INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT (IJSREM)
VOLUME: 09 ISSUE: 05 | MAY - 2025 SJIF RATING: 8.586 ISSN: 2582-3930

Tomato Leaf Disease Prediction Using Deep Learning Techniques

M Charishma
Student
meduru.charishma@gmail.com

Beldona Anupama
Student
banupama2003@gmail.com

Shaik Yasir Ahamed
Student
yasiryassu727@gmail.com

Kurakula Dinesh
Student
kurakuladinesh963@gmail.com

Ms Swetha Rajagopal
Assistant Professor, SOCSE
swetha.rajagopal@presidencyuniversity.in

Abstract: Tomato cultivation is vital in India, but leaf diseases can reduce yields by 20–40%. Manual detection is time-consuming, labor-intensive, and costly. To address this, a synthesized dataset from agricultural fields was used to train three CNN models for early disease detection. A General CNN achieved 81% accuracy, MobileNet V2 chosen for its lightweight design and real-time suitability achieved 85.2%, while VGG Net reached 77%. Source code is available at https://github.com/Yasir727/Tomatoo-Leaf-Disease-Prediction.

Index Terms - Convolutional Neural Network (CNN), Early Disease Detection, MobileNet V2, Precision Agriculture, Tomato Leaf Disease

INTRODUCTION

India is the sec second-largest tomato producer. Contributing 11% to global output. Tomato leaf diseases significantly impact yield and farmer income. For example, Septoria leaf spot can cause up to 50% yield loss, and Tomato Yellow Leaf Curl Virus can lead to 90–100% crop loss. Tomato leaf diseases like Early and Late Blight, Septoria Leaf Spot, TYLCV, Powdery Mildew, and Bacterial Spot severely affect plant health. They cause symptoms like leaf spots, curling, yellowing, and defoliation, leading to reduced yield, fruit quality, and farmer income.

FIGURE 1. HEALTHY, EARLY BLIGHT, BACTERIAL SPOT, LEAF MOLD, TOMATO YELLOW LEAF CURL VIRUS (TYLCV), LATE BLIGHT

Overall, leaf diseases are a major factor impacting tomato crop yields, with potential losses varying widely depending on the specific pathogen and environmental conditions.

Traditional methods for identifying tomato leaf diseases, such as manual inspection and laboratory testing, are time-consuming, labor-intensive, and often costly. These methods lack scalability and can result in delayed disease detection, allowing infections to spread unchecked. Although experienced farmers can visually identify diseases, this approach is inefficient for large-scale operations. Furthermore, excessive use of fungicides can harm the environment and promote pathogen resistance.

This project uses AI and Machine Learning, specifically CNNs, to detect and classify tomato leaf diseases. Models like General CNN, MobileNetV2, and VGGNet were tested for accuracy and speed. Trained on leaf images, the system is integrated into a web app, enabling farmers to upload photos for instant diagnosis and guidance—supporting real-time, data-driven farming and reducing crop losses.

LITERATURE SURVEY

Deep learning techniques have significantly advanced the field of plant disease detection, allowing researchers to classify and identify diseases based on leaf images. Various studies have explored different convolutional neural network (CNN) architectures and datasets to improve classification accuracy. One study, Analysis on Prediction of Plant Leaf Diseases using Deep Learning [1], analyzed the prediction of plant leaf diseases using the Plant Village dataset, applying CNN with multiple layers and leveraging Keras and TensorFlow for training. However, this study lacked real-world applicability, as it did not account for large-scale farming conditions. Another research effort, Automatic Prediction of Plant Leaf Disease Using Deep Learning Models [2], used deep learning models, including pre-trained networks and transfer learning, to enhance disease identification. The inclusion of hardware like the Jetson Nano Kit allowed for accelerated computations, but the study failed to integrate cost-effective real-time detection tools, limiting its usability in low-resource agricultural settings.

Several approaches have focused on improving the efficiency and robustness of plant disease classification. The study Plant Disease Prediction and Classification using Deep Learning [3] implemented CNN-based image processing and normalization techniques to classify plant diseases, yet it did not address the challenge of model interpretability and real-time scalability. Another research paper, Plant Disease Prediction using Deep Learning and IoT [4], explored the integration of IoT and deep learning for plant disease detection. By incorporating sensors and drones to analyze environmental factors alongside leaf images, the model aimed to provide a more holistic analysis of plant health. However, this approach introduced practical challenges such as sensor corrosion and hardware maintenance, making long-term implementation difficult in open agricultural fields.

Deep learning models like ResNet50, InceptionV3, and ResNet152V2 have been explored for detecting plant diseases, achieving high accuracy in leaf image classification. The study Plant Disease Detection and Management using Deep Learning Approach [5] focused on cotton and other crops, demonstrating the effectiveness of these models but overlooking key factors such as soil conditions and environmental stress, which are crucial for a more comprehensive disease detection system.

Previous studies in plant disease detection using deep learning face several limitations. Many models were trained on the Plant Village dataset, which lacks real-world variability, leading to reduced effectiveness in diverse field conditions. Computational complexity and reliance on high-end hardware limited real-time applicability, especially in low-resource settings. Furthermore, most models overlooked environmental factors such as soil quality, weather, and pest interactions. To address these gaps, we propose a lightweight and efficient MobileNet-based model trained on a custom dataset collected from varied geographic regions. This approach enhances model adaptability, accuracy, and suitability for real-world deployment.

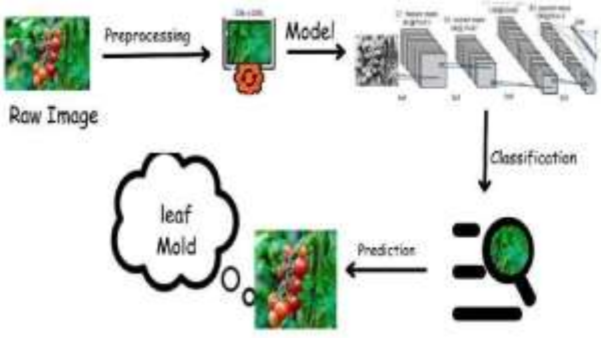
DESIGNING

Designing in software development involves planning the system's structure, UI, data flow, and technology to ensure functionality, scalability, and a smooth user experience.

Architecture

The system architecture for plant leaf disease detection is designed to ensure accurate and efficient classification of plant diseases based on image processing and deep learning techniques. The process begins with capturing or uploading a raw image of a plant leaf. This image serves as the primary input for the system, which undergoes several stages before reaching the final classification.

FIGURE 2. ARCHITECTURE DIAGRAM



Preprocessing

Preprocessing is essential for optimizing input images for accurate classification. Tomato leaf images, acquired via camera or dataset, are first resized to a consistent dimension (e.g., 256×256 pixels). Image enhancement techniques such as histogram equalization, Gaussian blur for noise reduction, and contrast adjustment are applied to improve quality. To enhance model generalization, data augmentation methods like rotation, flipping, and brightness variation are employed. The preprocessed image is then passed to the deep learning model for feature extraction.

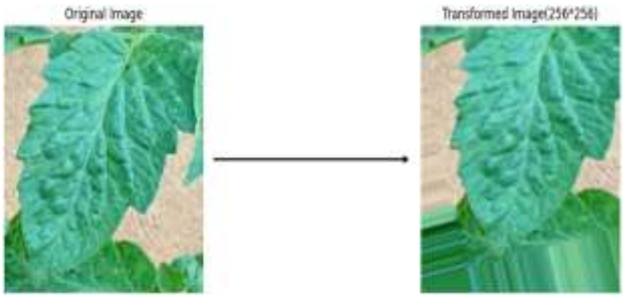


FIGURE 3. IMAGE PREPROCESSING

Model Processing

After preprocessing, the image is passed through a CNN with convolutional layers for feature detection, pooling for dimensionality reduction, ReLU for non-linearity, and dropout to prevent overfitting—enabling accurate disease classification.

Classification

After feature extraction, the processed image is passed to the the classification layer, which typically consists of fully connected layers followed by a SoftMax or sigmoid activation function. The model predicts the likelihood of the leaf belonging to various disease categories, such as "Leaf Mold," based on learned patterns. Classification accuracy is influenced by the quality of training data, model architecture, and feature extraction depth.

Model Processing

After preprocessing, the image is fed into a CNN that extracts disease-related features using convolutional and pooling layers. ReLU adds non-linearity, and dropout prevents overfitting, enabling accurate classification.

Classification

After feature extraction, the processed image is passed to the classification layer, which typically consists of fully connected layers followed by a SoftMax or sigmoid activation function. The model predicts the likelihood of the leaf belonging to various disease categories, such as "Leaf Mold," based on learned patterns. Classification accuracy is influenced by the quality of training data, model architecture, and feature extraction depth.

Prediction

Upon classification, the model generates a final prediction, labeling the input image and providing a confidence score to indicate the certainty of the classification (e.g., "Leaf Mold: 92% confidence"). This output is presented visually to the user, often with an annotated image highlighting the affected areas. If integrated into an application, the model may also offer recommendations for disease treatment and preventive measures.

Decision and Action

The final step involves taking action based on the model's prediction. If a disease is detected, farmers or experts can apply control measures such as fungicides, adjusting irrigation, or isolating affected plants to prevent further spread. In precision agriculture, AI-driven predictions can be integrated into automated systems to trigger real-time interventions, such as pesticide spraying or sending alerts to farmers. This approach aims to minimize crop damage and enhance agricultural productivity through timely, accurate disease detection.

IMPLEMENTATION

Dataset Collection

Dataset collection is crucial for model accuracy, involving the use of the Tomato Village dataset from Kaggle for labeled data and a region-specific dataset from Agriculture Field for real-time insights. This ensures robust, unbiased models with improved performance.

Dataset – 1

The Tomato Village dataset from Kaggle provides labeled images of tomato leaves affected by various diseases, including Early Blight, Late Blight, Septoria Leaf Spot, Bacterial Spot, and Leaf Mold. This dataset is used to train a model for efficient disease detection, helping improve early identification and crop management.

TABLE 1. TOMATO VILLAGE DATASET USED FOR TRAINING

Disease name	Number of training images	Number of test images	Total
Healthy	424	240	664
Bacterial Spot	386	175	561
Early Blight	431	240	671
Yellow Leaf Curl	392	145	537
Leaf mold	405	145	573

Dataset-2

For this project, we collected our own dataset from Agricultural filed, ensuring that the data was specifically tailored to the plant diseases we aimed to classify. The dataset consists of images categorized into four different classes, including one class for healthy leaves and three classes representing different plant diseases. Each image in the dataset was captured in high resolution under controlled conditions to maintain consistency in lighting and background, which helps in improving model accuracy.

TABLE 2. DATASET COLLECTED AND USED FOR TRAINING

Disease name	Number of training images	Number of test images	Total
Healthy	400	100	500
Black_Spot	400	100	500
Leaf_miner_files	400	100	500
Yellow_Leaf_Curl	400	100	500

Data Preprocessing

Before training the deep learning models, the collected dataset was preprocessed to enhance the accuracy of predictions. The preprocessing steps included resizing images to 256x256 pixels, normalization. These steps helped improve model generalization and performance.

Model Training

We trained three different models—MobileNetV2, CNN, and VGGNet—to classify plant diseases. Each model was trained on the same dataset, and their performance was evaluated based on accuracy and loss metrics. Below are the details of each model:

Convolutional Neural Network (CNN)

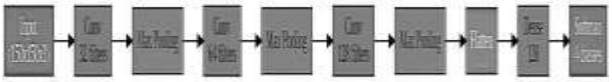


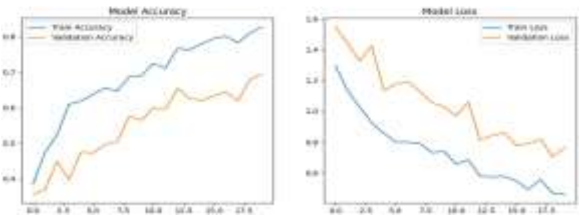
FIGURE 4. CNN ARCHITECTURE

We used a CNN architecture with an input size of 150×150×3, comprising three convolutional layers (32, 64, 128 filters) with max pooling, followed by a Dense layer with 128 neurons and a Softmax output layer for 4-class classification. This design effectively captures spatial features while maintaining computational efficiency.

VGGNet

FIGURE 5. VGGNET ARCHITECTURE

We employed a transfer learning-based CNN using VGG16 (without top layers) as a feature extractor for 150×150×3 RGB images. Extracted features are flattened and passed through a Dense layer (256 units), followed by a 0.5 Dropout and a Softmax layer for 4-class classification. This approach enhances accuracy and reduces training time by leveraging pre-trained knowledge.



MobileNetV2

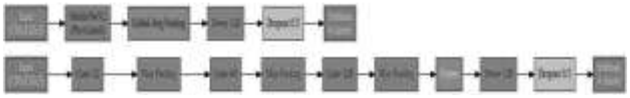


FIGURE 6. MobileNetV2 Architecture

We compare two CNN architectures for classifying 150×150×3 images. The first uses MobileNetV2 with Global Average Pooling, a Dense layer (128 units), Dropout (0.5), and a Softmax layer (4 classes), leveraging transfer learning for better generalization. The second is a custom CNN with three convolutional layers (32, 64, 128 filters), max pooling, a Dense layer (128 units), Dropout (0.5), and Softmax, offering flexibility for tailored applications.

Deployment using Flask

To make the model accessible to users, we developed a Flask web application where users can upload an image and receive a disease prediction along with information about the cause and recommended pesticides.

User Interface for Disease Prediction

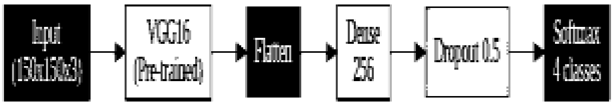
The Flask web application features a simple and user-friendly interface where users can upload an image of a plant leaf. Once uploaded, the system processes the image and returns the predicted disease name, possible cause, and recommended pesticide.

Key Features of the Web Application

- Image Upload Functionality
- Real-time Disease Classification
- Display of Disease Cause and Recommended Pesticide

The front-end was built with HTML, CSS, and Bootstrap for responsiveness. Using MobileNetV2, we developed a deep learning-based system integrated into a Flask web app, allowing users to upload leaf images and get disease diagnoses with preventive tips—helping farmers take timely action.

Experimental Results



This section presents the evaluation of the implemented models using accuracy graphs and the final output of the developed web application.

Model Performance Evaluation

To assess the performance of the models, accuracy graphs were generated for three different approaches. These graphs provide insights into the learning behavior and final accuracy achieved by each model.

CNN model

FIGURE 7. CNN MODEL ACCURACY AND LOSS GRAPH

The two plots show training accuracy and loss trends for a machine learning model. The accuracy plot reveals rising training accuracy (>0.8) and lower validation accuracy (~0.7), indicating overfitting. The loss plot shows steadily decreasing training loss, while validation loss fluctuates and remains higher, further suggesting the model is not generalizing well to unseen data.

VGGNET

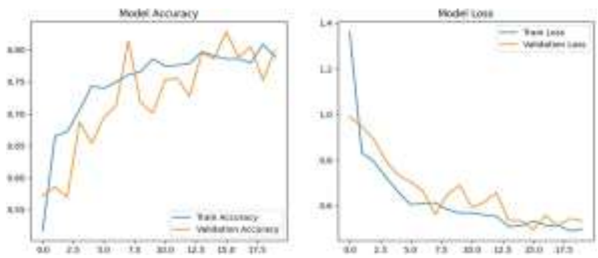


FIGURE 8. VGGNET ACCURACY AND LOSS GRAPH

The two plots display training progress of a machine learning model. Accuracy curves (left) show both training and validation accuracy increasing and closely aligned, indicating good generalization. The loss curves (right) similarly decrease together, with minor fluctuations in validation loss. Overall, the model shows effective learning with minimal overfitting, suggesting it generalizes well to unseen data.

Mobile net

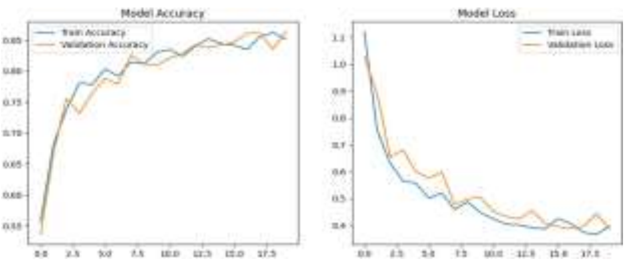


FIGURE 9. MOBILENETV2 ACCURACY AND LOSS GRAPH

The image presents training accuracy and loss plots for a machine learning model. The accuracy plot shows both training and validation accuracy steadily increasing to around 0.85, indicating strong generalization. The loss plot shows both training and validation losses decreasing consistently with only minor fluctuations, suggesting effective learning and minimal overfitting. Overall, the model demonstrates well-balanced performance on both training and unseen data.

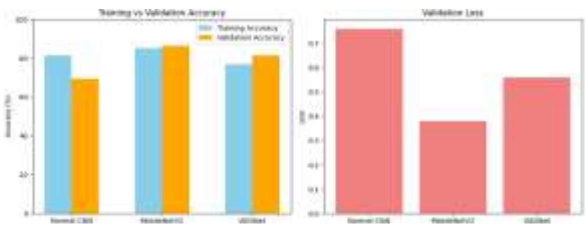


FIGURE 10. COMPARISON GRAPH BETWEEN THREE MODELS

The given image consists of two bar charts comparing the performance of three different deep learning architectures: Normal CNN, MobileNetV2, and VGGNet. The left plot presents a comparison of training vs. validation accuracy, while the right plot displays validation loss for each model. To evaluate the trained models, we analyzed accuracy, loss, precision, recall, and F1-score. MobileNetV2 achieved the highest performance, as shown in the table below:

Table 3. Model Evaluation table

The comparison shows that MobileNetV2 outperforms Normal CNN and VGGNet, with the highest accuracy and lowest validation loss. Normal CNN shows overfitting, while VGGNet performs moderately well. MobileNetV2 offers the best generalization and overall performance.

Web Application Output

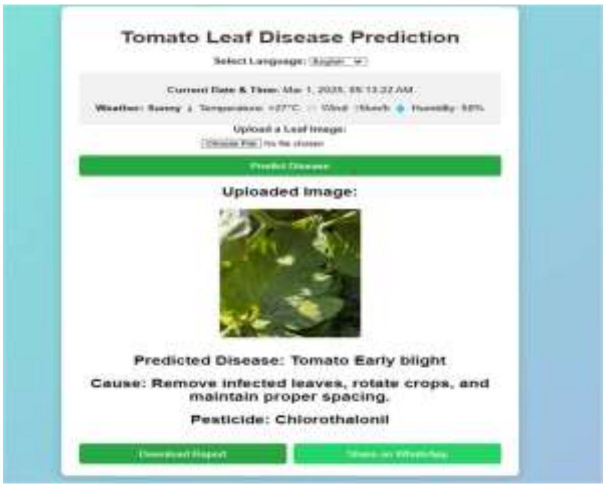


FIGURE 11. PREDICTION PAGE

Our tomato leaf disease prediction system sounds like a valuable tool for farmers and gardeners. By leveraging image processing and deep learning, it offers early detection of plant diseases, helping users identify issues like Early Blight, Late Blight, and Septoria Leaf Spot. The system not only diagnoses the diseases but also provides actionable insights on causes, effects, treatments, and preventive measures, including recommendations for pesticides. Additionally, with features like report downloads and WhatsApp sharing, it ensures accessibility and ease of use. Ultimately, this project supports farmers in minimizing crop losses, improving plant health, and promoting sustainable farming practices. Great initiative to use technology for improving agriculture.

CONCLUSION

This project implemented a deep learning-based image analysis system using CNN, MobileNet, and VGGNet models. MobileNet showed the highest accuracy with efficient computation, making it the best choice for deployment. The system allows users to upload tomato leaf images and receive instant predictions using MobileNet, providing real-time disease detection. A web application was developed for easy access, making AI-powered image classification user-friendly and accessible. The project demonstrates the practical benefits of lightweight AI models in real-world applications, combining deep learning with web technologies for efficient and accessible plant disease detection.

Model	Accuracy(%)	**Loss(%)**	**Precision(%)**	**Recall(%)**	**F1-score(%)**
CNN	81	45	72	69	67
VGG Net	77	52	83	80	79
MobileNetV2	85.2	40	88	86	86

FUTURE WORK

Future enhancements to our deep learning-based image analysis system include integrating advanced models like Efficient Net or ViTs for improved accuracy, fine-tuning MobileNet with diverse data for better generalization, and optimizing the web app using TensorFlow.js or Web Assembly. Cloud deployment (AWS Lambda or Google Cloud AI) will boost scalability and response time. Usability will be enhanced through batch processing, real-time predictions, API support, and Grad-CAM visualizations. A mobile app with offline capability will support remote use, while user feedback, active learning, multilingual support, and accessibility features will ensure continuous improvement and inclusivity.

REFERENCES

[1] Nandhini, S. and Ashok kumar K., 2021, March. Analysis on prediction of plant leaf diseases using deep learning. In 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS) (pp. 165-169). IEEE.

[2] Lakshmi, M.A. and Gomathi, V., 2021, December. Automatic prediction of plant leaf diseases using deep learning models: a review. In 2021 5th International conference on electrical, electronics, communication, computer technologies and optimization techniques (ICEECCOT) (pp. 569-574). IEEE.

[3] Lakshmanarao, A., Babu, M.R. and Kiran, T.S.R., 2021, September. Plant disease prediction and classification using deep learning ConvNets. In 2021 International Conference on Artificial Intelligence and Machine Vision (AIMV) (pp. 1-6). IEEE.

[4] Gupta, A.K., Gupta, K., Jadhav, J., Deolekar, R.V., Nerurkar, A. and Deshpande, S., 2019, March. Plant disease prediction using deep learning and IoT. In 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 902-907). IEEE.

[5] Kothawade, P., Varma, S. and Gogate, U., 2022, February. Plant disease detection and management using deep learning approach. In 7th International Conference on Computing in Engineering & Technology (ICCET 2022) (Vol. 2022, pp. 30-35). IET.