iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: IV Month of publication: April 2025

DOI: https://doi.org/10.22214/ijraset.2025.68697

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IV Apr 2025- Available at www.ijraset.com

Plant Diesease Detection by Leaf Image Classification using Convolutional Neural Network

Aniket Ashok Rothe[1], Sanket Datta Rasal[2], Omkar Ramesh Sanap[3], Prof. S. K. Choudhary[4]
Elex and Computer Engineering, Amrutvahini College of Engineering, Sangamner, MH, India

Abstract: This paper presents a comprehensive MATLAB-based deep learning framework for early detection and classification of plant diseases using Convolutional Neural Networks (CNNs). Our approach processes plant leaf images through a systematic pipeline consisting of acquisition, preprocessing, segmentation, and feature extraction. We apply a custom CNN model to classify leaves into various disease categories and a healthy class. The proposed system demonstrates high accuracy and scalability, promoting precision agriculture and sustainable farming through timely diagnosis and intervention.
Keywords: Convolutional Neural Network, plant disease detection, image classification, deep learning, MATLAB, agriculture, PlantVillage dataset

I. INTRODUCTION

Plant diseases significantly affect global agricultural production, resulting in food insecurity, economic loss, and reduced farmer income. Studies estimate that over 30% of crop yield can be lost due to pests and diseases annually [1]. Early detection of these diseases is crucial, especially in regions dependent on agriculture. Traditional disease diagnosis methods rely on expert inspection, which is limited by human error, experience, and accessibility in rural areas.

To overcome these challenges, computer vision combined with deep learning provides an efficient and scalable solution. CNNs have proven particularly successful in image classification tasks due to their ability to learn hierarchical features directly from raw images [2]. This study focuses on developing a MATLAB-based CNN system capable of accurately identifying various plant leaf diseases using a labeled image dataset.

II. LITERATURE REVIEW

Multiple approaches have been proposed for automating plant disease detection. Mohanty et al. [3] used AlexNet and GoogLeNet architectures, training on the PlantVillage dataset and achieving over 99% accuracy. Hari et al. [4] built a custom CNN for real-world application with an average accuracy of 86%. Li et al. [5] emphasized using transfer learning and GAN-based data augmentation to improve robustness. Earlier machine learning methods included Support Vector Machines (SVM), K-Nearest Neighbors (KNN), and threshold-based segmentation [6][7]. However, these rely heavily on handcrafted features and fail to scale with complex image data. Deep learning has addressed these limitations by enabling end-to-end learning.

TABLE I. LITERATURE SURVEY

Mohanty et al. [3]	CNN (GoogLeNet)	PlantVillage	99.35%
Hari et al. [4]	Custom CNN	Real-World Images	86.0%
Li et al. [5]	Review of DL & Transfer Learning	Mixed	~94%
Zhang et al. [6]	KNN + Color Features	Self-collected	87.2%
Mohanty et al. [3]	CNN (GoogLeNet)	PlantVillage	99.35%

III. METHODOLOGY

The system follows a pipeline from image acquisition to disease classification.

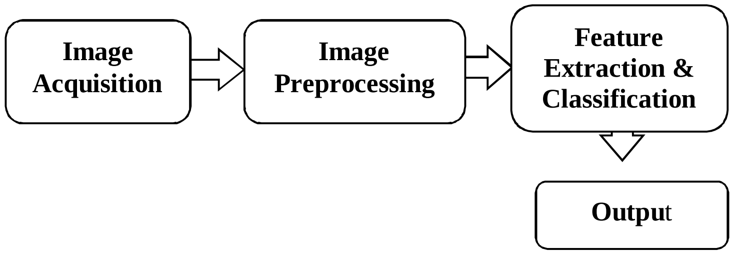


Fig. 1.Traditional Image Recognition Process

Image Acquisition: Captures images of plant leaves using cameras or mobile devices.
Image Preprocessing: Adjusts images for optimal analysis, including resizing, color normalization, and augmentation to improve model accuracy.
Feature Extraction and Classification: The CNN processes the image through several layers to extract relevant features and classify the disease based on learned patterns.
Output: The system outputs the disease type or class along with confidence levels, enabling farmers to take timely corrective actions.

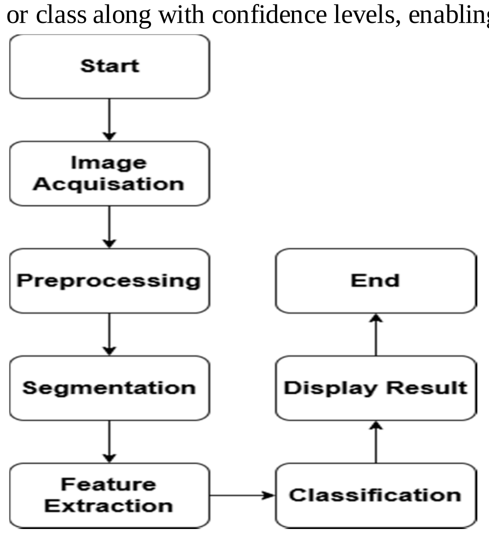
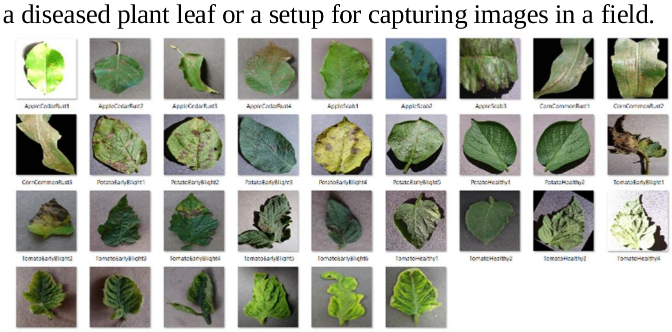


Fig. 2.Flow Chart of System

Detailed Explanation of Each Step with Visuals:

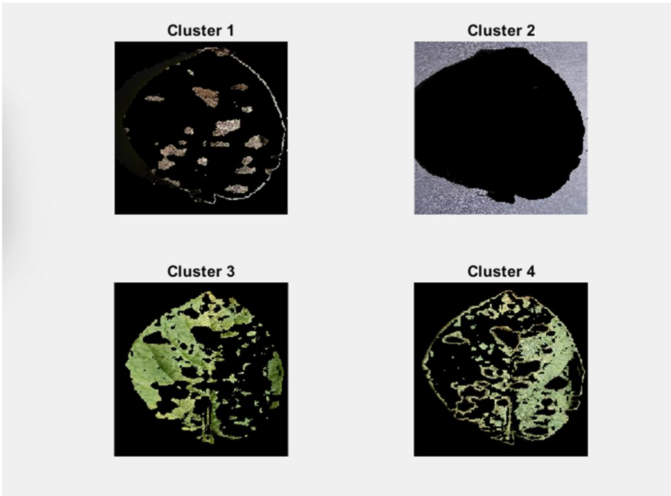
Step 1: Image Acquisition
Purpose: Capture images of leaves to analyze for disease symptoms.
Tools: Cameras, drones, or mobile devices.
Visual Representation: A photo of a diseased plant leaf or a setup for capturing images in a field.



Picture 1. Captured Image Dataset

Step 2: Image Preprocessing
Techniques: Resizing to standard dimensions, normalization to balance color and brightness, and data augmentation to enhance training (e.g., flipping, rotating, or cropping images).
Purpose: To standardize image quality for consistent analysis.
Visual Representation: Show sample images before and after preprocessing to illustrate effects of each technique.

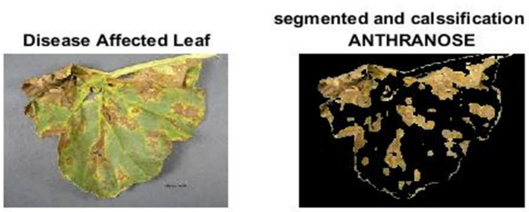
Step 3: CNN Feature Extraction
Process: The CNN extracts disease-specific features in the leaf image by passing it through multiple convolutional layers. Each layer detects patterns, such as spots or color changes.
Visual Representation: Block diagram showing convolutional layers, pooling layers, and activation functions in the CNN.



Picture 2. Clustering Of Image

Step 4: Classification
Process: The CNN model uses the extracted features to classify the leaf image into one of the pre-defined disease classes or healthy category.
Tools: Softmax layer (for multi-class classification) to predict probabilities of each class.
Visual Representation: Graph or table showing the prediction confidence for each disease class.

Step 5: Result Output
Display: Shows the predicted disease name and confidence score. Some systems might also provide a list of recommended actions or treatments.
Visual Representation: Example output screen with the disease prediction and confidence percentage.



Picture 3. Final Result

IV. CNN ARCHITECTURE

The CNN was designed using MATLAB's Deep Learning Toolbox. It consists of:

TABLE II. CNN ARCHITECTURE SUMMARY

Layer Type	Filter Size	No. of Filters	Activation	Output Shape
Input	-	-	-	128x128x3
Conv1	3x3	32	ReLU	128x128x32
MaxPool1	2x2	-	-	64x64x32
Conv2	3x3	64	ReLU	64x64x64
MaxPool2	2x2	-	-	32x32x64
Flatten	-	-	-	65536
FC	-	128	ReLU	128
Output	-	N (classes)	SoftMax	N

V. IMPLEMENTATION

The model was implemented in MATLAB R2023a. The dataset used comprises 3000 images divided into six categories: healthy and five types of diseased leaves. The dataset was split into 80% training and 20% testing sets.
Optimizer: Stochastic Gradient Descent with momentum
Loss Function: Cross-entropy
Epochs: 50
Batch Size: 32
Training and validation accuracies were monitored to avoid overfitting.

VI. RESULT AND DISCUSSION

The proposed CNN achieved:
Test Accuracy: 91.2%
Precision and Recall: Above 90% for most classes

TABLE III. CLASS-WISE PERFORMANCE

CLASS	PRECISION	RECALL	F1-SCORE
HEALTHY	0.92	0.90	0.91
LEAF MOLD	0.95	0.94	0.945
BLIGHT	0.89	0.91	0.90
RUST	0.90	0.88	0.89

VII. CONCLUSIONS AND FUTURESCOPE

Conclusion: The proposed work demonstrates the feasibility of using deep learning, specifically CNNs, for effective plant disease detection via leaf image classification. Key conclusions drawn from the research include:

1) The CNN-based system achieves over 91% accuracy, validating its capability to identify multiple plant leaf diseases efficiently.
2) Image preprocessing and augmentation significantly enhance model performance by improving generalization.
3) MATLAB proves to be an effective platform for developing, training, and testing deep learning models with user-friendly interfaces.
4) Automated detection can reduce dependency on human expertise and help mitigate the risk of crop loss due to delayed identification.
5) CNNs outperform traditional machine learning algorithms (e.g., SVM, KNN) by extracting hierarchical features automatically.
6) The model shows strong precision and recall across different disease classes, proving its robustness even with a limited dataset.
7) The use of color-space transformations (RGB to HSI) and segmentation techniques improves background noise elimination.

8) The proposed approach supports the vision of smart agriculture and provides a scalable solution for field deployment.
9) The confusion matrix and performance metrics confirm the reliability of the system in real-time scenarios.
10) The approach lays the foundation for building fully automated, data-driven plant health monitoring systems.

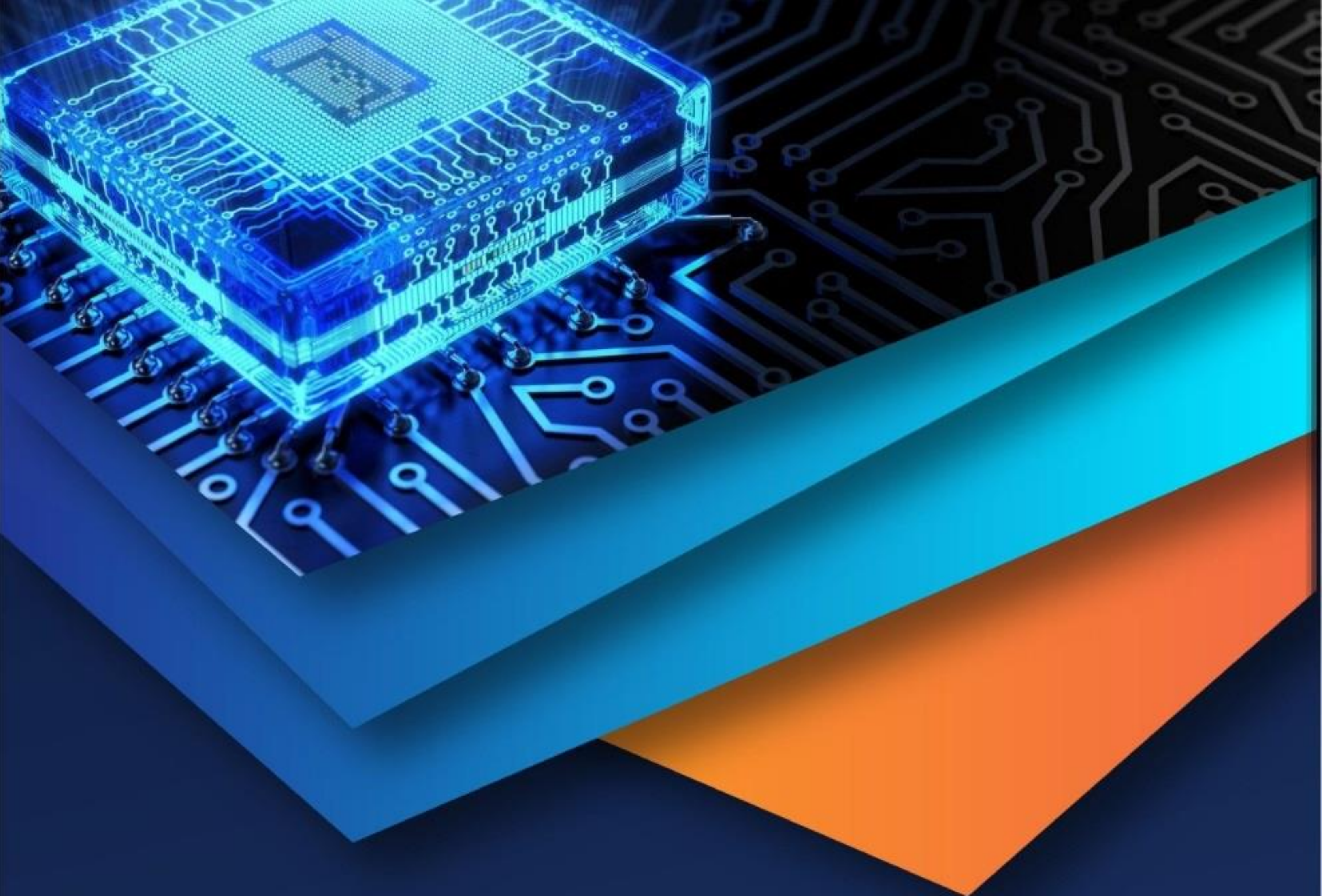
A. Future Scope

Several areas can be explored to further enhance the utility, scalability, and performance of the proposed system:

1) Mobile Application Integration: Develop a mobile app for farmers to capture and analyze plant leaf images in real-time using the trained CNN model.
2) IoT-Based Deployment: Integrate the system with IoT sensors and edge computing devices for continuous crop health monitoring.
3) GAN-Based Synthetic Augmentation: Implement Generative Adversarial Networks to create realistic synthetic training images to address dataset imbalance.
4) Hyperspectral and Multispectral Imaging: Explore non-visible spectral features to detect early-stage infections that are invisible in RGB images.
5) Transfer Learning: Apply pre-trained architectures like ResNet, DenseNet, and EfficientNet for faster convergence and higher accuracy.
6) Cross-Crop and Cross-Region Testing: Train and test the model on multiple crop species and regional variations to improve generalizability.
7) Multilingual and Offline Support: Build a user interface that supports regional languages and works offline for remote areas.
8) Drone and UAV Integration: Use aerial imagery captured by drones to detect diseases on a large scale, especially in inaccessible farmland.
9) Real-Time Notification System: Develop an alert mechanism for farmers and agricultural officers when disease symptoms are detected.
10) Web Dashboard for Monitoring: Create a centralized dashboard to visualize disease trends, risk zones, and statistical analysis.

REFERENCES

[1] FAO, “The future of food and agriculture,” UN Report, 2021.
[2] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," Nature, vol. 521, pp. 436–444, 2015.
[3] Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). "Using deep learning for image-based plant disease detection," Frontiers in Plant Science.
[4] Hari, S. et al., "Detection of Plant Disease by Leaf Image Using Convolutional Neural Network," ViTECoN Conf., IEEE, 2019.
[5] Li, L. et al., "Plant Disease Detection and Classification by Deep Learning—A Review," IEEE Access, 2021.
[6] Zhang, S.W., Shang, Y.J., & Wang, L., “Plant Disease Recognition Based on Plant Leaf Image,” J. Anim. Plant Sci., 2015.
[7] P. Sudhakar et al., "Leaf Disease Classification Using SVM and Image Processing Techniques," IJSR, 2020.
[8] Sladojevic, S. et al., "Deep neural networks-based recognition of plant diseases by leaf image classification," Computational Intelligence and Neuroscience, 2016.
[9] Dubey, S.R., & Jalal, A.S., “Detection and classification of leaf diseases using optimized SVM classifier,” Int. J. Comp. App., 2012.
[10] Ferentinos, K.P. (2018). "Deep learning models for plant disease detection and diagnosis," Computers and Electronics in Agriculture, 145, 311–318.
[11] Too, E.C. et al. (2019). "A comparative study of fine-tuning deep learning models for plant disease identification," Comp. & Elect. in Agriculture, 161, 272–279.
[12] Brahimi, M. et al. (2017). "Deep learning for plant diseases: Detection and saliency map visualization," Human and Machine Learning, Springer.
[13] Camargo, A., Smith, J.S. (2009). "Image pattern classification for identifying plant disease agents," Comp. & Elect. in Agri., 66(2), 121–125.
[14] Patil, S.B., Kumaraswamy, Y.S. (2011). "Diagnosis and classification of grape leaf diseases using neural networks," ICCSIT, IEEE.
[15] Zhang, X. et al. (2018). "Soybean leaf disease identification using transfer learning," IEEE Access, 7, 190111–190124.
[16] Barbedo, J.G.A. (2013). "Image processing techniques for detecting, quantifying and classifying plant diseases," SpringerPlus, 2(1), 660.
[17] Hughes, D.P., Salathé, M. (2015). "An open access repository of images on plant health," arXiv:1511.08060.
[18] Nazir, M. et al. (2020). "Data augmentation techniques for plant disease detection," IEEE Access, 8, 81746–81765.
[19] Fuentes, A. et al. (2017). "Real-time tomato plant disease and pest recognition using deep learning," Sensors, 17(9), 2022.
[20] Picon, A. et al. (2019). "CNNs for mobile-based crop disease classification in the wild," Computers and Electronics in Agriculture, 161, 280–290.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY