ISSN NO: 2350-1146 I.F-5.11

Image-Based Plant Disease Prediction Using Machine Learning Techniques

Prof. Jotsna H. Chavhan
Department of Computer Technology
KaviKulguru Institute of Technology and Science
Ramtek, Nagpur, India
jotchavhan@gmail.com

Prof. Dipti A. Mirkute
Department of Computer Science and Engineering
Jawaharlal Darda Institute of Engineering and Technology
Yavatmal, Maharashtra, India
diptijilhare1984@gmail.com

Abstract - **Plant diseases pose a significant threat to agricultural productivity, causing economic losses and worsening food insecurity. Early and accurate detection is vital for effective management and sustainable farming. This study explores the use of machine learning to develop a reliable system for plant disease prediction. Using image processing and classification algorithms, the system analyzes leaf images to identify disease patterns with high accuracy. Convolutional Neural Networks (CNNs) and other advanced models help distinguish between healthy and diseased plants. Experimental results show the system can detect various diseases with high precision and minimal human input, aiding farmers in decision-making and improving crop health and productivity.**

Index Terms – Plant Disease Detection, Machine Learning, Image Classification, Convolutional Neural Network (CNN), Leaf Image Analysis, Disease Prediction System.

I. INTRODUCTION

Agriculture serves as the foundation of economic stability and growth and crop health plays a vital role in ensuring good yield. Farmers often face losses due to crop diseases, which can spread quickly if not detected early. Traditional methods of identifying diseases are time- consuming and require expert knowledge. To address this challenge, propose a Crop Disease Prediction System using Machine Learning. This system helps farmers detect diseases at an early stage using image processing and classification techniques. It analyzes images of crops and predicts possible diseases with high accuracy. Machine learning paradigms are trained on a large dataset of diseased and healthy crops. The approach specifies real-time disease detection and suggests preventive measures. It reduces dependency on experts and speeds up decision-making. Farmers can access predictions using a mobile or web application. Early detection helps in reducing crop loss and improving productivity. The system is cost-effective and easy to use for farmers. It supports multiple crops and diseases for better coverage. By using artificial intelligence make agriculture more efficient. This system promotes sustainable farming practices. It ultimately helps in ensuring food security and economic growth.

II. PROPOSED APPROACH AND SYSTEM ARCHITECTURE

A. Modules

1) Data Collection

A well-defined data collection process is essential for developing an accurate machine learning-based plant disease detection system. The dataset used comprises high-quality images of healthy and diseased plant leaves, obtained from Kaggle and agricultural research sources. Additionally, high-resolution images are manually captured using digital cameras and smartphones under controlled lighting conditions to ensure consistency and clarity. The accuracy of any machine learning algorithm depends on image of input and therefore correctness of the training dataset.

Datasets include: Yield dataset val, test, train.

2) Data Pre-processing

Data Cleaning is the first step, where duplicate, irrelevant, low-quality or blurry images are removed to improve dataset reliability. Standardization of image size and resolution ensures uniformity, making it easier for machine learning models to process the data. Data Augmentation is applied to increase dataset diversity and prevent overfitting. Techniques such as rotation, flipping, brightness adjustment and scaling help the model generalize better across different conditions. Image segmentation is used to isolate diseased regions from the background, making classification more accurate.

3) Training ML Model

Training a plant disease detection model involves proper data preparation, selecting suitable learning techniques, and optimizing performance. Without using IoT devices, the model relies on image-based datasets from public sources and manually captured images. The dataset is split into 70% training, 20% validation, and 10% testing. Pre-processing steps such as noise reduction, contrast enhancement, resizing, and normalization are applied to improve image quality and consistency. During training, labeled plant leaf images are used to train Convolutional Neural Networks (CNNs), which automatically learn patterns in color, texture, and shape to accurately identify healthy and diseased plants.

4) Model Evaluation

After training, the model is evaluated using key metrics.

Accuracy measures prediction correctness, while precision and recall assess how accurately diseases are identified. The F1-score balances both for a reliable evaluation. A confusion matrix highlights correct and incorrect classifications, aiding in pinpointing areas for improvement. Cross-validation ensures consistent performance on different datasets. Error analysis of misclassified images helps identify the causes of mistakes, guiding model refinement.

5) Model Exportation and Integration with Web App

Model evaluation assesses the performance of a trained machine learning model to ensure its accuracy in diagnosing plant diseases. It involves testing on unseen data using metrics like accuracy, precision, recall, and F1-score. A confusion matrix highlights misclassifications, while cross-validation methods, such as k-fold, verify robustness across data splits. Poor performance may require hyperparameter tuning, more training data, or alternative model architectures. A well- evaluated model ensures reliable predictions, aiding farmers in making informed decisions. Continuous monitoring and retraining with updated data further improve accuracy, making the model a dependable tool for plant disease detection.

B. System Architecture

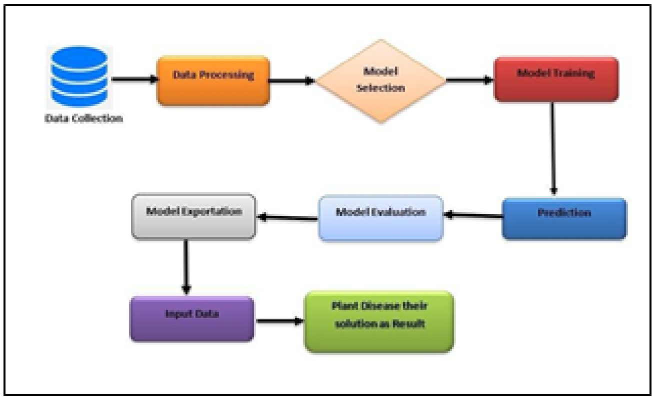


Fig. 1. System Architecture

The system architecture diagram represents a complete machine learning pipeline for developing a Plant Disease Detection System, starting from data collection to prediction and result generation. It clearly outlines the steps followed in building a model capable of identifying plant diseases and suggesting suitable remedies.

1) Data Collection

- Initial phase where a large dataset of plant leaf images is collected.
- The data may include various categories such as healthy leaves and leaves affected by different diseases.

2) Data Processing

Preprocessing techniques are applied to clean and prepare the data.

- Image resizing
- Normalization
- Noise reduction
- Augmentation for better model performance

3) Model Section

- The most suitable Machine Learning or Deep Learning model is chosen.
- Models could include Convolutional Neural Networks (CNN), SVM, or hybrid models.

4) Model Training

- The chosen model is trained on the preprocessed data.
- During this phase, the model learns to distinguish among different disease types by analyzing image features.

5) Prediction

- After training, the model is applied to make predictions on new or unseen data.
- It assigns the input image to a specific disease category or marks it as healthy.

6) Model Evaluation

The model's performance is tested using metrics like

- Accuracy
- Precision
- Recall
- F1 Score

7) Model Exportation

- The final trained and evaluated model is saved/exported for deployment.
- The model becomes ready for real-time application use.

8) Input Data (User Interface)

- End-users provide input (new plant leaf image) via an application or interface.
- The exported model is used to analyze the image and return results.

9) Plant Disease and Their Solution as Result

- Detected disease name.
- Suggestions or remedies for treatment.
- Confirmation if the plant is healthy.

C. Equations

The Figure 2 shows the Use Case diagram, represents a use case or functional flow of an Plant Leaf Disease Prediction System using Machine Learning (ML). It illustrates how different users(User, Admin) and the ML Model interact within the system to perform various operations.

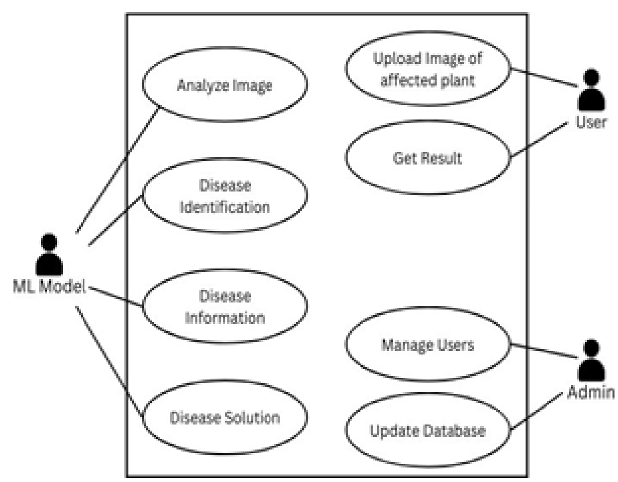


Fig. 2. Use Case Diagram

1) ML Model Functions

Analyze Image: The ML model processes the uploaded leaf image using Deep Convolutional Neural Networks (CNNs) to extract features and identify patterns, determining the specific plant disease.

Disease Information: Provides detailed insights into the detected disease, including symptoms, causes, and its impact on the plant.

Disease Solution: Recommends treatments such as pesticides, organic remedies, and best agricultural practices for the diagnosed disease.

2) User Functions

Upload Image of Affected Plant: Users (such as farmers or agricultural professionals) upload images of diseased leaves for analysis. Get Result: Users receive the output including disease name, confidence level and treatment recommendations.

3) Admin Functions

Manage Users: Admin can add, remove, or update user access,

roles, or credentials. Update Database: Admin updates the dataset or information base (e.g., adding new disease types, solutions, or training data for the ML model).

D. Data Flow Diagram

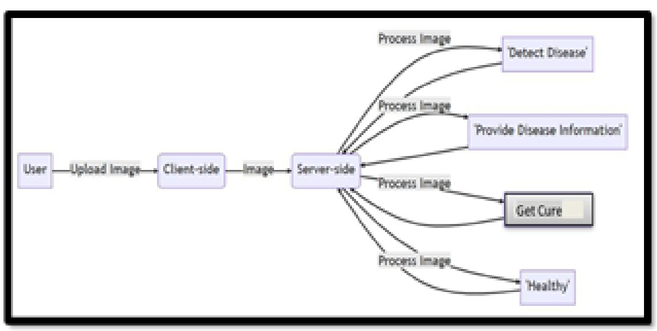


Fig. 3. Data Flow Diagram

The figure 3 shows the Data Flow Diagram, illustrates the detailed operational flow of a Plant Disease Detection System, highlighting how an image uploaded by a user is processed through different stages to determine the health status of a plant and provide appropriate disease information or cure.

1) User

The starting point of the system.

Users (e.g., farmers, agriculture experts) upload an image of a plant leaf through the system interface.

2) Client-Side

This is the frontend interface that takes the input (uploaded image) from the user. It serves as a bridge to send the data (image) to the server side.

3) Server-Side

The backend processing unit responsible for handling the core logic and running the Machine Learning model.

The uploaded image is processed through multiple stages, including:

- Image analysis
- Feature extraction
- Disease prediction/classification

4) Processing Outcomes

The server-side processing may result in one of the following outcomes:

- Detect Disease: If a disease is identified, the system labels it and sends a response.
- Provide Disease Information: The system retrieves and displays detailed data about the detected disease (e.g., symptoms, causes, spread, impact).
- Get Cure: Based on the diagnosis, the system recommends possible treatments or remedies (chemical/organic solutions).
- Healthy: If no disease is detected, the system confirms that the plant is healthy.

III. IMPLEMENTATION

A. Convolutional Neural Network Algorithm

Convolutional Neural Networks (CNNs) form the core of the plant disease detection system, automatically extracting key features from images. In this framework, CNNs effectively recognize patterns and textures linked to various plant diseases, enabling accurate classification. Their ability to learn layered visual features makes them ideal for image-based tasks like disease detection.

The proposed system, implemented using a standard plant leaf image dataset, demonstrated high accuracy and reliability in identifying plant diseases. Training and validation accuracy and loss curves showed consistent improvement without overfitting, aided by techniques like data augmentation and dropout.

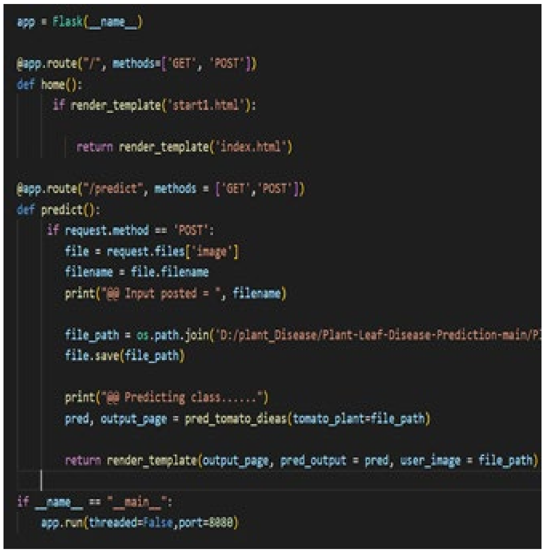
The results confirm that CNNs are highly effective for plant disease detection, successfully learning complex features such as color changes, texture variations, and damage patterns—tasks that are challenging for traditional machine learning methods.

B. Implementation Process

- Data collection: This involves collecting a large dataset healthy and diseased plants. The dataset is used such that it contains labelled classes.
- Data Transformation: This involves transforming the data into desired set of size and features.
- Model training: A machine learning model i.e. vgg16 is used which consists of 16 convolution layers as well as pooling and batch normalization layers.
- Model evaluation: The trained model will be evaluated on a test set of images to determine its accuracy and effectiveness in detecting a particular disease.

- Integration: The model will be integrated with frontend of the web application to give the desired output
- User interface: The user interface will be provided to users to upload images of the affected plant.

Overall, implementing this model requires a thorough understanding of machine learning, web development and cloud computing, as well as expertise in the specific technologies used.



```
app = Flask(__name__)

@app.route("/", methods=['GET', 'POST'])
def home():
        if render_template('start1.html'):

            return render_template('index.html')

@app.route("/predict", methods = ['GET','POST'])
def predict():
      if request.method == 'POST':
        file = request.files['image']
        filename = file.filename
        print("@@ Input posted = ", filename)

        file_path = os.path.join('D:/plant_Disease/Plant-Leaf-Disease-Prediction-main/P

        file.save(file_path)

        print("@@ Predicting class......")
        pred, output_page = pred_tomato_dieas(tomato_plant=file_path)

        return render_template(output_page, pred_output = pred, user_image = file_path)

if __name__ == "__main__":
    app.run(threaded=False,port=8080)
```

Fig. 4. Plant Disease Prediction System

C. Dataset

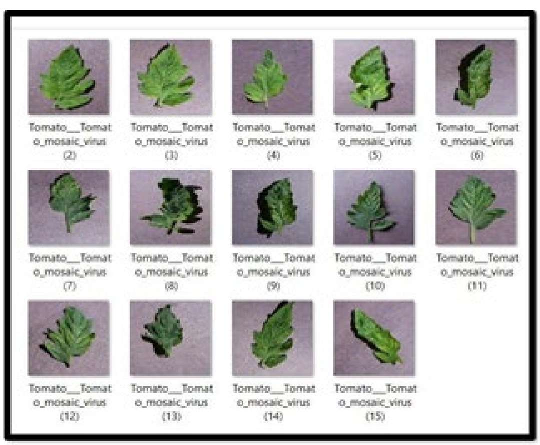


Fig. 5. Dataset

The Tomato Plant Healthy and Diseased Leaf Dataset is a collection of images used to train and evaluate machine learning models for automatic tomato plant disease detection. This dataset includes images of both healthy and diseased tomato leaves, enabling the model to distinguish between normal and infected plants.

1) Data Acquisition

The dataset is sourced from agricultural research institutes, farms and online datasets like Kaggle, PlantVillage. Images are captured under different lighting conditions using smartphones or other any devices. Each image is labeled according to whether it represents a healthy or diseased tomato leaf.

2. Dataset Composition

The dataset consists of two main categories: Healthy Leaves –

Images of normal, disease-free tomato leaves. Diseased Leaves

– Includes various infections such as:

- Tomato Bacterial Spot (small black lesions)
- Tomato Early Blight (brown spots with rings)
- Tomato Late Blight (dark, water-soaked lesions)
- Tomato Leaf Mold (yellow spots and fuzzy growth)
- Tomato Yellow Leaf Curl Virus (TYLCV) (leaf curling and yellowing)

2) Data Preprocessing

To improve model performance, the dataset undergoes:

- Resizing – Standardizing image dimensions.
- Normalization – Scaling pixel values between 0 and 1.
- Augmentation – Techniques like rotation, flipping and contrast adjustments increase dataset variability.

3) Data Splitting

- Training Set (80%) – Used to teach the model patterns of disease.
- Validation Set (10%) – Used to fine-tune model hyperparameters.• Testing Set (10%) – Evaluates final model accuracy on unseen data.

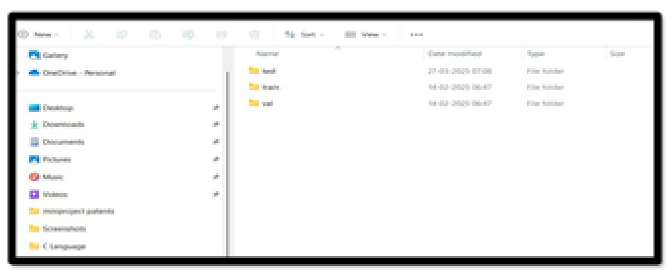


Fig. 6. Splitting Dataset

4) Usage in Plant Disease Prediction System

The dataset is employed to train Convolutional Neural

Networks (CNNs) for image classification.

The trained model detects tomato plant diseases in real time when a new leaf image is input. It helps farmers and researchers diagnose diseases early, reducing crop losses and improving productivity.

- Prediction Button

Below the file upload section, there is a “Predict” button (styled in cyan blue) which, when clicked, initiates the disease prediction process using a pre-trained machine learning model.

IV. RESULT AND DISCUSSION

A. User Interface

The user interface (UI) is essential in enabling effective interactions between users and the interactive exploration and summarization system. The design of the UI was focused on providing an intuitive and efficient experience for plant disease prediction system.

Fig. 7. Home Page of Plant Disease System

The above user interface illustrates the home page of a Plant Disease Prediction System, designed to provide a simple, visually appealing and user-friendly interface for identifying plant leaf diseases and suggesting possible cures.

It clearly communicates the primary goal of the system — to detect plant leaf diseases and provide treatment suggestions.

Key Elements

- File Upload Section

A "Choose File" button allows users to upload an image of the affected plant leaf from their device. Next to it is a text field that displays the name of the selected file.

Fig. 8. Predict Plant Leaf Disease & Get Treatment

The Plant Disease Prediction System Output Page the above showcases the result or output page of the Plant Disease Prediction System, where the system displays the predicted disease after analyzing the uploaded plant leaf image.

Key Elements

- Uploaded Leaf Image Display

At the center-left of the page, the uploaded plant leaf image a tomato leaf is shown inside a bordered box. This helps users visually confirm the image they submitted for prediction.

- Predicted Disease Information

On the right side of the image, the system displays the disease detected from the uploaded leaf. In this case, the prediction reads: "Tomato – Bacteria Spot Disease", clearly indicating the specific condition affecting the plant.

- Treatment Details

Above the disease name, there is a detailed treatment section providing actionable advice to the user. The treatment recommendation suggests using copper fungicides, explaining how and when they should be applied to effectively manage the bacterial spot disease on tomato plants. It also includes tips on preventive measures and frequency of treatment, depending on weather conditions.2. Prediction of Healthy Leaf

Fig. 9. Prediction of Healthy Leaf

Prediction of Healthy Leaf Detection Output Page The above image represents another output page of the Plant Disease Prediction System, specifically showcasing the scenario where the system identifies that the uploaded plant leaf is healthy and disease-free.

- Uploaded Leaf Image Display

On the left side of the page, there is a clear image of the uploaded tomato leaf, placed within a bordered box. The leaf appears green, fresh and free from any visible symptoms of disease, aligning with the system's prediction.

- Prediction Result

On the right side, the result is displayed as: "Tomato – Healthy and Fresh" in a black-highlighted label. This clearly informs the user that no disease has been detected in the leaf.

- Additional Message

A short and reassuring message is presented below the prediction: "There is no disease on the Tomato leaf." This provides confirmation to the user that the plant is in good health and does not require treatment.

V. CONCLUSION

The Plant Disease Prediction System uses machine learning to provide an effective solution for early detection of leaf diseases. By combining image processing with intelligent algorithms, it enables quick and accurate identification, supporting timely treatment and reducing crop losses. Its user-friendly interface makes it accessible to users of all technical backgrounds, including those in rural areas. With future integration of real- time data and mobile features, the system can significantly advance precision agriculture. Ongoing feedback from disease outcomes helps refine the model, enhancing performance and scalability for sustainable farming.

REFERENCES

[1] S. D. Khirade and A. B. Patil, (2021). "Plant disease detection using image processing", Proceedings of the International Conference on Computing Communication Control and Automation.

[2] S. C. Madiwalar and M. V. Wyawahare, (2017). "Plant disease identification: A comparative study", International Conference on Data Management, Analytics and Innovation (ICDMAI).

[3] S. D.M., Akhilesh, S. A. R. M.G. and P. C. (2019). "Image based Plant Disease Detection in Pomegranate Plant for Bacterial Blight", International Conference on Communication and Signal Processing (ICCSP).

[4] P. Moghadam, D. Ward, E. Goan, S. Jayawardena, (2017). "Plant Disease Detection Using Hyperspectral Imaging", International Conference on Digital Image Computing: Techniques and Applications (DICTA).

[5] G. Shrestha, Deepsikha, M. Das and N. Dey (2020). "Plant Disease Detection Using CNN", IEEE Applied Signal Processing Conference (ASPCON).

[6] S.P. Mohanty, D.P. Hughes and M. Salathé, (2016). "Using deep learning for image- based plant disease detection", Front. Plant Sci., vol.7.

[7] X. Yang and T. Guo, (2018). "Machine learning in plant disease research", Eur. J. BioMed. Res., vol. 3, no. 1, p. 6, http://dx.doi.org/10.3389/fpls.2016.01419 http://dx.doi.org/10.18088/ejbmr.3.1.2017, pp. 6-9.

[8] Godliver Owomugisha, John A. Quinn, Ernest Mwebaze (2020). "Automated Vision- Based Diagnosis of Banana Bacterial Wilt Disease and Black Sigatoka Disease", Preceding the 1'st international conference on the use of mobile ICT in Africa.

[9] G. Shrestha, Deepsikha, M. Das and N. Dey, (2020). "Plant Disease Detection Using CNN", IEEE Applied Signal Processing Conference (ASPCON).

[10] S. D.M., Akhilesh, S. A. Kumar, R. M.G. and P. C. (2019). "Image based Plant Disease Detection in Pomegranate Plant for Bacterial Blight," International Conference on Communication and Signal Processing (ICCSP).

[11] S.-N. Ren, Y. Sun, H.-Y. Zhang and L.-X. Guo (2019). "Plant disease identification for small sample based on one-shot learning", Jiangsu J. Agricult. Sci., vol. 35, no. 5, pp. 1061–1067.