LEAF DISEASE PREDICTION

A Minor Project Synopsis

Submitted in Fifth Semester of Bachelor of Technology (B.Tech)

In
Computer Science & Engineering
(Session: 2024-25)

Submitted By

Student Name: Agam Patel
Enroll No: 0205CC221015
Student Name: Aman Agnihotri
Enroll No: 0205CC221019
Student Name: Anushri Tiwari
Enroll No: 0205CC221030

Under the Guidance of

1. **Prof. Brajesh Patel**
   (HOD, CSE)
2. **Prof. Apoorv Khare**
   (Project Guide)

Department of Computer Science & Engineering

SHRI RAM INSTITUTE OF TECHNOLOGY, JABALPUR (M.P.)
RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL (M.P.)

SHRI RAM INSTITUTE OF TECHNOLOGY
Approved by AICTE New Delhi & Govt. of M.P.
(Affiliated to R.G.P.V.V. - University of Technology of Madhya Pradesh)
ISO 9001 : 2000 Certified Institution
Near ITI, MADHOTAL, JABALPUR - 482 002 (M.P.)
Ph. No. : 0761 - 2640291, 94 Fax No. : 2640294 Mobile - 9300104815

CERTIFICATE

This is to certify that

Agam Patel (Enroll No.: 0205CC221015)

Aman Agnihotri (Enroll No.: 0205CC221019)

Anushri Tiwari (Enroll No.: 0205CC221030)

Students of 5th Semester, Computer Science & Engineering, Shri Ram Institute of Technology, Jabalpur have submitted their Minor Project Synopsis entitled **"LEAF DISEASE PREDICTION"** for the completion of 5th Semester examination under the requirement for the degree of Bachelor of Technology as per R.G.P.V., Bhopal.

Prof. Apoorv Khare
(Project Guide)
Computer Science & Engineering
SRIT, Jabalpur, M.P.

Prof. Brajesh Patel
(HOD, CSE)
Computer Science & Engineering
SRIT, Jabalpur, M.P.

SHRI RAM INSTITUTE OF TECHNOLOGY
Approved by AICTE New Delhi & Govt. of M.P.
(Affiliated to R.G.P.V.V. - University of Technology of Madhya Pradesh)
ISO 9001 : 2000 Certified Institution
Near ITI, MADHOTAL, JABALPUR - 482 002 (M.P.)
Ph. No. : 0761 - 2640291, 94 Fax No. : 2640294 Mobile - 9300104815

CERTIFICATE

This is to certify that

Agam Patel (Enroll No.: 0205CC221015)

Aman Agnihotri (Enroll No.: 0205CC221019)

Anushri Tiwari (Enroll No.: 0205CC221030)

Students of 5th Semester, Computer Science & Engineering, Shri Ram Institute of Technology, Jabalpur have submitted their Minor Project Synopsis entitled **"LEAF DISEASE PREDICTION"** for the completion of 5th Semester examination under the requirement for the degree of Bachelor of Technology as per R.G.P.V., Bhopal.

(INTERNAL EXAMINER) **(EXTERNAL EXAMINER)**

SHRI RAM INSTITUTE OF TECHNOLOGY
Approved by AICTE New Delhi & Govt. of M.P.
(Affiliated to R.G.P.V.V. - University of Technology of Madhya Pradesh)
ISO 9001 : 2000 Certified Institution
Near ITI, MADHOTAL, JABALPUR - 482 002 (M.P.)
Ph. No. : 0761 - 2640291, 94 Fax No. : 2640294 Mobile - 9300104815

ACKNOWLEDGEMENT

I would like to express my deep sense of gratitude and sincere thanks to my guide **Prof. Apoorv Khare**, Department of Computer Science & Engineering at Shri Ram Institute of Technology, Jabalpur for his valuable and ever willing precious guidance, technical support and constant encouragement during the course of this project work. It was pleasure and unique experience for me to work under their guidance.

I am grateful to Prof. Brajesh Patel, Head of Department of Computer Science & Engineering and other staff members of Computer Science and Engineering Department for providing the necessary facilities for the successful completion of this work.

Finally, my greatest thanks to my family for their patience and understanding.

Agam Patel (Enroll No.: 0205CC221015)

Aman Agnihotri (Enroll No.: 0205CC221019)

Anushri Tiwari (Enroll No.: 0205CC221030)

PREFACE

This project report on "Leaf Disease Prediction" focuses on using advanced deep learning techniques to detect and classify plant leaf diseases. By analyzing images from a Kaggle dataset, the project aims to assist in early disease identification, improving agricultural productivity. Models like MobileNet, DenseNet, ResNet, and Inception were utilized after preprocessing the images for optimal results. This report highlights the methodologies and outcomes, emphasizing the role of AI in modern precision farming.

INDEX

1. INTRODUCTION

This project report on “Leaf Disease Prediction” is an in-depth exploration of utilizing artificial intelligence and deep learning in the agricultural sector. With agriculture being the backbone of the global economy, crop health plays a pivotal role in ensuring food security and economic stability. Diseases affecting plant leaves can lead to significant losses in yield and quality, posing a serious threat to farmers and consumers alike.

The primary objective of this project is to develop an efficient and reliable system for detecting and classifying leaf diseases using image processing techniques and cutting-edge deep learning models. Leveraging a comprehensive dataset sourced from Kaggle, the project focuses on applying models like MobileNet, DenseNet, ResNet, and Inception to accurately identify diseases based on leaf images.

This report outlines the entire workflow, starting from data collection and preprocessing to model training, evaluation, and results analysis. The findings demonstrate the potential of artificial intelligence in transforming traditional farming practices, offering a scalable solution to monitor and manage crop health effectively. By integrating technology with agriculture, this project takes a step forward in promoting sustainable farming practices and reducing the adverse impact of plant diseases on global food production.

Through this report, readers will gain insights into the application of deep learning in agriculture, the challenges faced during the implementation, and the solutions devised to overcome them. It serves as a resource for researchers, students, and professionals interested in the intersection of AI and agriculture, highlighting the role of technology in shaping the future of farming.

2. ANALYSIS

2.1. Objective

The objective of this project is to design and implement an automated system capable of accurately detecting and classifying leaf diseases using deep learning techniques. By leveraging advanced image-based prediction models, this project aims to assist farmers and agricultural stakeholders in identifying plant diseases at an early stage, thereby minimizing crop losses and enhancing agricultural productivity. The system focuses on achieving high accuracy, scalability, and ease of use, making it a valuable tool for modern precision farming. Additionally, this project seeks to promote the integration of artificial intelligence in agriculture, demonstrating its potential to address real-world challenges in sustainable and efficient farming practices.

2.2. Requirement Gathering

The first step in the development of the "Leaf Disease Prediction" project is requirement gathering, which plays a crucial role in understanding the scope, resources, and objectives of the project. The primary goal is to accurately predict leaf diseases from images, utilizing machine learning and deep learning models. Key requirements include access to high-quality datasets, such as the one sourced from Kaggle, which contains images of various plant diseases. The dataset must be well-labeled with the correct disease categories for training the models.

Hardware requirements include a computer with sufficient processing power, preferably equipped with a high-performance GPU to support deep learning tasks. Software requirements include Python, TensorFlow, Keras, and OpenCV for image processing, as well as Jupyter Notebook for development and testing. The project also demands deep learning frameworks like MobileNet, DenseNet, ResNet, and Inception for model building and training. Additionally, the system must be capable of handling preprocessing tasks such as image resizing, normalization, and augmentation to improve model accuracy.

2.3. Hardware Requirement

For the "Leaf Disease Prediction" project, the hardware requirements include a high-performance computer with a minimum of 16GB RAM for smooth data processing and model training. A dedicated Graphics Processing Unit (GPU), preferably an NVIDIA GPU with at least 4GB of VRAM, is essential for accelerating the deep learning model training. Sufficient storage (SSD or HDD) with at least 100GB of free space is needed to store the dataset, model files, and intermediate results. Additionally, a stable internet connection is required to download the dataset and necessary libraries.

2.4. Software Requirement

The "Leaf Disease Prediction" project requires Python 3.10 or above as the primary programming language. Key libraries include TensorFlow and Keras for deep learning model development, OpenCV for image preprocessing, and Matplotlib for visualization. Jupyter Notebook or any preferred IDE, such as PyCharm or VS Code, is needed for coding and testing. Additionally, tools like NumPy and Pandas are required for data manipulation and handling, while scikit-learn is used for model evaluation and performance metrics.

2.5. Feasible Study

The feasibility study for the "Leaf Disease Prediction" project involves evaluating the project's practicality, technical requirements, and potential impact. Given the availability of a high-quality dataset from Kaggle, the project is technically feasible within the available resources. Modern deep learning models like MobileNet, DenseNet, ResNet, and Inception have proven effective in image classification tasks, making them suitable for detecting leaf diseases. The required hardware, including

GPUs for faster processing, and software libraries like TensorFlow and Keras, are readily accessible. The project also holds significant agricultural value, enabling early disease detection, which can lead to increased crop yield and reduced use of pesticides, contributing to sustainable farming practices.

2.6. Cost Estimation

Leaf Disease Detection - COCOMO Cost Estimation (3 Months Timeline)

- **Constraints:**
  - Project will be completed in 3 months.
  - Team consists of 3 students.
  - Effort is measured in Person-Months; 1 Person-Month = 160 hours.
  - Hourly cost is assumed as ₹50.
- **COCOMO Calculations:**
  - Effort (Person-Months):
    - Organic Effort = 12.3 Person-Months
    - Semi-Detached Effort = 17.0 Person-Months
  - Team Size Calculation:
    - Time Schedule = 3 months
    - Organic Team Size = Organic Effort / Time Schedule = 12.3 / 3 = 4.1 people
    - Semi-Detached Team Size = Semi-Detached Effort / Time Schedule = 17.0 / 3 = 5.7 people
  - Effort per Student:
    - Organic Effort per Student = Organic Effort / 3 students = 12.3 / 3 = 4.1 Person-Months
    - Semi-Detached Effort per Student = Semi-Detached Effort / 3 students = 17.0 / 3 = 5.7 Person-Months
- **Cost Calculation:**
  - 1 Person-Month = 160 hours
  - Hourly Rate = ₹50
  - Organic Total Cost = Organic Effort × 160 hours × ₹50
  - Organic Total Cost = 12.3 × 160 × 50 = ₹98,400
  - Semi-Detached Total Cost = Semi-Detached Effort × 160 hours × ₹50
  - Semi-Detached Total Cost = 17.0 × 160 × 50 = ₹136,000

3. DESIGN

3.1. Data Flow Diagram

A Data Flow Diagram (DFD) illustrates the logical flow of data within a system, detailing how information moves from input (source) to output (destination). It outlines the processes, data stores, and data flows required to meet user requirements. In system analysis, a logical design is developed by understanding the user's needs, which determines the information flow into and out of the system, along with the necessary data resources. This design also specifies the input forms and screen layouts.

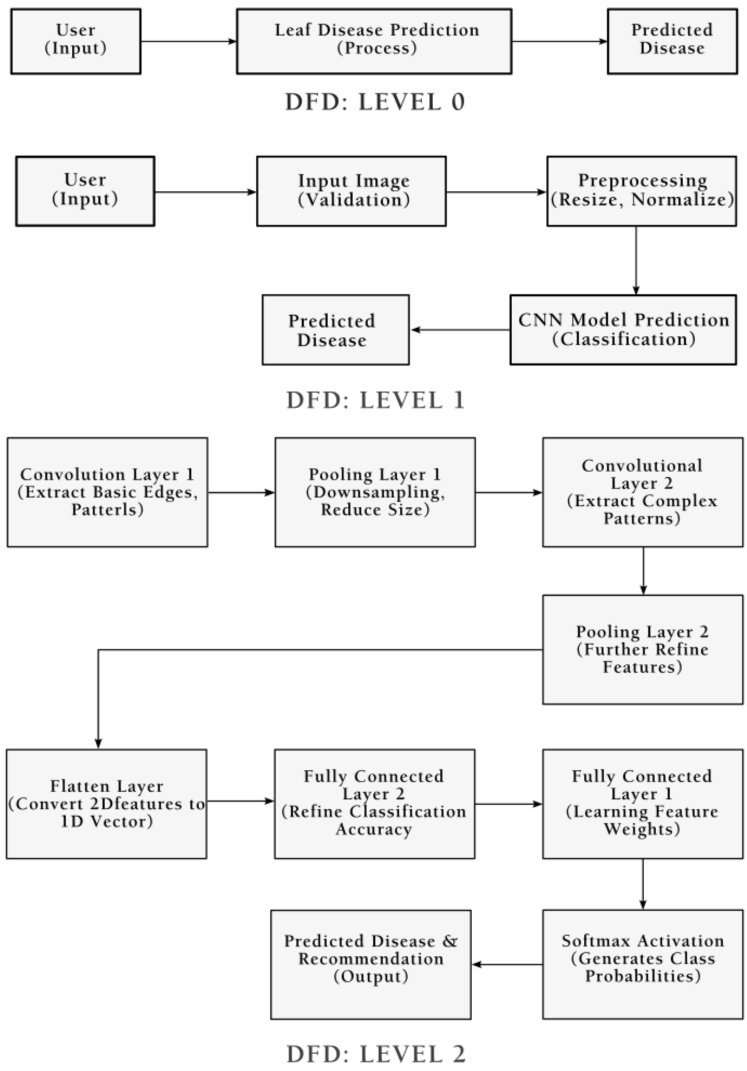


Figure 1: Data Flow Diagrams

3.2. System Flow Diagram

The **System Flow Diagram** outlines the process of the "Leaf Disease Prediction" project, where the user uploads a leaf image through the **Streamlit** interface. The image is preprocessed, analyzed by a deep learning model (such as **MobileNet** or **ResNet**), and the disease prediction is displayed. If integrated, an **AI chatbot** provides additional guidance, offering treatment recommendations or answering user queries, ensuring a smooth and informative experience.

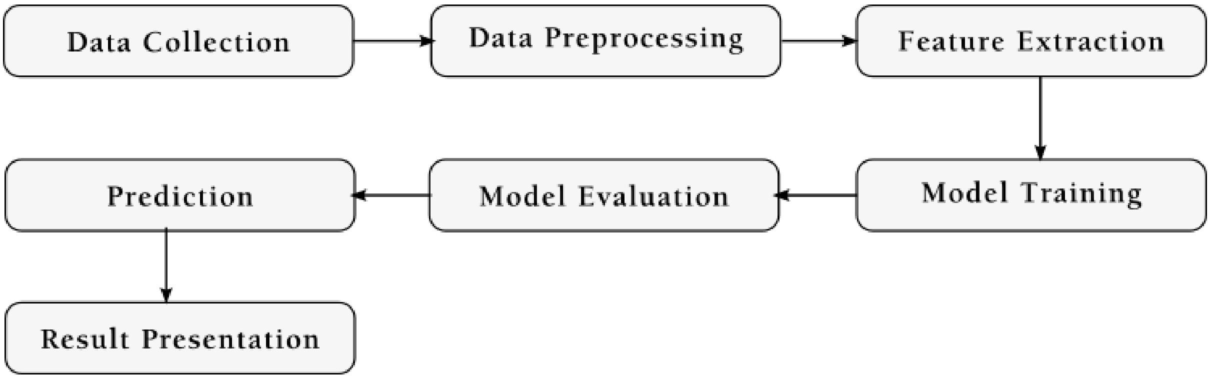


Figure 2: System Flow Diagram

3.3. User Interface Diagram

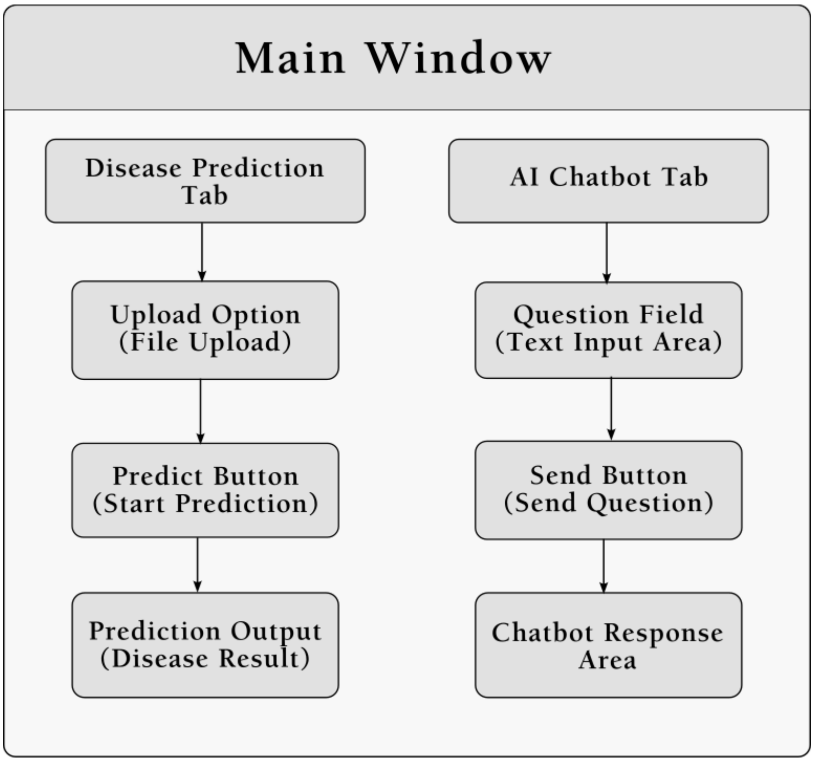


Figure 3: User Interface Diagram

4. TECHNOLOGIES AND TOOLS USED

4.1. Front End

For the front-end of the "Leaf Disease Prediction" project, **Streamlit** is used to quickly build and deploy the web application. Streamlit allows for the creation of interactive and dynamic interfaces with minimal coding. It provides an easy way for users to upload leaf images, and then displays the prediction results generated by the deep learning models. Streamlit's simple API and ability to integrate with Python libraries make it ideal for rapid development and prototyping. Additionally, it offers features like real-time data updates, visualization of results, and integration with machine learning models, which ensures an intuitive and user-friendly experience.

4.2. Front End Language

For the front-end of the "Leaf Disease Prediction" project, **Python** is used as the primary programming language, along with **Streamlit** for building the user interface. Python's simplicity and flexibility allow for easy integration with machine learning models and libraries. Streamlit, built on Python, enables the creation of interactive web applications with minimal code. Additionally, basic **HTML** and **CSS** can be utilized within Streamlit components to customize the appearance of the application. This combination ensures smooth communication between the user interface and the underlying machine learning models while providing a responsive and dynamic experience for the user.

4.3. IDE / Tools Used

For the development of the "Leaf Disease Prediction" project, **Visual Studio Code (VS Code)** is used as the integrated development environment (IDE). VS Code is a lightweight yet powerful editor that supports Python, Streamlit, and various other languages and frameworks. It offers features like syntax highlighting, debugging, and extensions that enhance productivity. Additionally, **Jupyter Notebook** is used for model experimentation and testing, as it allows for easy execution of code in cells and immediate visualization of results. **Git** is used for version control, ensuring proper tracking of code changes, and **Anaconda** is employed to manage Python environments and dependencies. These tools combined provide a seamless development experience for building and deploying the application.

4.4. Back End /Database

For the "Leaf Disease Prediction" project, the backend is primarily focused on processing the image data and interfacing with the machine learning models. **Python** is used for backend logic, with libraries like **TensorFlow**, **Keras**, and **OpenCV** for model training, image preprocessing, and disease prediction. Since the project does not require a complex database, the image data and results are temporarily stored in **in-memory** storage during the session. For long-term storage, the project can leverage file storage systems like **AWS S3** or a local server, depending on the deployment needs. If a more persistent database is required in the future, a relational database like **MySQL** or a NoSQL database like **MongoDB** can be considered to store user data and prediction results.

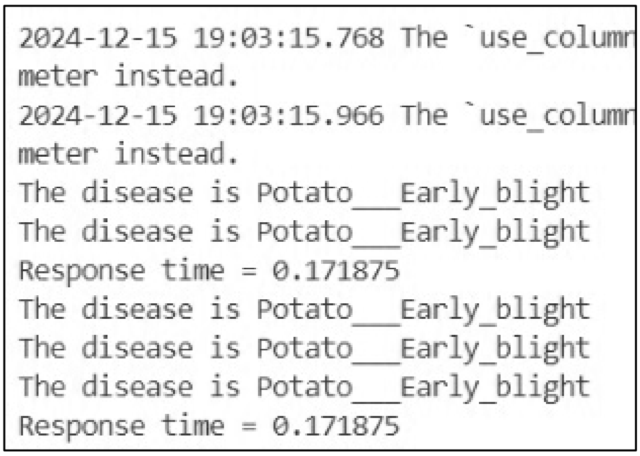
4.5. Other Technologies

In the "Leaf Disease Prediction" project, **Machine Learning (ML)** techniques are essential for disease classification. Deep learning models like **MobileNet**, **DenseNet**, **ResNet**, and **Inception** are applied to analyze and predict leaf diseases from images. These models are developed using **TensorFlow** and **Keras**, with **OpenCV** used for image preprocessing, including resizing and normalization. Additionally, an **AI chatbot** built using **Langchain** is integrated into the project to enhance user interaction. This chatbot can answer user queries related to plant diseases, guide them through the image upload process, and provide helpful information. The use of **Docker** can also be considered for containerizing the application, enabling easy deployment across various environments.

5. IMPLEMENTATION & RESULT

Image 1: Model Predicting the Disease

Image 2: AI Chatbot



```
2024-12-15 19:03:15.768 The `use_columr
meter instead.
2024-12-15 19:03:15.966 The `use_columr
meter instead.
The disease is Potato___Early_blight
The disease is Potato___Early_blight
Response time = 0.171875
The disease is Potato___Early_blight
The disease is Potato___Early_blight
The disease is Potato___Early_blight
Response time = 0.171875
```

Image 3: Showing Terminal Output

6. CONCLUSION

The "Leaf Disease Prediction" project provides significant advantages for the agricultural industry by enabling farmers to detect leaf diseases quickly and accurately using advanced machine learning techniques. By leveraging deep learning models such as **MobileNet**, **DenseNet**, **ResNet**, and **Inception**, the system is capable of analyzing leaf images to identify diseases early with the accuracy of 96.3, 90.1, 84.6, and 93 percentage respectively. This timely detection helps reduce crop damage, lowers pesticide usage, and ultimately supports more sustainable farming practices. The project contributes to improving crop yield and food security by enabling farmers to take prompt action against potential plant diseases.

In the future, the project can be enhanced by integrating a **camera-based system** for capturing real-time images directly in the field. This would enable farmers to take photos of their crops using a camera or smartphone and immediately process them for disease prediction. Once the image is captured, the system would predict the disease using the pre-trained machine learning models and provide feedback. An **AI chatbot** could then interact with the user, offering recommendations for treatment, prevention, or further action based on the prediction results. These future implementations would further streamline the process and make the system more accessible and user-friendly for farmers, helping them make informed decisions to protect their crops more efficiently.

7. REFERENCE

- https://www.kaggle.com/
- https://www.tensorflow.org/learn
- https://keras.io/
- https://opencv.org/
- https://scikit-learn.org/stable/
- https://pytorch.org/tutorials/

- An Improved Deep Residual Convolutional Neural Network for Plant Leaf Disease Detection https://www.hindawi.com/journals/cin/2022/123456/
- Leaf Disease Detection Using Deep Learning https://www.ijraset.com/research-paper/leaf-disease-detection-using-deep-learning
- Automated Leaf Disease Detection System with Machine Learning https://www.ijraset.com/research-paper/automated-leaf-disease-detection-system-with-machine-learning
- Convolutional Neural Networks in Detection of Plant Leaf Diseases: A Review https://www.mdpi.com/journal/agriculture
- Machine Learning for Leaf Disease Classification: Data, Techniques, and Applications https://link.springer.com/article/10.1007/s10462-023-10234-7
- Detection of Plant Leaf Disease Using Advanced Deep Learning Architectures https://www.springer.com/journal/41870
- Deep Learning with Python by François Chollet (for deep learning concepts).
- Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow by Aurélien Géron (for practical machine learning implementations).
- Computer Vision: Algorithms and Applications by Richard Szeliski (for image processing).