant Science.

**Chapter 7**

**LIST OF PUBLICATIONS**

**Title of the Paper:**

Multi-modal Generative AI for Plant Disease Detection

**Authors:**

Sai Siddharth Gimmedi, Kamsala Deepak Kumar , Chavali Vallabha Makarand,

Yakkala Sai Supraja, Dr. Rajathilagam B

**Conference Title:**

9th International Conference on Information and Communication Technology for Intelligent Systems (ICTIS THAILAND 2025)

**Status of the Publication:**

Accepted and Presented

An extended version of the paper has been drafted and submitted to the project guide for further review and improvement.