Content is available at: CRDEEP Journals
Journal homepage: http://www.crdeepjournal.org/category/journals/ijrem/

International Journal of Research in Engineering and Management (ISSN: 2456-1029)
A Peer Reviewed UGC Approved Quarterly Journal

SJIF: 4.45

Research Paper

Development of a Plant Disease Detection and Solution System Using Convolutional Neural Networks and React Framework: A Summary

Anup Raj, Kafil Aslam[1], Neha Bhatt[1], Shivam[1] , Vaibhavi Painuly[2] and Sharad Pratap Singh[31]
1-Scholar, Department of Computer Application Tula's Institute, Dehradun
2-Assistant Professor, Department of Computer Application Tula's Institute, Dehradun.
3-Assistant Professor, Department of CSE Tula's Institute, Dehradun

ARTICLE DETAILS

Corresponding Author:
Sharad P Singh

Key words:
Plant Disease Detection, CNN, Deep Learning, Machine Learning, Flask, Web Application, Agriculture, AI Solutions, Plant Health Management.

ABSTRACT

Plant diseases pose a significant threat to global agriculture, leading to reduced crop yields and substantial economic losses. This project presents a deep learning-based system for plant disease detection and management using Convolutional Neural Networks (CNN). The system is trained on a labeled dataset comprising thousands of images from multiple plant species including tomato, potato, onion, corn, and apple, covering a wide range of disease classes. The CNN model demonstrated high accuracy in classifying diseases, enabling early detection and proactive treatment. The application first verifies the plant species through image classification, then predicts the disease type and provides scientifically validated solutions and precautionary measures. To further assist users, the system integrates a Gemini-powered API for dynamic and contextual treatment suggestions. The system is scalable, interactive, and designed to support continuous learning and improvement based on real-world use cases.

1. Introduction

Plant diseases are a widespread issue in agriculture, affecting crops of all types and leading to significant yield losses globally. These diseases can impact plants at any stage of growth, and early diagnosis is essential for timely and effective treatment. However, due to vague visual symptoms and the unavailability of agricultural experts in rural areas, it becomes challenging to identify diseases accurately and promptly. With the advancement of deep learning and computer vision, scalable and automated solutions for plant disease identification have become viable. This paper presents a CNN-based system that detects multiple diseases across different crops and supports farmers with actionable treatment solutions. The development of CNNs has revolutionized image classification tasks, enabling precise and automated analysis of plant leaf images. The goal of this project is to design a comprehensive solution that not only detects crop diseases but also provides context-specific precautionary and remedial actions. By leveraging deep learning, the system bridges the gap between limited agricultural support and real-time field diagnosis.

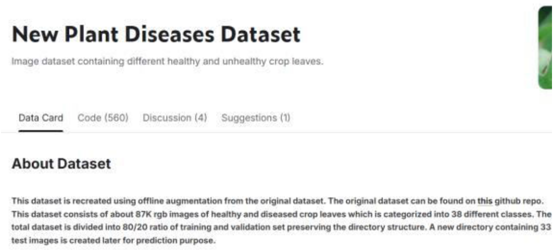


Fig 1: Dataset of Different Plant Leaves

[1]Author can be contacted at: *Department of Computer Application Tula's Institute, Dehradun*

Received: *20-05-2025*; Sent for Review on: 27*-05- 2024*; Draft sent to Author for corrections: *10-06-2025*; Accepted on*: 12-06-2025*; Online Available from 25*-06-2025*
DOI: 10.13140/RG.2.2.27108.90243

This research implements a disease detection system for several crops—namely **tomato, potato, onion, corn, and more**—using CNN[22]. It integrates a user-friendly interface and deliver detailed solutions. The model predicts the type of disease and verifies the plant species before proceeding, ensuring prediction accuracy and reliability. Users receive preventive tips, chemical or organic remedies, and in some cases, dynamic solutions powered by the Gemini API[24]. **Dataset**: The datasets are collected from Kaggle and other public repositories, comprising thousands of leaf images across different plant types and disease categories. Each dataset is carefully curated with high-quality, well-labeled images. Preprocessing steps such as resizing, normalization, and augmentations (rotation, flipping, zooming) are applied to increase the model's generalization and robustness to perform well in diverse environments.

2.Literature Review

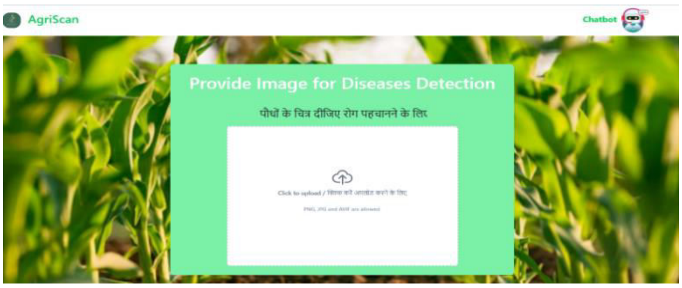
Recent advances in deep learning have significantly enhanced image-based diagnostics in agriculture. Convolutional Neural Networks (CNNs), inspired by the human visual system, effectively extract spatial patterns and have outperformed

```
# MODEL DEFINITION
model = Sequential([
    Conv2D(32, (3, 3), activation='relu', input_shape=(img_size[0], img_size[1], 3)),
    MaxPooling2D(2, 2),
    Conv2D(64, (3, 3), activation='relu'),
    MaxPooling2D(2, 2),
    Conv2D(128, (3, 3), activation='relu'),
    MaxPooling2D(2, 2),
    Flatten(),
    Dense(128, activation='relu'),
    Dropout(0.5),
    Dense(train_generator.num_classes, activation='softmax')
])

model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
```

traditional image processing methods. Popular architectures like ResNet, VGGNet, and InceptionNet have been widely applied to classify plant diseases, demonstrating strong accuracy in controlled settings.However, existing models face key challenges: limited datasets, inter-class visual similarity, and dataset bias reduce their generalization capabilities. Moreover, most past approaches emphasize accuracy but overlook real-world usability, lacking deployment-ready systems or user interaction layers. They often fail to verify if the input plant matches the predicted class, which can lead to misdiagnosis—especially when models are trained only on a specific crop. Our model addresses these issues by incorporating a plant verification step before prediction, ensuring the disease is matched only within the correct plant category. Our system also features a web-based interface using Flask, allowing real-time image upload, classification, and solution suggestions tailored per disease. Unlike previous models, which end at prediction, our system integrates dynamic treatment solutions via the Gemini API, offering up-to-date remedies and precautions. This creates a comprehensive end-to-end platform that goes beyond detection to provide actionable guidance, closing the loop between diagnosis and response. By combining strong deep learning foundations with practical deployment, plant verification, and AI-powered solutions, our model overcomes the limitations of earlier research and makes plant disease diagnosis more accessible and reliable.

This research aims to develop a **CNN-based multi-class classification model** for the accurate diagnosis of plant diseases across multiple species—**tomato, potato, onion, corn, and more**—with support for multiple disease classes per plant. The model is trained on a structured, labeled image dataset, with **data augmentation** techniques (rotation, flipping, zooming) applied to enhance generalization.



The system extracts spatial features such as color, texture, and lesion patterns using deep convolutional layers to classify input leaf images into corresponding disease categories. Performance is quantitatively evaluated using accuracy, precision, recall, and F1-score.To reduce inter-class and inter-species misclassification, the model includes a pre-classification plant verification step. The complete solution is deployed as a Flask-based web application, allowing users to upload images for real-time disease detection. Integrated with the Gemini API, the system provides contextual remedies and preventive solutions, delivering an end-to-end intelligent plant health management platform.

3. Methodology

Model architecture: The CNN architecture is designed to have an optimal balance between complexity and computational efficiency. These include: Input Layer: Processes plant leaf images resized uniformly (e.g., 128×128×3) to ensure consistency across the dataset. Convolutional Layers: Apply filters to extract relevant features such as texture, color variation, and disease patterns, followed by ReLU activation to introduce non-linearity. Pooling layers: Apply max-pooling to reduce the spatial size and the computation involved. Fully connected layer: Combine high-level features extracted from convolutional layers and map them to classification categories. Output Layer A Softmax activation function is applied for multi-class classification of plant diseases and healthy cases. Training Process: The model was developed using Tensor Flow and Keras. Training parameters include: Optimizer: Adam with a learning rate of 0.001. Loss Function: Categorical Crossentropy. The regularization techniques used dropout layers with a rate of 0.5 and L2 regularization to control overfitting. Early stopping was based on the validation loss to optimize the training time. Evaluation Metrics: The following metrics were used to evaluate performance: Accuracy : Accuracy Total Correct prediction. Precision: True positive instances in positive predictions. Remember: True Positives-actual positive correctly predicted instances. F1-Score: Harmonic mean of precision and recall. The classes were divided into an 80% training set and a 20% validation set.

Framework Integration:

The The web-based system was developed using the Flask framework. The application features the following components:
Image Upload: Allows users to upload a plant leaf image for classification Real-time Results: Displays the predicted disease class along with a score. Treatment Prescriptions: Provides treatment and prevention measures based on the predicted plant disease class using Gemini API integration The trained CNN model was saved in HDF5 (.h5) format and integrated into the backend using TensorFlow. The frontend was developed using HTML, CSS, and JavaScript, offering a clean and interactive user interface.

Implementation:

Data Preprocessing: The dataset was loaded using TensorFlow's Image Data Generator to enable efficient image augmentation and batching. Augmentation Techniques: Real-world variability was simulated using transformations like rotation, flipping, zooming, and width/height shifts. Data Splitting: The dataset was split into 80% training and 20% validation sets to ensure proper generalization.

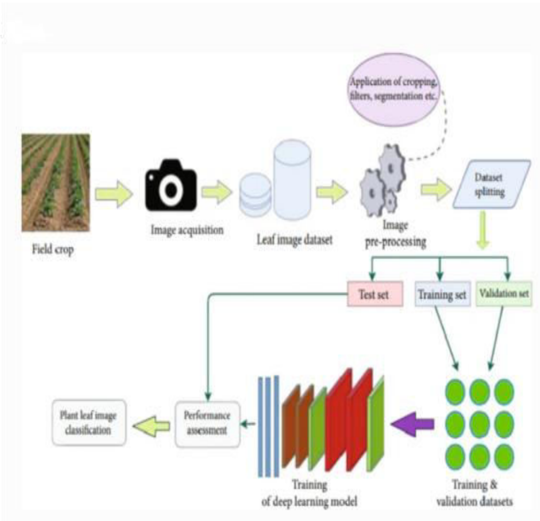


Fig 2 : Gemini API used for solutions

The proposed CNN-based system effectively classifies multiple plant leaf diseases with high accuracy (94%), showcasing strong feature extraction and generalization capabilities. Its integration with Flask ensures easy deployment and accessibility through a user-friendly web interface.

Model Training

Strengths:

High Accuracy: The 94% validation accuracy reflects the robustness of the model.
It is an easy-to-use web interface accessible even to non-technical users.

Layer Construction: Sequential CNN layers were built using Keras.
Initial experiments established the number of convolutional layers to be 4 and kernel sizes to be 3x3.
The model was trained on the augmented data in mini-batches Initial weights were randomly initialized.
Validation Monitoring: Loss and accuracy metrics were logged at every epoch. Overfitting potential was monitored with the help of early stopping.

Deployment Architecture

Model Serialization: The trained model was saved in TensorFlow's .h5 format, making it compatible with Flask-based backend integration. API Development: RESTful APIs were built using Flask to handle image uploads and return classification results dynamically. Frontend Integration: The UI was developed using HTML, CSS, and JavaScript. Users can upload plant leaf images, receive real-time predictions, and view corresponding treatment suggestions powered by the Gemini API.

4. Result:

Model Performance: On the validation set, this trained CNN was able to have the following:
Validation Accuracy: 94%
Precision, Recall, and F1-Score: All eight classes have high values, which indicates balanced performance.
Confusion Matrix Analysis: The confusion matrix showed minimal misclassifications. Minor overlaps existed between visually similar classes—for instance, Early Blight and Late Blight in tomatoes. These could be further reduced with advanced approaches like attention mechanisms or transfer learning in future iterations.

Example Predictions

Input: Leaf image of infected potato.
Predicted Class: Potato – Early Blight. Confidence: 96%
Input: Tomato leaf image with visible disease patches.
Predicted Class: Tomato – Leaf Mold. Confidence: 94%
Statistical Metrics: The statistical analysis of model performance Included:
Accuracy per Class: Above 90% for all classes.
Scalability: New diseases or plant types can be added easily.

Limitations:

Dataset Diversity: Needs broader image variety from real farm settings.
Explainability: No visual cues on what influenced predictions.
Real-life Validations: Requires testing on live field images.

Future Work:

Dataset Expansion: Add more diverse and real-world images.
Explainable AI: Combine with Grad-CAM or SHAP to visualize what is causing certain features to influence the prediction.
Mobile application development: Expand the system on mobile platforms for the real-time, on-the-go diagnosis.
Connect with agricultural systems and consult experts for refinement.

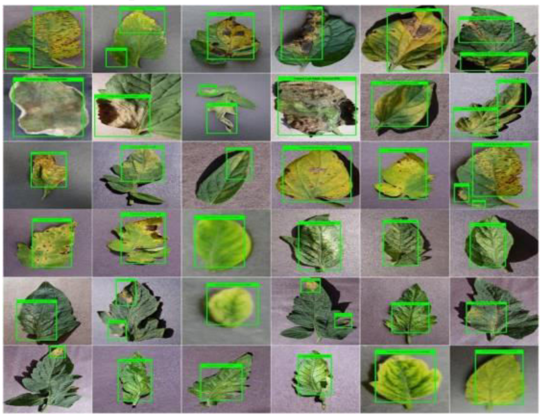


Fig 3: Different Datasets

References

Chollet, F., *Deep Learning with Python*, Manning Publications, 2021.
Khandagale, H. P., Patil, S., Gavali, V. S., Chavan, S. V., Halkarnikar, P. P., & Meshram, P. A. (2025). *Design and Implementation of FourCropNet: A CNN-Based System for Efficient Multi-Crop Disease Detection and Management*. arXiv preprint arXiv:2503.08348.
He, K., Zhang, X., Ren, S., and Sun, J., *Deep Residual Learning for Image Recognition*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 770-778, 2016.
Kaggle, *Plant Dataset*, Retrieved from https://www.kaggle.com.
Mustofa, S., Munna, M. M. H., Emon, Y. R., Rabbany, G., & Ahad, M. T. (2023). *A comprehensive review on Plant Leaf Disease*

detection using Deep learning. arXiv preprint arXiv:2308.14087.
LeCun, Y., Bengio, Y., and Hinton, G., *Deep Learning*, Nature, 521(7553), 436-444, 2015.
Abade, A. S., Ferreira, P. A., & Vidal, F. B. (2020). *Plant Diseases recognition on images using Convolutional Neural Networks: A Systematic Review*. arXiv preprint arXiv:2009.04365.
Thapa, R., Snavely, N., Belongie, S., & Khan, A. (2020). *The Plant Pathology 2020 challenge dataset to classify foliar disease of apples*. arXiv preprint arXiv:2004.11958.arXiv