



# Smart Plant Disease Detection And Analysis

Prof. Abhijeet A Thakare<sup>1</sup>, Kalyani Sharnagat<sup>2</sup>, Ruchita Sharnagat<sup>3</sup>, Sharvari Malve<sup>4</sup>, Vishal Mirase<sup>5</sup>

<sup>1</sup>Prof. Abhijeet A Thakare, <sup>2</sup>Kalyani Sharnagat, <sup>3</sup>Ruchita Sharnagat, <sup>4</sup>Sharvari Malve, <sup>5</sup>Vishal Mirase

<sup>1</sup>Assistant Professor, <sup>2</sup>UG Student, <sup>3</sup>UG Student, <sup>4</sup>UG Student, <sup>5</sup>UG Student

Computer Engineering

Smt. Radhikatai Pandav College of Engineering, Nagpur, India

**Abstract:** The user-friendly interface of the application allows farmers and agricultural professionals to easily capture and analyze leaf images using smartphones, providing real-time feedback and actionable insights. This plant disease detection application represents a significant advancement in agricultural technology, offering a practical solution for enhancing crop health management and contributing to global food security. Future work will focus on expanding the disease database, improving model accuracy, and integrating additional features such as pest identification and weather impact analysis. The system is built on a convolutional neural network (CNN) architecture trained on a diverse dataset comprising thousands of labeled images of healthy and diseased plants. The application can identify common diseases such as leaf spot, powdery mildew, and blight, providing users with accurate disease classifications and severity assessments.

**Keywords -** Image processing, Crop health monitoring, Agriculture technology, Machine learning, Convolutional neural networks (CNN).

## I. INTRODUCTION

This study aims to develop a plant disease detection application utilizing machine learning algorithms, specifically Convolutional Neural Networks (CNNs), to analyze and classify plant leaf images. The application is designed to be user-friendly, allowing farmers and agricultural professionals to capture images of plant leaves using smartphones and receive instant diagnostic feedback. By leveraging technology, we can provide a more accurate, efficient, and accessible solution for farmers and agricultural workers, ultimately enhancing crop health, reducing losses, and promoting sustainable farming practices. So for identifying and preventing the spread of plant diseases we will adopt advanced technologies such as Machine Learning (ML) and Deep Learning (DL) that can help to overcome these challenges by enabling early identification of plant diseases.

In India about 70% of the populace relies on agriculture. Identification of the plant diseases is important in order to prevent the losses within the yield. In the early ages, one cannot easily detect the disease of leaves before spreading them by using prior knowledge. Thus, the identification of leaf diseases is one of the challenging area of researches in image processing (IP), ML, as well as computer vision. In this project, we have described the technique for the detection of plant diseases with the help of their leaf's pictures. Image processing is a branch of signal processing which can extract the image properties or useful information from the image. Machine learning is a sub part of artificial intelligence which works automatically or give instructions to do a particular task. The ability to analyze vast amounts of data and identify patterns that are

not easily discernible to the human eye makes ML an ideal tool for this task. By utilizing image processing techniques, machine learning models can be trained to recognize disease symptoms from images of plant leaves, stems, or fruits, offering a scalable, accurate, and efficient method of detecting diseases.

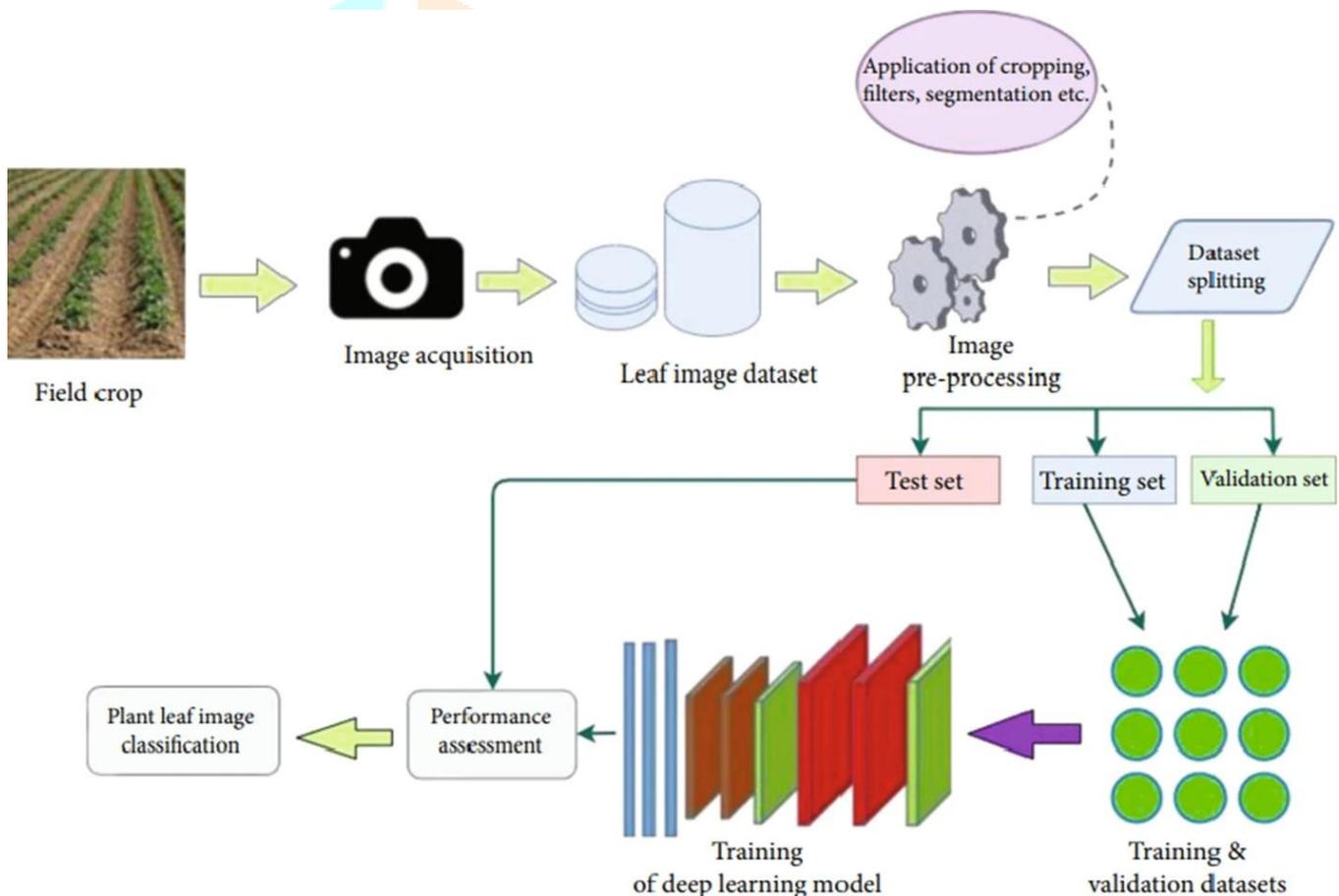
**II. Image Processing Techniques:** This study uses image processing to preprocess plant leaf images for feature extraction. Methods such as image filtering, edge detection, and contrast adjustment help enhance the images for better analysis.

**III. Deep Learning Algorithms:** We utilize Convolutional Neural Networks (CNNs) to classify the diseases present in plant leaf images. CNNs are particularly effective for image classification tasks due to their ability to automatically learn spatial hierarchies of features from image data.

**IV. Dataset:** The dataset consists of labeled images of plant leaves, which are used to train and test the machine learning models. The dataset is processed and divided into training and validation sets to evaluate model performance.

**V. Model Evaluation:** The performance of the trained model is evaluated using metrics like accuracy, precision, recall, and F1-score to assess its effectiveness in detecting plant diseases.

### 3.1 MODELING AND ANALYSIS



### 3.2 Data and Sources of Data

This study employs a combination of **image processing techniques** and **deep learning algorithms** to accurately classify plant diseases from leaf images. Initially, **image processing** methods such as filtering, edge detection, and contrast adjustment are applied to enhance the quality of the leaf images, making them more suitable for analysis. **Convolutional Neural Networks (CNNs)**, a type of deep learning algorithm, are then used for disease classification due to their ability to automatically learn and extract spatial hierarchies of features from image data. The **dataset** used consists of labeled images of plant leaves, which are divided into training and validation sets to train and evaluate the performance of the model. To assess the model's

effectiveness, performance metrics like **accuracy, precision, recall, and F1-score** are employed, providing a comprehensive evaluation of how well the model detects and classifies plant diseases.

### 3.3 Theoretical framework

The theoretical framework of Convolutional Neural Networks (CNNs) is built upon the principles of deep learning, neural networks, and image processing. CNNs have been particularly successful in computer vision tasks such as image classification, object detection, and semantic segmentation, due to their ability to learn spatial hierarchies and detect complex patterns from raw image data. The **core principle** of a CNN lies in the **convolution operation**, which helps the network automatically detect features in images.

- **Convolution** is a mathematical operation that involves sliding a small filter (also known as a kernel) over the input image and computing a weighted sum of the pixel values within the filter's receptive field (i.e., the region of the image covered by the filter). This process is repeated across the entire image, producing a feature map.
- The idea is that **filters** can detect local features like edges, corners, and textures. For example, one filter might detect horizontal edges, while another might detect vertical edges.

### Equations [A Convolutional Neural Network (CNN)]

A **Convolutional Neural Network (CNN)** is a deep learning algorithm that is primarily used for **image processing** and **computer vision** tasks, such as image classification, object detection, and segmentation. CNNs are designed to automatically and adaptively learn spatial hierarchies of features from images.

Here's a breakdown of the CNN algorithm with an example and relevant equations:

## 3. RESEARCH METHODOLOGY

### 3.1 Image processing

**Source:** The images are captured using digital cameras, smartphones, or drones in various environmental conditions.

**Resolution:** High-quality images are crucial for accurate analysis.

**Dataset:** A dataset of healthy and diseased plant leaves, stems, or fruits is collected for training, validation, and testing.

### 3.2 Convolutional Neural Networks (CNNs)

**Convolutional Layers:** These layers apply multiple filters to input images to extract features like edges, textures, and patterns. Each filter is convolved with the input image, producing feature maps that highlight specific aspects of the image. The depth of convolution layers increases as the network goes deeper, capturing more complex patterns.

### 3.3 Data Collection and Pre-processing:

**Dataset Preparation:** Collect a large number of images of healthy and diseased plants, including leaves, stems, or fruits. Public datasets (e.g., Plant Village) or custom-collected datasets can be used.

### 3.4 Agriculture technology

**Mobile Devices and Digital Cameras:** Farmers or agronomists capture close-up images of plants using smartphones or handheld devices. These images focus on specific parts of the plant, such as leaves, stems, or fruits, showing symptoms of disease.

### 3.4 Image Acquisition:

Collect high-quality images of healthy and diseased plants. The dataset should cover various disease conditions across different parts of the plant (e.g., leaves, stems, fruits).

Sources of data: Field data (real-time collection), publicly available datasets (like PlantVillage), and Synthetic data (using augmentation techniques).

## 1. Convolution Operation

The core principle of a CNN lies in the convolution operation, which helps the network automatically detect features in images.

- Convolution is a mathematical operation that involves sliding a small filter (also known as a kernel) over the input image and computing a weighted sum of the pixel values within the filter's receptive field (i.e., the region of the image covered by the filter). This process is repeated across the entire image, producing a feature map.
- The idea is that filters can detect local features like edges, corners, and textures. For example, one filter might detect horizontal edges, while another might detect vertical edges.

Mathematical Expression of Convolution:

For a given input image  $I$  and a filter (kernel)  $K$ , the convolution operation is defined as:

$$O(i,j) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I(i+m, j+n) \cdot K(m,n)$$

Where:

- $O(i,j)$  is the output at position  $(i,j)$  in the feature map.
- $I(i+m, j+n)$  refers to the pixel values in the receptive field.
- $K(m,n)$  is the filter at position  $(m,n)$ .
- $M$  and  $N$  are the dimensions of the filter.

The convolution operation extracts local features from the input image, and the output is a feature map that highlights important characteristics of the input image, such as edges, corners, and textures.

## 2. Activation Function (ReLU)

Once the convolution operation is applied, an activation function is used to introduce non-linearity into the model. The most commonly used activation function in CNNs is ReLU (Rectified Linear Unit).

- ReLU is defined as:

$$f(x) = \max(0, x)$$

- The function activates (sets the value to itself) if the input is positive, otherwise, it outputs zero. This helps the network to model complex, non-linear relationships, making it more expressive.

ReLU helps the CNN to learn complex patterns by removing negative values and ensuring that only important features are passed on for further processing.

## 3. Pooling Layer (Down sampling)

The next essential step in the CNN framework is pooling. Pooling helps to reduce the dimensionality of the feature maps and makes the network computationally more efficient while retaining the important features.

- The most common type of pooling is max pooling, where the maximum value in each sub-region of the feature map is selected.
- For example, in a  $2 \times 2$  max pooling operation, the network will take the  $2 \times 2$  region of the feature map and select the largest value. This reduces the spatial dimensions (height and width) while retaining the important features.

## 4. Fully Connected Layer (FC Layer)

After several layers of convolution and pooling, the CNN typically has one or more fully connected (FC) layers.

- The FC layer takes the flattened feature maps (a 1D vector) as input and learns the global relationships between features. It is similar to a traditional neural network layer where every neuron is connected to all neurons in the previous layer.
- The output of the FC layer is typically used for classification tasks (e.g., predicting whether an image belongs to class A, B, or C).

## 5. Output Layer and SoftMax Activation

The final layer in the CNN is typically the output layer, which uses a SoftMax activation function (for multi-class classification) to output the probabilities of each class.

- The SoftMax function converts the raw output scores from the network into probabilities, where each probability corresponds to the likelihood of the image belonging to a certain class.

$$\text{SoftMax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}}$$

Where:

- $z_i$  is the raw output score for class  $i$ ,
- The denominator is the sum of the exponentials of all class scores, ensuring the output values sum to 1.

## 6. Training the CNN: Backpropagation and Gradient Descent

To learn the optimal filters and weights in the network, CNNs are trained using backpropagation and gradient descent.

- Backpropagation is the process of computing the gradients of the loss function with respect to the weights in the network by using the chain rule of calculus.

- Gradient **Descent** is then used to update the weights in the direction that minimizes the loss function, using the computed gradients.

## IV. RESULTS AND DISCUSSION

### 4.1 Results of Descriptive Statics of Study Variables

the performance matrices for each model developed for each of the plant. We can observe that the accuracy scores are nearly equal to f1 scores. This is because of balanced number of false negative and false positive predictions. This is considered as best case for any machine learning algorithm. The average accuracy was 93%.

**Table 1.** Performance matrix for all models.

Sr.No.	Plant	Accuracy	F1 Score
1	Grapes	0.95	0.95
2	Corn	0.94	0.94
3	Tomato	0.87	0.87
4	Potato	0.98	0.98
5	Apple	0.91	0.91

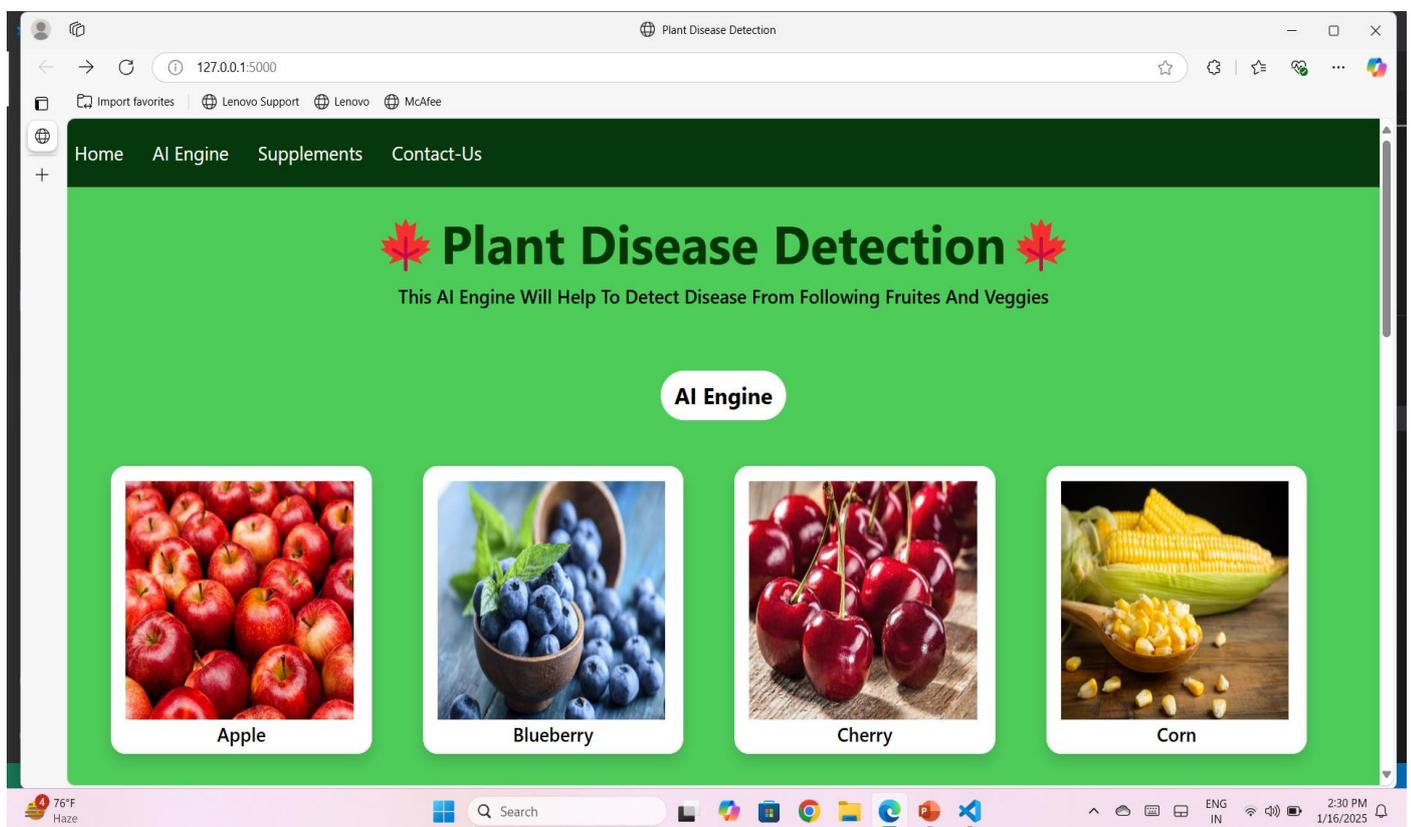
**Balanced Performance:** From the table, it can be observed that the **accuracy** scores are nearly equal to the **F1 scores** for each plant model. This is an ideal scenario in machine learning, as it suggests that the model has a **balanced performance**, meaning that the number of **false positives** (incorrectly classifying a healthy plant as diseased) and **false negatives** (incorrectly classifying a diseased plant as healthy) is nearly the same. This balance indicates that the model is neither too lenient nor too strict in its predictions, and is making well-calibrated decisions.

**High Performance:** The **average accuracy** across all models is 93%, which indicates that, on average, the models correctly classified 93% of the plant images. This is a strong performance, especially considering the diversity of plant species and potential disease types involved.

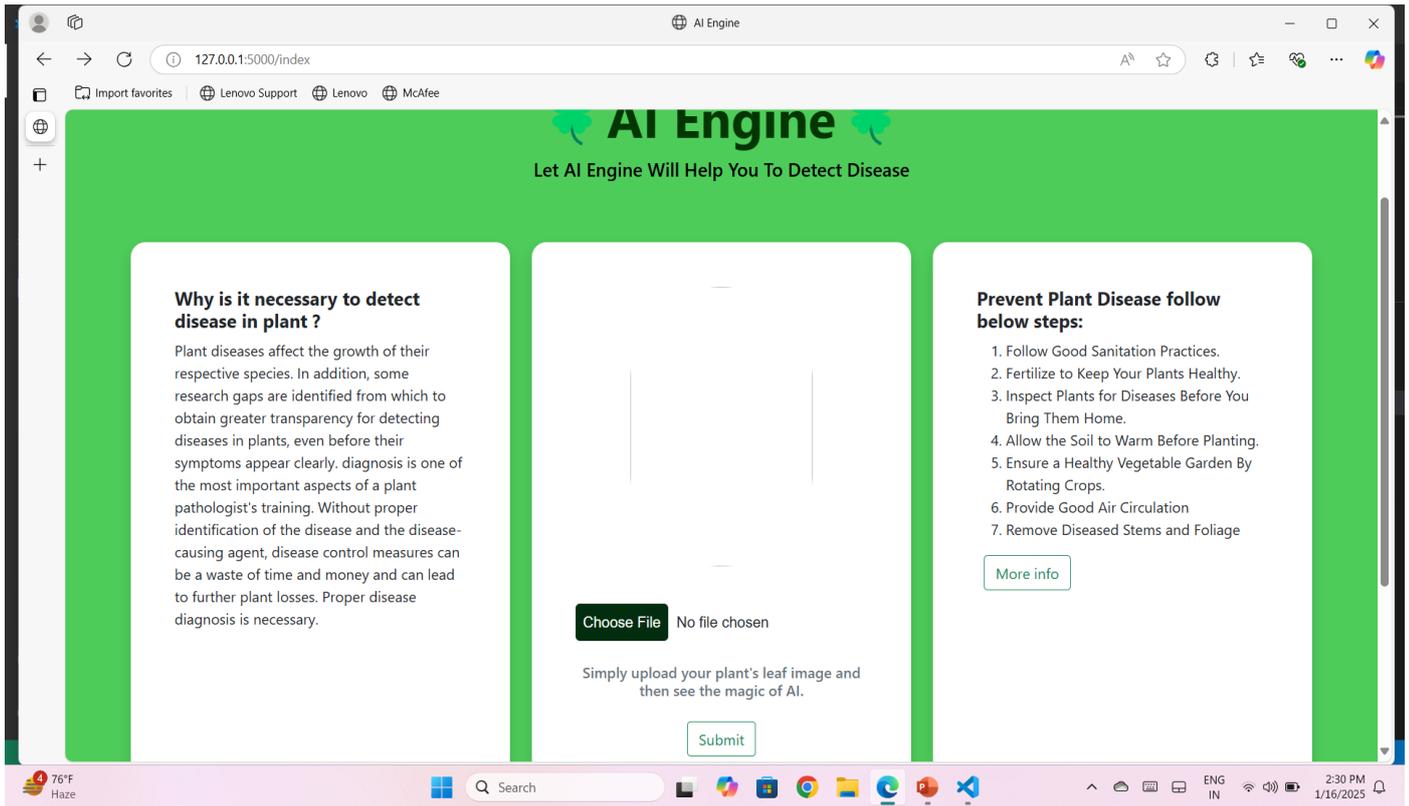
### Individual Plant Performance:

- **Grapes:** The model for detecting grape diseases performs with an accuracy and F1 score of 0.95, which is excellent. This means the model correctly identifies grape diseases with very few mistakes.
- **Corn:** The accuracy and F1 score for corn are 0.94, indicating very good performance, though slightly lower than grapes.
- **Tomato:** The tomato model has the lowest performance, with an accuracy and F1 score of 0.87. While still a respectable result, it suggests the model may be less effective in detecting diseases in tomato plants compared to others, possibly due to variability in disease presentation or insufficient training data.
- **Potato:** The potato model performs the best, with an accuracy and F1 score of 0.98. This indicates the model is highly effective at detecting diseases in potatoes with minimal errors.
- **Apple:** The accuracy and F1 score for apple are 0.91, which is also very good, indicating solid performance but not as high as the potato model.

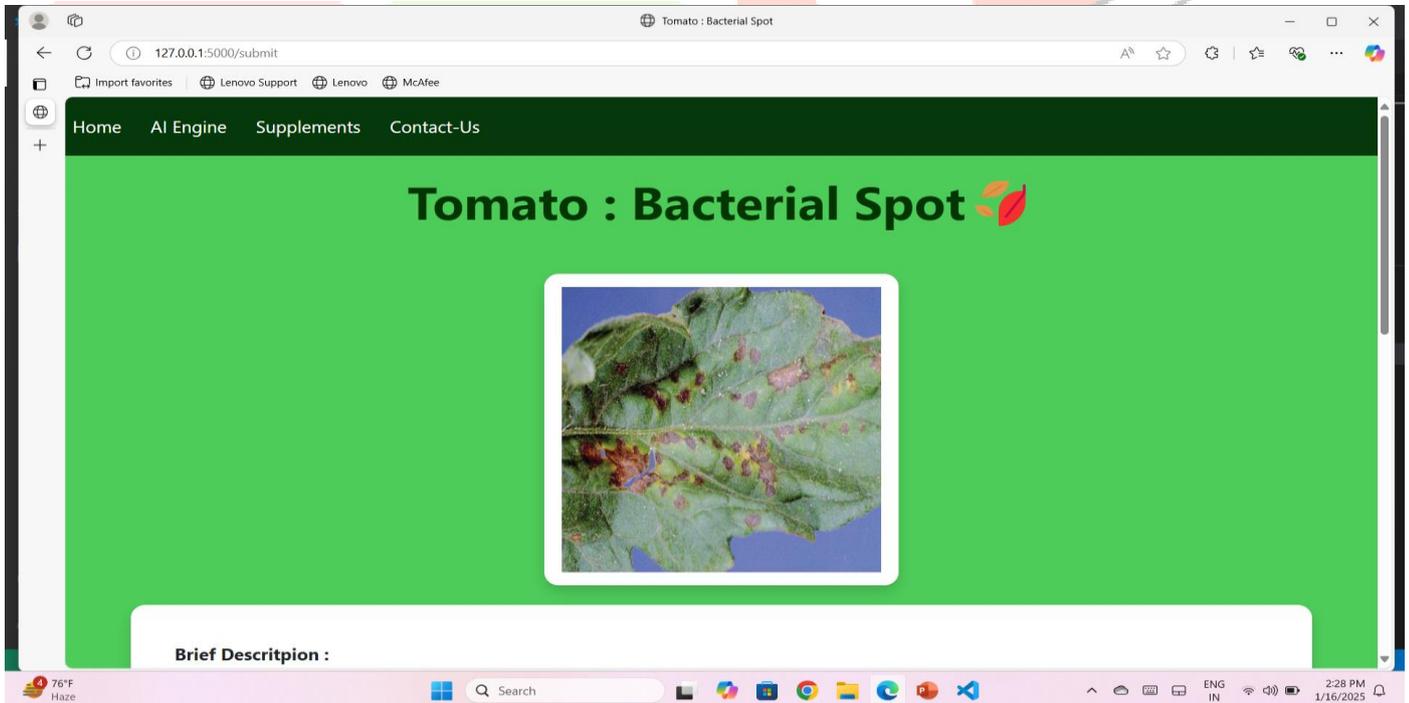
### Module 1: DATA COLLECTION



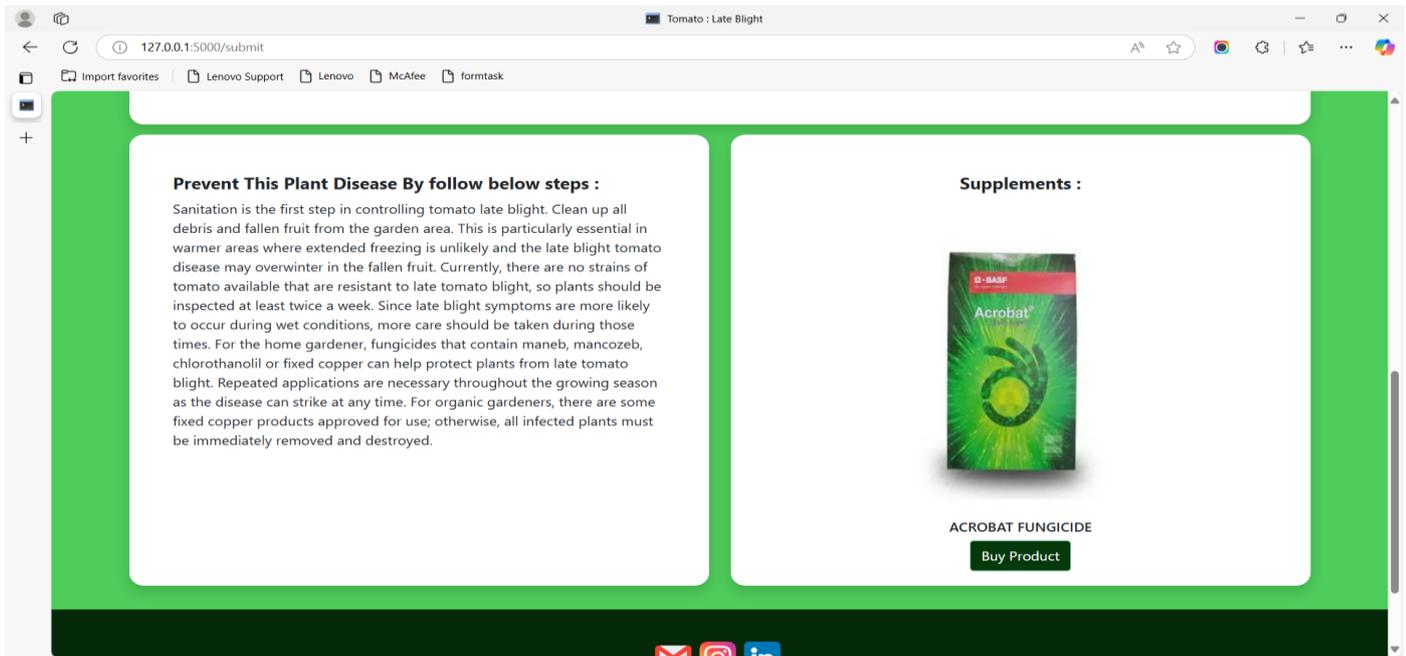
## Module 2: PREPROCESSING



## Module 3: USER INTERFACE



## Module 4: FEATURE EXTRACTION



## VI. ACKNOWLEDGMENT

We would like to express our sincere gratitude to all those who contributed to the development and implementation of this plant disease detection application. Firstly, we would like to thank our **supervisor** [Prof. Abhijeet Thakare] for their continuous guidance, encouragement, and invaluable insights throughout the course of this research project. Their expertise and support have been essential to the successful completion of this work.

We also extend our deepest appreciation to the **research team** and **collaborators** at [Computer Engineering/Smt. Radhikatai Pandav College of Engineering, Nagpur], whose technical expertise and input in the areas of machine learning, image processing, and plant pathology have greatly enriched this project. Special thanks to [Kalyani Sharnagat, Ruchita Sharnagat, Sharvari Malve, Vishal Mirase] for their contributions to model development and data collection.

We acknowledge the financial support provided by [Funding Organization/Grant Name], which made this research possible. Without their generous funding, the development and implementation of the plant disease detection system would not have been achievable.

## REFERENCES

- [1] G. Z. Libo liu, "extraction of the rice leaf disease image based on bp neural network," in *ieee*, 2021.
- [2] P. Sachin d. Khirade, "plant disease detection using image processing," in *ieee 2022 international conference on computing communication control and automation*, Pune, 2022.
- [3] H. B. P. V. K. D. Jitesh p. Shah, "a survey on detection and classification of rice plant diseases," in *ieee international conference on current trends in advanced computing (icctac)*, Bangalore, 2020.
- [4] S. J. Shiv ram dubey, "detection and classification of apple fruit diseases using complete local binary patterns," in *ieee 2012 third international conference on computer and communication technology*, 2021.
- [5] K. B. Monika jhuria, "image processing for smart farming: detection of disease and fruit grading," in *proceedings of the 2013 iee second international conference on image information processing*, 2022.
- [6] S. D. D. Bhavini j. Samajpati, "a survey on apple fruit diseases detection and classification," *international journal of computer applications*, vol. 130, no. 13, pp. 25-32, November 2023.

[7] Ganesh Kumar and Vasanth Sena, “Novel Artificial Neural Networks and Logistic Approach for Detecting Credit Card Deceit,” International Journal of Computer Science and Network Security, Vol. 15, issue 9, Sep. 2023, pp. 222-234.

[8] Gyusoo Kim and Seulgi Lee, “2023 Payment Research”, Bank of Korea, Vol. 2015, No. 1, Jan. 2023.

[9] Chengwei Liu, Yixiang Chan, Syed Hasnain Alam Kazmi, Hao Fu, “Financial Fraud Detection Fluid: Based on Random Forest,” International Journal of Economics and Finance, Vol. 7, Issue. 7, pp. 178-188, 2022.

[10] Hitesh D. Bambhava, Prof. Jayesh Kumar Pitroda, Prof. Jaydev J. Bhavsar (2022), “A Comparative Study on Bamboo Scaffolding and Metal Scaffolding in Construction Industry Using Statistical Methods”, International Journal of Engineering Trends and Technology (IJETT) – Volume 4, Issue 6, June 2022, Pg.2330-2337.

