An Integrated Deep Learning Framework for Plant Disease Detection, Severity Analysis, and AI-Based Cure Recommendation

Akansha Agrawal[1*], Ajay Suri[1] & Kimmi Verma[1]

1 Department of Electronics & Communication Engineering, ABES Engineering College, Ghaziabad, India agrawalakansha2022@gmail.com

Abstract. Agriculture forms the foundation of human civilization and plays a critical role in ensuring global security of food. However, diseases in crops are caused by pathogens, fungi, and bacteria significantly reduce yield and quality [1], [6], [11]. Traditional visual inspection–based diagnosis is subjective, error-prone, and not scalable [3], [6]. To address these challenges, this paper presents **PlantSarthi**, an intelligent deep learning system for automated detection of disease in leaves of plant, severity estimation, and AI-assisted advisory generation. A **Convolutional Neural Network (CNN) based on the VGG-19 architecture** is employed to classify plant diseases from leaf images. Disease severity is estimated using **Grad-CAM–based activation heatmap analysis** [7], [10], where the spatial extent of disease-relevant regions is numerically quantified and categorized into *mild, moderate,* and *severe* levels. Additionally, a Generative AI module provides disease-specific precautionary and treatment suggestions to support farmer decision-making. Experimental evaluation on a publicly available plant disease dataset achieves a **85% classification accuracy**, demonstrating the effectiveness and practical applicability of the proposed system for agricultural decision support.

Keywords: Generative AI, CNN, Deep Learning

1 INTRODUCTION

1.1 Background and Motivation

Agriculture is the backbone of food production. The damage caused to crops owing to plant diseases causes an annual **loss of 20-40%** globally. This mostly affects the developing nations as there is very little expertise available for diagnosis. Human inspection of data is prone to error and severely lacking in sustainability and scalability [3], [6], [9].

With the emergence of **Artificial Intelligence (AI) and Computer Vision**, new opportunities have been opened up for automating this process through **image recognition** with the aid of **deep learning algorithms.** Nevertheless, the existing automated systems mostly limit themselves to disease classification alone and do not provide other information like – severity of the disease or visual explanation of model decision, thereby limiting their practical usability considerably in agriculture.

A. Agnihotri et al. (eds.), *Proceedings of the Conference on Bridging Engineering Disciplines with AI and Machine Learning (BEDAIML 2026)*, Advances in Intelligent Systems Research 209,
https://doi.org/10.2991/978-94-6239-697-5_12

1.2 Problem Definition

Diseases in plants are a major threat to agricultural production worldwide, which affects millions of people every year. To ensure good management and maximum crop yield, early and accurate detection is very important. Traditional ways of doing things, which involve manually checking, are quite tedious and inaccurate too. As a result, high demand exists for systems that are automated and reliable. Examinations by a specialist manually lead to misidentification and delays.

The advancement of image analysis and machine learning has created the ability to automatically detect plant diseases, but estimating the reliability of disease severity remains complex [5], [10]. This is mostly due to public datasets lacking explicit severity annotations, which is a common problem. Deep learning models provide the ability to learn patterns related to diseases, such as discoloration, texture, and lesion structure, and provide the researcher the ability to assist in the classification, while large-scale datasets provide datasets to assist the researcher in training the models. Automation leaves the decision-making and acting to the systems to protect the crops and ease the burden of the farmer.

1.3 Solution

Our project focuses on addressing the issue of detection of diseases in plants in the agriculture sector. We understand how diseases in plants have impacted agricultural productivity globally. As a result, there is a requirement for a timely and accurate detection method that will help inform relevant measures.

To address this problem, our project aims to develop one more machine learning solution that would help in agriculture. The objective is to develop a machine learning-based tool that can effectively detect and identify plant diseases very early to enhance productivity and ensure food security.

Use supervised deep learning and the Convolutional Neural Network (CNN) architecture to influence our approach. The labeled images of various crop leaves are used to train a model based on VGG-19 for disease classification. [2], [4], [9], [13].

To estimate disease severity, Grad-CAM is applied to a trained CNN model. The activation heatmaps that are generated can help find the area in the image of leaf affected by disease. The heatmaps generated will analyze the numerical spatial extent of the infection, and its categorization is mild, moderate, and severe.

By employing optimized deep learning frameworks and batch-based inference for system design, efficiency and scalability are taken into consideration. The solution is lightweight and deployable on web-based architecture and offers practical usability in response to agriculture's needs.

Explainable AI methods describe the reasoning for the decisions made by the system [7], [8]. These methods help users understand the framework used in detection of the disease and its level of severity.

1.4 Research Objective

This research aims to create systems to identify diseases utilizing deep learning. Specifically,

1. Create a model based on CNN that can identify different diseases of plants based on images of their leaves.
2. Identify and implement suitable techniques for pre-processing input images to improve the quality of images and the performance and outcomes of the model.
3. Evaluate the capability of the model on a varied dataset of images of plant leaves that are healthy and diseased.
4. Evaluate the degree of the disease using the heatmap analysis based on Grad-CAM. This entails quantifying the extent of the disease-affected areas and categorizing the severity of the affected regions into mild, moderate, and severe levels.
5. Integrate a Generative AI–based advisory module to generate disease-specific precautionary measures and treatment suggestions based on the detected disease.
6. Design and develop a user-friendly interface for the deployment of the detection system for plant diseases, facilitating easy interaction for end-users, particularly farmers and agricultural professionals.

2 LITERATURE REVIEW

2.1 Deep Learning–Based Plant Disease Detection: A Comprehensive Review, *Boulent, J., et al. (2022)*

This study reviews deep learning techniques to detect plant disease in the last five years. The researchers examine more recent approaches based on CNNs used to identify plant disease from images of their leaves and also discuss their merits and demerits. The research study indicates that current deep learning models can achieve high classification accuracy but most of the existing systems just focus on disease detection and do not have severity estimation and explainable outputs.

2.2 Tomato Crop Disease Classification Using a Pre-Trained Deep Learning Algorithm *Aravind KR, Raja P, and Anirudh R.(2021)[2]*

The purpose of this research is to classify 3 major tomato crop diseases (Early Blight, Late Blight and Leaf Mold) using pre trained deep learning algorithm VGG16. Through transfer learning, the model was fine-tuned using tomato leaf images dataset. The paper also suggests extending such models into real-time mobile applications for farmers to identify crop diseases quickly and accurately.

2.3 Explainable Deep Learning in Agricultural Image Analysis,*Li, Y., et al. (2024)[8]*

This study applies explainable deep learning techniques, including Grad-CAM, to agricultural image analysis tasks. The authors show that explainability improves the reliability and interpretability of a model to the end-users. However, the study centers on visualization and does not go on to quantitatively assess the severity of Grad-CAM outputs or generate guidance.

The literature shows that even though new deep learning approaches have proven to be very successful in detecting plant diseases, few of them attempt to measure the severity of the plant disease, apply explainable ai methods to provide quantitative assessments, or measure the severity of a disease using explainable ai methods. Neither of these approaches have considered using Generative AI to provide disease-specific prevention and treatment recommendations. Integrating these components, the **PlantSarthi** system offers an innovative approach to the agriculture domain by incorporating Applied CNNs for disease detection, explainable AI-based severity assessment using Grad-CAM, and Generative AI-based treatment recommendations into a single, fully integrated, and operational agritech decision support system.

3 RESEARCH METHODOLOGY

3.1 Data Collection:

Gather a heterogeneous dataset consisting of images of healthy and diseased [9], [12], [13] leaves of different plants. The images should be from different plants and pathogen-affected and healthy leaves taken from leaves in controlled environments. Fig. 1. Illustrates the details of the dataset used.

Fig. 1. Dataset Details.

No Of Images	Plant	Disease Name
100	Apple	Healthy
2008	Apple	Disease Scab
2016	Apple	Disease: Black rot
1987	Apple	Disease: Cedar apple rust
1859	Corn	Disease: Cercospora leaf spot
1642	Corn	Disease: Common rust
1907	Corn	Disease: Northern Leaf Blight
1692	Grape	Disease: Black rot
1888	Grape	Disease: Esca (Black Measles)
1920	Grape	Disease: Leaf blight (Isariopsis)
1824	Potato	Disease: Early blight
1939	Potato	Disease: Late blight
1926	Tomato	Disease: Bacterial spot
1702	Tomato	Disease: Early blight
1920	Tomato	Disease: Late blight
1851	Tomato	Disease: Leaf Mold
1882	Tomato	Disease: Septoria leaf spot
1745	Tomato	Disease: Two-spotted spider mite
1741	Tomato	Disease: Target Spot
1827	Tomato	Disease: Yellow Leaf Curl Virus
1961	Tomato	Disease: Tomato mosaic virus
1790	Tomato	Healthy

3.2 Data Preprocessing:

Preprocess the collected images to standardize their size, resolution, and color distribution. Techniques such as resizing **256 × 256 pixels**, normalization, and augmentation are applied to enhance the quality and diversity of the dataset. Fig. 2 depicts the sample images from dataset.

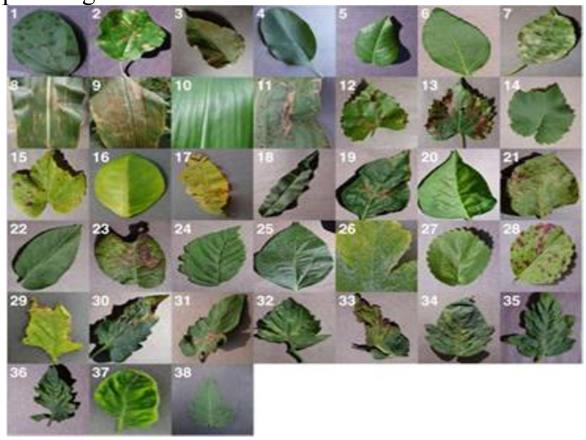

Fig. 2. Sample images in the dataset

3.3 Model Development:

A CNN architecture is designed and implemented for detection of disease in plants. This model is based on **VGG-19 architecture** [2], [4], [13], [14] which is better for image based classification.

3.4 Training and Evaluation:

Using the cleaned dataset, the designed CNN model is trained. Splitting the dataset into training, validation and testing sets purpose to assess the model effectiveness. The performance of a model is mainly assessed by classification accuracy which reflects CNN's efficiency for the detection of disease. After classification using **Grad-CAM–based activation heatmap**, the severity of disease is obtained.

3.5 Disease Severity Estimation Using Grad-CAM:

Post categorization of disease, severity estimation is done using Grad-CAM or Gradient weighted Class Activation Mapping. Grad CAM generates activation heatmaps that highlight disease relevant regions in the leaf image [7], [10]. The heatmap has been resized and adjusted, and high activation region determines category and classifies the severity of a disease as mild, moderate, and severe.

3.6 Generative AI–Based Advisory Module:

Disease specific precautionary measures and treatment guidance based on detected disease and level of severity generated by Generative AI module. Which is a decision support component to assist users to understand suitable crop management actions.

3.7 Deployment of Model:

Web based deployment is used for deploying the trained model. An interface which enables users to submit images of plant leaves and receive information on disease predictions and level of severity. In practical implementations, the deployment is designed to be light and simple to use**.**

3.8 Validation and Feedback:

The performance of the deployed system in real time is validated by accumulating feedback from end users, particularly farmers and agricultural experts. Incorporate user feedback to filter and enhance the system's functionality / usability. By using this proposed methodology, we aim to develop an efficient solution for automated detection of diseases in plants, which in turn contributes to the improvement of precision agriculture and effective techniques of crop management.

4 TECHNOLOGIES USED

A holistic approach is adopted, which focuses on efficient techstack for integration of machine learning models and user end applications. All the technologies used play a vital role in the development, training and deployment of Plant Disease Detection System.

1. The entire project code base is written in Python and this script acts as a utility that performs all data pre-processing required, as well as the training of the CNN and also acts as the backend.

2. Flask is a lightweight Python framework used to build the backend API.

This model is for the frontend and it will be used to get predictions from the user.

3. The NumPy library helps with the numerical operations and the array operations, it is also used for managing image matrices. We can use it to normalise our data and pre-process it before feeding it into our CNN.

4.TensorFlow utilized to design, train and evaluate the VGG 19 deep learning CNN model for the detection of the plant disease. It also utilized the GPU for the faster training of the model and optimal solution.

5 PROJECT WORKFLOW

The PlantSarthi system project flow is a set of stages which begin with data set preparation to system deployment.

Steps Involved:

- **Data Collection:** Visuals of healthy & diseased plant leaves are collected from opensource datasets such as *PlantVillage Kaggle* [9], [12], [13].
- **Data Preprocessing:** The images are normalized, augmented (rotation, zoom, flip) and resized to 256×256 pixels to improve dataset diversity and prevent overfitting.
- **Model Training:** We use a trained VGG-19 Convolutional Neural Network which we feed preprocessed data to out of TensorFlow. Also, we apply early stopping and save the best performing model.
- **Model Evaluation:** Classification accuracy over test dataset is used to evaluate model. In addition, Grad-CAM activation heatmaps are generated to support disease severity estimation.
- **Disease Severity Estimation:** Grad-CAM is used with the trained CNN model to present activation heat maps which in turn we use to put diseases into mild, moderate and severe categories based on the level of activation.
- **Integration with Generative AI:** Once we have a successfully trained model and have estimated the severity of disease, we integrate a Generative AI which in turn puts forth text-based suggestions for disease treatment and prevention,
- **Web Application Integration:** The trained model has been deployed using a Flask backend and a Next.js frontend, enabling customers to upload images and get predictions in real-time.
- **Deployment:** Complete system can be deployed on the server for real-time accessibility and user interaction.

6 APPLICATION FLOW

The system which has designed a deep learning model based on VGG-19 to detect plant diseases from leaf images. We start out by preparing the dataset and loading the trained model. As a user uploads a leaf image it is preprocessed and put through the CNN for disease prediction.

Post prediction we apply Grad-CAM to identify which areas of the leaf are affected by disease which in turn determines the severity. We present to the user based on what disease is present and the severity level our Generative AI which puts forth preventive and treatment suggestions.

We have put this full-scale workflow into a web application which the user interfaces with to upload images and gets back disease, severity, and advisory info in a very simple and easy to use format. Fig. 3. Illustrates the complete architecture of the system for analyzing diseases in plants.

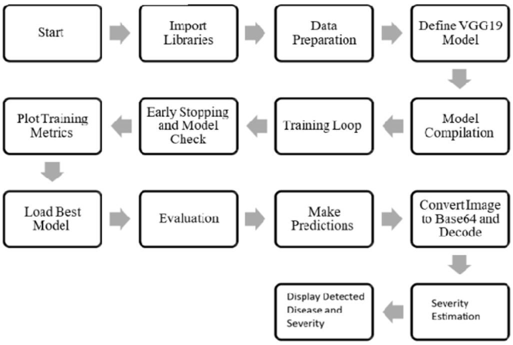

Fig. 3. System Architecture

Steps in Execution:

- User uploads or captures a leaf image through the web interface.
- The image is converted into a 256 × 256 RGB matrix for model compatibility.
- The trained VGG-19 model is loaded into the flask backend and the disease class is predicted.
- The Generative AI module receives the prediction to formulate a solution and precaution description.
- To identify affected regions and estimate disease severity, **Grad-CAM** is applied to the prediction.
- The interface will display the results which include the disease name, severity, and ensure the level and cure from AI Generation. Fig. 4 depicts our flow of the application.

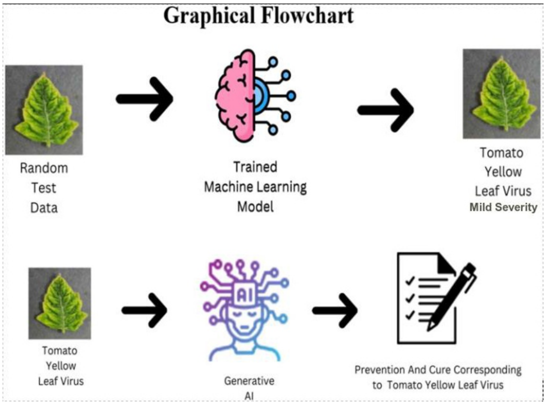

Fig. 4. Application Flow

7 EXPERIMENTAL RESULTS AND PERFORMANCE EVALUATION

Performance of our proposed **PlantSarthi** system was assessed based on model accuracy and loss trends recorded during training and validation. The **VGG-19 Convolutional Neural Network (CNN)** was trained using the preprocessed leaf image dataset and assessed for its learning behavior, convergence, and generalization. Moreover, an explainability driven disease severity by **Grad CAM** is examined.

7.1 Model Accuracy Analysis

In the Fig. 5., we see the model's accuracy which plays out over epochs. Training accuracy, which goes up very gradually across the board and then levels off, is which we see to be a sign of the model's effective learning of disease related features. Also we see that validation accuracy which also trends up although with more variation, which we take to be the model's consistent learning behavior and that it is also doing a good job of not overfitting.

By the final stages the training accuracy has hit about 75-77% which is where it levels off and at the same time we see that the validation accuracy has stabilized at around 83-85% which we take to be that the VGG19 network does an effective job of telling the difference between healthy and diseased plant leaves in our specific data set..

```
import matplotlib.pyplot as plt

plt.plot(h['accuracy'])  # Plotting training accuracy
plt.plot(h['val_accuracy'], c="red")  # Plotting validation accuracy in red
plt.title("acc vs v-acc")  # Setting the title of the plot
plt.show()  # Displaying the plot
```

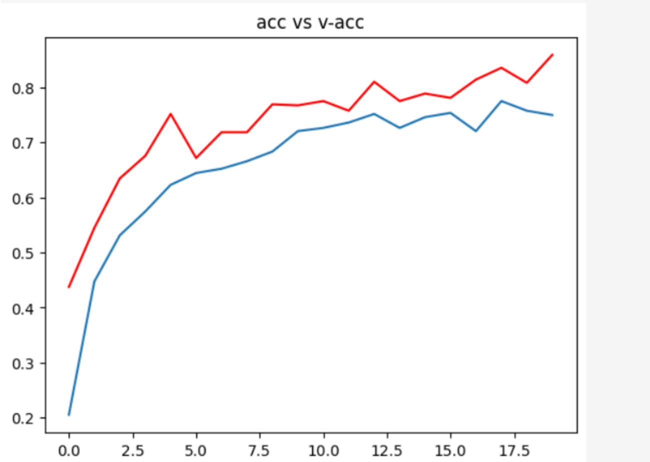

Fig. 5. Accuracy Plot training / validation

7.2 Model Loss Evaluation

The following plot (Fig. 6.) displays the training and validation loss of the current model for every training epoch. The initial losses are very high, meaning the initial network prediction has very large errors, and then they rapidly decrease, which indicates a good optimization of the weights of the model.
The training loss is steadily going down from a high initial value and seems to stabilize. The validation loss is also decreasing but with small oscillations. The lack of a clear upward trend in the validation loss suggests that the model is not strongly overfitting. In the latter epochs, both the loss curves converge to a relatively low value, indicating a stable learning behavior and higher confidence in the prediction.

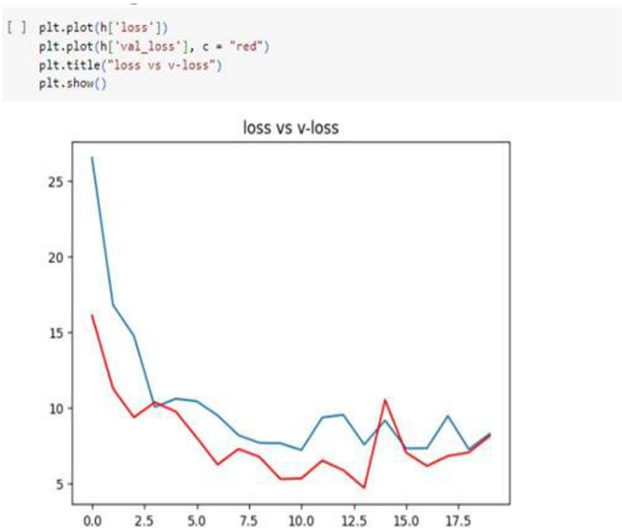

Fig. 6. Loss Plot training vs validation

7.3 Disease Severity Estimation Analysis

Disease severity estimation in this work is performed using a **Grad-CAM–based activation heatmap analysis**. After disease classification, class specific activation heatmaps are generated by Grad-CAM that identify the regions in the leaf image affected by disease, providing interpretability of the model's predictions.

To quantify severity, the heatmap is resized in order to match the input image resolution and normalized. A fixed threshold is applied to identify high-activation regions corresponding to infected areas. The severity ratio is then calculated as the proportion of activated pixels to the total number of pixels:

$$\text{Severity Ratio} = \frac{\text{Number of Activated Pixels}}{\text{Total Pixels}}$$

Based on this ratio, disease severity is categorized into three levels:

- **Mild:** severity ratio < 5%

- **Moderate:** 5% ≤ severity ratio < 25%
- **Severe:** severity ratio ≥ 25%

Due to the absence of ground-truth severity annotations in publicly available plant disease datasets, quantitative performance metrics for severity estimation are not reported. The severity categorization is therefore treated as a **rule-based, explainability-assisted decision-support mechanism** rather than a supervised prediction task.

8 CONCLUSION

The **PlantSarthi** system successfully integrates machine learning and Generative AI for automated detection of diseases in plants and recommendation generation. CNN architecture with a **VGG19** backbone shows stable learning behavior and good generalization on the PlantVillage dataset, while also demonstrating good performance regarding the classification of plant diseases from leaf images.

In addition to analysing the diseases, the setup includes a **Grad CAM based disease severity estimation** & classification of the disease into mild, moderate, and severe. **Generative AI** gives relevant advice on treatment and preventative measures, which in turn enhance the practical applicability of the system.

References

[1] J. Liu and X. Wang, "Plant diseases and pests detection based on deep learning: A review," *Plant Methods*, vol. 17, no. 22, 2021.

[2] B. Aravind, P. Raja and R. Anirudh, "Tomato Crop Disease Classification Using a Pre-Trained Deep Learning Algorithm," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 10, no. 4, pp. 45–50, 2021.

[3] R. D. G. Garcia and R. L. dos Santos, "Decision support systems for plant disease management: A review," *Computers and Electronics in Agriculture*, vol. 175, 2020.

[4] A. Fuentes, S. Yoon, S. Kim, and D. Park, "Deep learning-based plant disease detection in real-field conditions," *Sensors*, vol. 21, no. 1, 2021.

[5] M. Wang, Y. Li, and Z. Wang, "Multi-task deep learning for plant disease classification and severity estimation," *IEEE Access*, vol. 9, pp. 128–141, 2021.

[6] R. K. Gupta, A. Kumar, and S. Bansal, "Vision-based plant disease detection using deep learning: A comprehensive review," *Artificial Intelligence in Agriculture*, vol. 6, pp. 1–15, 2022.

[7] A. Akbar, M. Z. Islam, and J. Kim, "Explainable AI for plant disease diagnosis using deep learning," *Expert Systems with Applications*, vol. 213, 2023.

[8] S. Mishra, R. Sachan, and P. Singh, "An integrated decision-support system for plant disease detection and treatment recommendation," *Computers and Electronics in Agriculture*, vol. 214, 2024.

[9] H. Saleem, J. Potgieter, and K. Arif, "Plant disease detection and classification by deep learning," *Plants*, vol. 8, no. 11, 2019.

[10] N. Ferentinos and C. Kastritis, "Deep learning-based severity estimation for crop disease management," *Agronomy*, vol. 11, no. 7, 2021.

[11] A. Kamilaris, A. Kartakoullis, and F. X. Prenafeta-Boldú, "A review on the practice of big data analysis in agriculture," *Computers and Electronics in Agriculture*, vol. 143, 2020.

[12] S. Mohanty, D. P. Hughes, and M. Salathe,"Using deep learning for image-based plant disease detection,"*Frontiers in Plant Science*, vol. 7, 2016.

[13] Y. Lu, S. Yi, N. Zeng, Y. Liu, and Y. Zhang, "Identification of crop diseases using deep convolutional neural networks," *Neurocomputing*, vol. 267, pp. 378–384, 2017.
[14] J. Chen, L. Zhang, and Q. Wang, "Lightweight CNN models for real-time plant disease detection," *IEEE Access*, vol. 10, 2022.