Systems and Computing
Volume 2, Issue 2, 1-17, 2026.
https://doi.org/10.64409/sycom.v2.i2.61

Research Article
Open Access

Smart Plant Disease Diagnosis via MERN Stack Interface and PyTorch Deep Learning Models

Lokanna Kadakolmath [1], Bharath J P [1], Sujay Kulkarni [1], Shalini G V [1]

1. *Department of Computer Science & Engineering (AI & ML), Global Academy of Technology, Bengaluru, India*

Corresponding author: Lokanna Kadakolmath (lokanna@gat.ac.in).

Manuscript Review Record:
Submitted:
May 9, 2026

Accepted:
July 1, 2026

Published:
July 15, 2026

Cite This:
Lokanna K., Bharath J P, Sujay K. and Shalini G. V. "Smart Plant Disease Diagnosis via MERN Stack Interface and PyTorch Deep Learning Models". *Systems and Computing*. Volume 2, Issue 2, 1-17, 2026. https://doi.org/10.64409/sycom.v2.i2.61

Copyright:
Articles published in SyCom are open access and distributed under the terms of the Creative Commons Attribution 4.0 International License (CC BY 4.0).

Abstract- Context: Early identification of plant diseases plays a crucial role in enhancing crop productivity and promoting sustainable agricultural practices. Advances in artificial intelligence and web-based technologies have paved the way for smart systems capable of automatically diagnosing diseases in crops like rice and sugarcane. **Objective:** This research focuses on developing a smart plant disease diagnosis system that integrates deep learning techniques with a MERN (MongoDB, Express.js, React.js, Node.js) stack to provide accurate, real-time classification of rice and sugarcane leaves diseases through a user-friendly web interface. **Method:** The proposed framework employs a Convolutional Neural Network (CNN) built with PyTorch and trained using a carefully curated dataset of diseased rice and sugarcane leaf images. The developed model was incorporated into a web application built using the MERN stack to enable seamless frontend-backend communication and real-time disease prediction. The model's effectiveness was assessed using evaluation metrics such as precision, recall, F1-score, and confusion matrix analysis. **Results:** The CNN model achieved high classification performance, with an average class accuracy of 95.92%, overall classification accuracy of 91.83%, average precision of 91.85%, average recall of 92.05%, and average F1-score of 91.86%. Confusion matrix analysis further validated the model's efficiency in accurately recognizing rice and sugarcane leaves diseases. The integrated web platform demonstrated efficient and user-friendly real-time disease prediction capabilities. **Conclusions:** The developed AI-based plant disease detection system highlights the effectiveness of integrating deep learning techniques with modern web technologies to support scalable agricultural solutions. The system provides a practical solution for farmers and agronomists seeking early and accurate crop disease detection. Future enhancements may include multilingual support, mobile application integration, and agronomic advisory modules to further advance precision agriculture initiatives.

Keywords- Convolutional Neural Networks (CNNs), Crop Disease Identification, MERN Stack Application, PyTorch Deep Learning, Smart Agriculture Systems, Sustainable Agriculture.

Acknowledgement

The authors would like to express their sincere gratitude to the anonymous reviewers for their insightful comments and constructive suggestions. Their valuable feedback has greatly enhanced the quality, rigor, and clarity of this manuscript.

1. Introduction

Farming serves as a fundamental component in sustaining worldwide food availability and promoting economic development, particularly in nations that depend heavily on the agricultural sector. Nevertheless, the condition of crops continues to face major challenges because of various plant infections, which significantly reduce harvest output and adversely affect the income of cultivators. As stated by the "Food and Agriculture Organization" (FAO), pests and plant-related infections account for nearly 40% of annual agricultural production losses worldwide, leading to financial damages surpassing $220 billion each year [1]. Hence, the rapid recognition and reliable identification of crop diseases are crucial for limiting agricultural losses, increasing farming performance, and preserving long-term food sustainability.

Historically, the identification of plant infections has depended on physical assessment carried out by skilled botanists or agricultural professionals. Under this traditional method, infected crops are carefully observed for visible indications including color variation, patches, wounds, and irregular surface patterns on foliage. Even though this practice has been commonly followed for many years, it requires considerable human effort, consumes significant time, involves personal judgment, and relies greatly on specialized expertise. In several countryside and isolated farming areas, the availability of qualified experts is insufficient, which often results in postponed identification and corrective measures. Moreover, numerous crop infections display closely related visual characteristics, raising the possibility of incorrect categorization and unsuitable guidance [2]. These limitations emphasize the importance of developing intelligent, dependable, and expandable disease recognition solutions capable of supporting cultivators in continuous crop observation and agricultural management.

To enhance the effectiveness of disease identification, scholars have investigated artificial intelligence and digital image analysis approaches for automatic crop disease recognition. Initial approaches mainly utilized conventional image analysis frameworks that included image collection, enhancement, attribute extraction, and categorization procedures. Designed attributes such as pigmentation, surface pattern, structure, and boundary details were obtained through methods like "Histogram of Oriented Gradients" (HOG), "Gray Level Co-occurrence Matrix" (GLCM), and color distribution evaluation [3]. The derived attributes were subsequently categorized using computational learning methods including Support Vector Machines (SVM), k-Nearest Neighbors (k-NN), and Decision Trees [4]. Although these approaches increased automation when compared with human observation, their efficiency largely relied on manually designed attributes and regulated image acquisition environments. Differences in illumination, background distractions, picture resolution, and leaf positioning frequently affected prediction precision and overall system reliability.

New progress in Artificial Intelligence (AI) and Deep Learning (DL) has greatly revolutionized visual-based crop disease identification. Specifically, Convolutional Neural Networks (CNNs) have shown exceptional performance in autonomously extracting layered visual representations directly from plant leaf images, thereby removing the dependence on handcrafted attribute design [5]. Approaches based on CNN architecture have produced high precision and improved stability in infection recognition activities because of their capability to identify intricate spatial structures, surface details, and disease-related patterns. Advanced deep learning platforms such as PyTorch and TensorFlow have additionally accelerated experimentation and practical implementation by offering adaptable resources for network construction, optimization, and prediction processes [6].

Although deep learning architectures have attained impressive results, numerous real-world obstacles still exist in implementing these technologies for practical agricultural applications. Many existing approaches are limited to laboratory settings, offline implementations, or standalone desktop applications without scalable deployment mechanisms.

Additionally, deep learning models typically require significant computing power and lack accessible interfaces for non-technical users such as farmers and field workers. As a result, real-time accessibility, usability, and large-scale deployment remain significant concerns in current plant disease diagnosis systems [7].

Bridging this gap requires the integration of AI-powered disease classification models with scalable and user-friendly web technologies. To mitigate these obstacles, this research presents a smart plant disease diagnosis system that combines a PyTorch-based CNN classifier with a modern MERN stack (MongoDB, Express.js, React.js, and Node.js) web application. The proposed architecture enables accurate image-based disease classification while simultaneously providing real-time interaction, image upload functionality, prediction visualization, and database integration through a responsive web interface.

The React.js frontend provides an intuitive environment where users can upload plant leaf images and instantly receive diagnostic feedback. The backend, developed using Node.js and Express.js, manages API communication, handles routing operations, and interacts with the MongoDB database for storing user and prediction data. The PyTorch inference engine is integrated into the backend to process uploaded images and generate disease predictions efficiently. This integrated architecture ensures scalability, modularity, and seamless communication between the AI model and the web application components.

A major benefit of the developed framework is its accessibility and scalability. Since the platform is browser-based, it can be accessed through desktops, tablets, and smartphones without requiring specialized software installation. This feature is particularly beneficial in rural agricultural regions, where smartphone usage is increasing while technical infrastructure and expert support remain limited. By enabling real-time disease diagnosis through an internet-enabled interface, the system can help farmers in making faster and more informed crop management decisions.

Numerous contemporary research works have investigated the use of AI and deep learning methods for crop disease identification. Too et al. [4] demonstrated the effectiveness of deep CNN architectures for plant disease recognition but did not provide a scalable deployment framework for practical agricultural use. Similarly, Sladojevic et al. [5] developed a deep neural network-based plant disease classification model; however, their implementation remained largely confined to academic experimentation without real-time web integration. In contrast, the proposed system focuses not only on achieving high classification accuracy but also on delivering a real-time, field-deployable solution through full-stack web development and AI integration.

Furthermore, the modular design of the proposed system enables future extensibility and enhancement. Additional functionalities such as multilingual support, mobile application integration, GPS-based disease mapping, weather-based risk analysis, and agronomic advisory systems can be incorporated into the platform. These improvements align with the broader vision of precision agriculture and smart farming initiatives aimed at enhancing agricultural productivity through intelligent, data-driven technologies [8].

The combination of PyTorch-driven deep learning architectures with the MERN stack offers an efficient and expandable approach to addressing the ongoing issue of plant disease identification. By delivering AI-driven insights through a responsive and accessible web platform, the proposed system empowers farmers, agronomists, and agricultural organizations to perform rapid and accurate disease identification. This study describes the development, execution, and assessment of the developed framework, highlighting its efficiency, dependability, and suitability for practical agricultural settings.

2. Related Work

Crop disease identification has historically relied on physical examination and visual assessment carried out by skilled agricultural experts and plant specialists. Even though this traditional practice performs well under supervised conditions, it demands extensive human effort, requires considerable time, is vulnerable to manual inaccuracies, and is frequently unavailable to cultivators in remote or low-resource areas. With the continuous progress of AI and ML, intelligent crop disease recognition frameworks have emerged as a reliable substitute capable of delivering rapid, precise, and scalable agricultural diagnostic solutions. Specifically, DL methodologies have achieved outstanding performance in image-oriented disease recognition applications because of their capability to autonomously learn distinctive representations from leaf photographs without relying on handcrafted attribute extraction.

Initial investigations in this domain concentrated on the utilization of CNN architectures for crop disease identification. Sladojevic et al. [5] introduced one of the pioneering CNN-based approaches for identifying 13 different plant diseases using leaf images. Their work demonstrated the feasibility of deep neural networks in agricultural diagnostics and achieved encouraging classification performance. However, the study primarily relied on laboratory-collected images and lacked support for practical deployment in real-world farming environments. Similarly, Mohanty et al. [2] utilized deep CNN architectures trained on the PlantVillage dataset and achieved an impressive accuracy of 99.35% across 14 crop species and 26 disease categories. Their research significantly contributed to establishing CNNs as a reliable solution for automated disease detection. Nevertheless, the dataset consisted mostly of images captured under controlled conditions, restricting the system's capability to adapt effectively to environments characterized by varying illumination, complex backgrounds, and overlapping leaves.

Further improvements were explored by Ferentinos [3], who compared several deep learning architectures including AlexNet, VGG, and GoogLeNet for plant disease recognition. The study concluded that deeper architectures achieved superior classification performance when trained on sufficiently diverse datasets. The researcher further highlighted the significance of data enhancement methods and hyperparameter tuning in strengthening model reliability and minimizing overfitting issues. Too et al. [4] later investigated transfer learning approaches using pre-trained models such as MobileNet, ResNet50, DenseNet, and InceptionV3. Their findings indicated that transfer learning significantly reduced training complexity while maintaining high classification accuracy even with limited agricultural datasets. This approach became particularly important because collecting large-scale labeled plant disease datasets is expensive and time-consuming.

Researchers have also explored lightweight deep learning models suitable for mobile and embedded devices. Howard et al. [9] proposed MobileNet, a computationally efficient CNN architecture based on depth-wise separable convolutions. MobileNet became highly suitable for agricultural applications where inference must be performed on low-power devices or smartphones. Similarly, Sandler et al. [10] introduced MobileNetV2, which improved computational efficiency and feature representation using inverted residual structures and linear bottlenecks. These lightweight architectures facilitated the development of mobile-based disease diagnosis systems capable of operating in remote agricultural regions with limited computational resources.

In addition to CNN-based approaches, researchers have investigated advanced object detection and segmentation techniques for identifying diseased regions in plant leaves. Ren et al. [11] developed Faster R-CNN, an object detection framework that significantly improved localization accuracy and enabled precise identification of infected plant areas. Redmon et al. [12] proposed the YOLO (You Only Look Once) architecture, which offered real-time object detection capabilities with high speed and accuracy. YOLO-based models have increasingly been adopted in smart farming applications due to their suitability for real-time disease monitoring.

Furthermore, He et al. [13] introduced Mask R-CNN, which combined object detection and instance segmentation, allowing detailed identification of infected leaf regions and disease severity estimation.

The challenge of dataset diversity and domain adaptation has also received significant attention in the literature. Barbedo [14] argued that models trained solely on curated datasets such as PlantVillage often perform poorly in practical agricultural environments due to environmental variability and domain shifts. To address this limitation, researchers have explored advanced data augmentation and generative approaches. Shorten and Khoshgoftaar [15] reviewed modern image augmentation techniques and demonstrated their effectiveness in improving deep learning generalization. Similarly, Goodfellow et al. [16] introduced Generative Adversarial Networks (GANs), which have been widely used to generate synthetic diseased leaf images for dataset balancing and enhancement.

Recent studies have further integrated Internet of Things (IoT) technologies with machine learning for smart agriculture applications. Kamilaris and Prenafeta-Boldú [17] reviewed deep learning applications in agriculture and highlighted the growing importance of combining AI with IoT-based monitoring systems for precision farming. Their survey demonstrated that AI-driven agricultural systems can improve crop productivity, optimize resource utilization, and reduce economic losses caused by diseases. Likewise, Liakos et al. [18] emphasized the role of machine learning and sensor technologies in modern agriculture, particularly for disease prediction and automated crop monitoring systems.

Several researchers have also investigated cloud-based and web-enabled plant disease diagnosis systems. Surendiran et al. [19] proposed an Android application integrated with a CNN model for disease detection. Although the system demonstrated practical applicability, the authors noted computational limitations associated with on-device inference. Lorick et al. [20] later developed a cloud-based disease diagnosis framework in which images uploaded by users were processed on a remote server using TensorFlow models. While this architecture improved computational efficiency, it depended heavily on stable internet connectivity and expensive cloud infrastructure, limiting usability in rural regions.

Contemporary web application technologies have contributed substantially to connecting AI innovation with real-world implementation. Full-stack frameworks such as the MERN stack provide scalable architectures for integrating machine learning models into interactive web applications. According to Manerikar et al. [21], MERN-based systems support asynchronous communication, RESTful APIs, and responsive interfaces, making them suitable for AI-driven agricultural platforms. Porter et al. [22] also explored React.js-based interfaces for smart farming and remote monitoring applications, demonstrating the effectiveness of web technologies in improving user accessibility and interaction.

PyTorch has become one of the most widely adopted platforms for developing deep learning architectures in both scholarly and industrial studies. Owing to its dynamic computation graph, intuitive programming interface, and strong community support, PyTorch enables rapid experimentation and efficient model training. Several studies [23–26] employed PyTorch for implementing CNN-based plant disease classifiers and achieved promising results. However, many of these works focused primarily on offline model evaluation without considering real-time user interaction, deployment scalability, or integration with modern web frameworks.

Recent developments in Explainable Artificial Intelligence (XAI) and transformer-based deep learning architectures have significantly advanced crop disease detection systems. Selvaraju et al. [27] introduced Gradient-weighted Class Activation Mapping (Grad-CAM), an interpretability technique that highlights image regions contributing most strongly to CNN predictions, thereby improving transparency and user trust in AI-driven agricultural applications. Furthermore, Dosovitskiy et al. [28] proposed Vision Transformers (ViTs), which achieved remarkable performance in image classification by leveraging self-attention mechanisms to capture global contextual information and long-range spatial dependencies. Extending these advancements to agricultural disease recognition, Salman et al. [29] developed a Vision Transformer-based framework combined with a Mixture of Experts strategy, demonstrating enhanced robustness under challenging field conditions characterized by varying illumination, complex backgrounds, and environmental noise.

In parallel, Alhammad et al. [30] integrated XAI techniques with deep learning models to provide visual explanations for disease predictions, enabling more interpretable and reliable decision support for precision agriculture. More recently, Chung et al. [31] introduced a resource-efficient Edge AI framework for real-time pest and disease detection, highlighting the feasibility of deploying lightweight deep learning models on edge devices for low-latency and energy-efficient agricultural monitoring. Collectively, these advancements indicate a growing shift toward intelligent, explainable, and deployable AI solutions capable of supporting practical crop disease management in real-world agricultural environments.

As summarized in Table 1, substantial progress has been made in crop disease detection using machine learning, deep learning, explainable AI, Vision Transformers, and Edge AI. While existing studies report high accuracy, many are limited to controlled datasets and laboratory settings, with insufficient focus on real-world deployment challenges such as environmental variability, domain shift, computational constraints, dataset imbalance, and accessibility for farmers in remote areas. Although recent advances in Explainable AI and Edge AI have enhanced model transparency and deployment feasibility, few solutions integrate accurate disease detection with a practical user platform. To address these gaps, this study combines a PyTorch-based CNN model with a MERN-stack web application for real-time rice and sugarcane disease diagnosis, providing a scalable, interactive, and farmer-friendly decision-support system.

Table 1. Analytical comparison of recent crop disease detection studies.

Reference	Dataset Size	Crop Type	Architecture	Deployment Strategy	Accuracy (%)
Mohanty et al. [2]	54,306 images	Multiple crops	AlexNet, GoogLeNet	Cloud-based classification	99.35
Ferentinos [3]	87,848 images	Multiple crops	Deep CNN	Offline classification	99.53
Too et al. [4]	54,306 images	Multiple crops	ResNet50, DenseNet121, InceptionV3	Laboratory environment	99.75
Salman et al. [29]	Field-acquired plant disease images	Multiple crops	ViT + Mixture of Experts	Real-field deployment	97.20
Alhammad et al. [30]	Plant disease datasets	Multiple crops	CNN + XAI (Grad-CAM)	Explainable agriculture systems	96.80
Chung et al. [31]	Agricultural pest and disease images	Multiple crops	Lightweight Edge AI Model	Edge-device deployment	95.40
Proposed Method	9,387 images	Rice and Sugarcane	Custom CNN	Edge-compatible disease classification	95.92

3. Methodology

The developed framework combines a PyTorch-driven CNN architecture for crop disease recognition with a MERN-based full-stack web platform to provide an interactive, real-time diagnostic solution. The implementation methodology is composed of five primary stages: (1) Dataset Acquisition and Image Preparation, (2) Deep Learning Architecture Construction using PyTorch, (3) Server-Side Development and API Connectivity, (4) Client-Side Interface Development with React.js, and (5) End-to-End System Integration and Deployment.

The operational process of the framework starts with the acquisition and preparation of plant leaf photographs obtained from the publicly accessible Rice Leaf Disease and Sugarcane Plant Disease datasets (Kaggle). The collected images were resized, standardized, and enhanced through augmentation techniques to strengthen adaptability and system stability. A customized CNN architecture was constructed using PyTorch and trained to identify 4 categories of crop diseases along with healthy leaf conditions. The network was optimized with the Adam optimization algorithm and assessed using evaluation metrics including accuracy, precision, recall, and F1-score.

A curated subset of the Rice Leaf Disease and Sugarcane Plant Disease datasets was used, comprising four classes: Rice Bacterial Blight (2,367 images), Rice Brown Spot (2,053), Sugarcane Rust (2,427), and Sugarcane Yellow Disease (2,540), for a total of 9,387 images. The dataset was split into training, validation, and testing sets in a 70:15:15 ratio (6,571, 1,408, and 1,408 images, respectively). The test set contained 355 Bacterial Blight, 308 Brown Spot, 364 Rust, and 381 Yellow Disease images. To enhance robustness and reduce overfitting, data augmentation techniques including rotation, flipping, zooming, shifting, shearing, and brightness adjustment were applied to the training set. All images were resized to 224 × 224 pixels before training and evaluation.

To make the system accessible to users, the CNN model was deployed via a Flask-based API, which is integrated into a MERN stack application. The frontend, devel-oped using React.js, allows users to upload leaf images. These images are routed through a Node.js and Express.js backend to the PyTorch inference server, which processes the image and returns the predicted disease label along with a confidence score. MongoDB is used to store prediction history and image metadata. This modular architecture ensures efficient communication between components and enables real-time, scalable deployment. The entire system is designed to provide farmers and agricultural workers with quick, reliable, and accurate plant disease diagnoses through a modern web interface.

The framework architecture shown in Figure 1 represents a complete workflow for intelligent crop disease identification by integrating deep learning techniques with advanced web-based technologies. The process begins with a user, typically a farmer or agricultural expert, who uploads a leaf image through a responsive web interface developed using React.js. This frontend communicates with a backend server built using Node.js and Express.js, which handles image routing, validation, and communication with other system components. After the image is uploaded, it is transmitted to a PyTorch-powered deep learning architecture deployed on a dedicated server, commonly via a REST API. The PyTorch network analyzes the image using a trained CNN architecture and generates the predicted disease category together with a confidence value. The obtained output is subsequently transferred back to the client interface through the backend and presented to the user instantly.

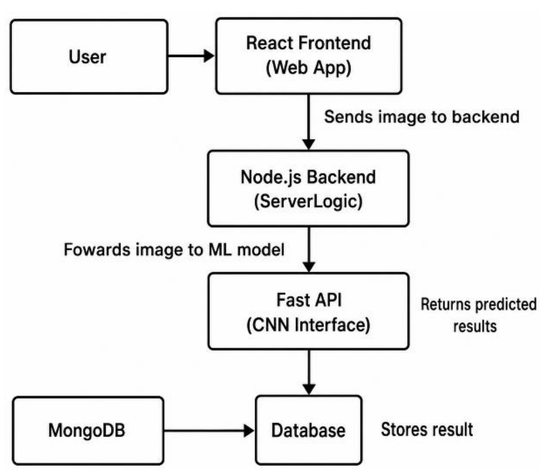

Figure 1. Structural design of the suggested model.

The proposed custom CNN architecture consists of four convolutional blocks designed for crop disease classification. The first, second, third, and fourth convolutional layers' employ 32, 64, 128, and 256 filters, respectively, with a kernel size of 3×3 and ReLU activation. Each convolutional layer is followed by Batch Normalization and a 2×2 Max-Pooling operation to improve feature extraction and reduce spatial dimensions. Dropout layers with a rate of 0.25 are incorporated after each convolutional block to mitigate overfitting.

The extracted features are flattened and passed to a fully connected dense layer containing 512 neurons with ReLU activation, followed by a Dropout layer with a rate of 0.5. The final classification layer consists of four neurons corresponding to the disease classes and employs the Softmax activation function. The model is trained using the Adam optimizer with a learning rate of 0.001, categorical cross-entropy loss function, batch size of 32, and 40 epochs.

Simultaneously, the backend interacts with a MongoDB database to store or retrieve relevant data such as prediction history, image metadata, and user records. This database ensures data persistence, allowing users to access previous diagnoses and enabling system analytics. The modular structure of this architecture separating the user interface, backend logic, model inference, and data storage ensures scalability, maintainability, and ease of deployment. The integration of PyTorch for accurate disease classification and the MERN stack for web delivery enables a highly functional, accessible, and efficient tool for real-world agricultural disease diagnosis, especially in low-resource or remote areas.

3.1. Image Classification

Image classification serves as a core element of the developed crop disease identification framework, as it facilitates the automated detection and categorization of infections from photographs of plant foliage. In farming-related applications, infections commonly generate observable indications such as color changes, patches, damaged regions, drying, and surface texture differences on leaves. Through the examination of these visual attributes, deep learning architectures can effectively identify infection-related patterns and differentiate healthy crops from affected ones. Timely and dependable disease recognition enables cultivators to implement precautionary actions at an appropriate stage, thereby minimizing agricultural damage and enhancing overall farming efficiency.

The image classification procedure starts with the acquisition of an annotated dataset consisting of photographs of both healthy and infected rice leaves. Every image is linked to a specific disease class, which functions as the reference label during the network training phase. The acquired images pass through preparation stages including resizing, standardization, noise elimination, and dataset enhancement to improve data consistency and strengthen model adaptability. Enhancement operations such as rotation, mirroring, scaling, and illumination modification assist the network in learning stable representations under diverse environmental situations.

Following the preprocessing stage, the dataset is separated into training, validation, and evaluation groups. A CNN developed with PyTorch is subsequently trained to automatically extract distinctive representations from the images. CNN-based architectures are highly suitable for visual recognition applications because they are capable of identifying spatial relationships, surface characteristics, and intricate visual structures present in leaf photographs. Throughout the learning process, the network continuously adjusts its internal weights through optimization techniques to reduce prediction mistakes and enhance classification performance.

The trained model is assessed using evaluation parameters including accuracy, precision, recall, and F1-score to determine its capability in crop disease recognition. A confusion matrix is additionally employed to examine prediction results across multiple disease classes and detect potential classification inaccuracies. After successful validation, the trained architecture is incorporated into the MERN-stack-powered web platform, enabling users to submit leaf photographs and obtain instant disease prediction results through an interactive and easy-to-use interface.

The combination of deep learning-driven image recognition with advanced web technologies offers a reliable, expandable, and accessible approach for intelligent agricultural applications. This approach supports rapid disease diagnosis and creates opportunities for future enhancements such as mobile deployment, multilingual support, and intelligent crop management advisory systems.

Figure 2 illustrate sample visualization of rice leaf disease classification showing ground truth labels and corresponding CNN model predictions for different disease categories, including Rice Bacterial Blight, Rice Brown Spot, Sugarcane Rust, and Sugarcane Yellow Disease classes.

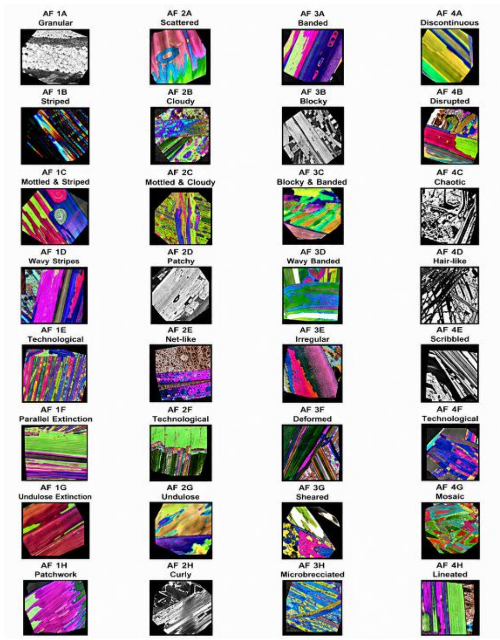

Figure 2. Ground truth and CNN-predicted labels for rice leaf disease classification.

4. Results and Discussion

The developed model was incorporated into a MERN-based full-stack web platform, enabling users to submit leaf photographs through a React-powered frontend interface. The uploaded image is processed through a Node.js backend and forwarded to a Flask API that executes the PyTorch model for prediction. The system generates predictions in under one second on average, making it suitable for real-time agricultural applications and field-level crop monitoring. Prediction results, along with confidence scores frequently exceeding 95%, are stored in MongoDB, enabling users to access their diagnosis history. Overall, the developed system demonstrates both high technical performance and practical usability for smart agriculture and automated plant disease detection. Figure 3 illustrates the output of the trained model in identifying various diseases from the input images of rice and sugarcane leaves.

To assess the efficiency and dependability of the developed crop disease identification framework, multiple experiments were performed emphasizing both the performance of the deep learning architecture and the operation of the integrated MERN–PyTorch platform. The assessment considered parameters including accuracy, precision, recall, F1-score, processing speed, and overall platform usability. The framework is driven by a customized CNN architecture constructed using PyTorch. After analysing several architectures, such as MobileNetV2 and ResNet18, the customized CNN was chosen because of its effective trade-off between prediction performance and computational cost. Trained using a curated subset of the Rice Leaf Disease and Sugarcane Plant Disease datasets containing various categories of healthy and infected plant leaf photographs, the architecture achieved an overall class-wise accuracy of **95.92%,** along with a precision of **91.85%,** recall of **92.05%,** and F1-score of **91.86%,** indicating stable and dependable disease recognition capability.

Figure 3. Predicted disease of inputted images of a rice and sugarcane leaves.

Despite achieving strong classification performance, the proposed model has certain limitations. The dataset used in this study, although containing images collected from agricultural sources, may not fully represent the diversity of real-world farming conditions. Variations in lighting, background clutter, leaf orientation, partial occlusion, image acquisition devices, and disease progression stages can introduce distribution shifts between training and deployment environments.

Consequently, model performance in field conditions may differ from laboratory evaluations. Future research should investigate domain adaptation, data augmentation using field-acquired images, and on-site validation to ensure reliable deployment in precision agriculture applications.

4.1. Confusion Matrix Analysis

Figure 4, illustrates the distribution of correct and incorrect predictions across the four disease classes, highlighting the model's ability to distinguish between visually similar diseases. A total of 1,408 test samples were used for evaluation. Most predictions are concentrated along the diagonal elements of the matrix, indicating that the majority of plant leaf images were correctly classified by the PyTorch deep learning model integrated into the MERN stack application. A total of 1,293 test instances were successfully recognized out of all evaluated samples, resulting in an overall classification accuracy of 91.83%, highlighting the robustness of the designed intelligent diagnostic framework.

The confusion matrix further reveals that certain disease classes possess visually similar characteristics, leading to a small number of misclassifications. In particular, Rice Bacterial Blight was occasionally confused with Rice Brown Spot, while Sugarcane Rust and Sugarcane Yellow diseases showed the highest mutual confusion. Specifically, 22 Sugarcane Rust samples were predicted as Sugarcane Yellow, and 29 Sugarcane Yellow samples were predicted as Sugarcane Rust. These errors may arise due to similarities in leaf discoloration patterns, lesion textures, and disease spread characteristics captured in the dataset. Despite these challenges, the model maintained consistently high accuracy values exceeding 95% for all classes in one-versus-rest evaluation.

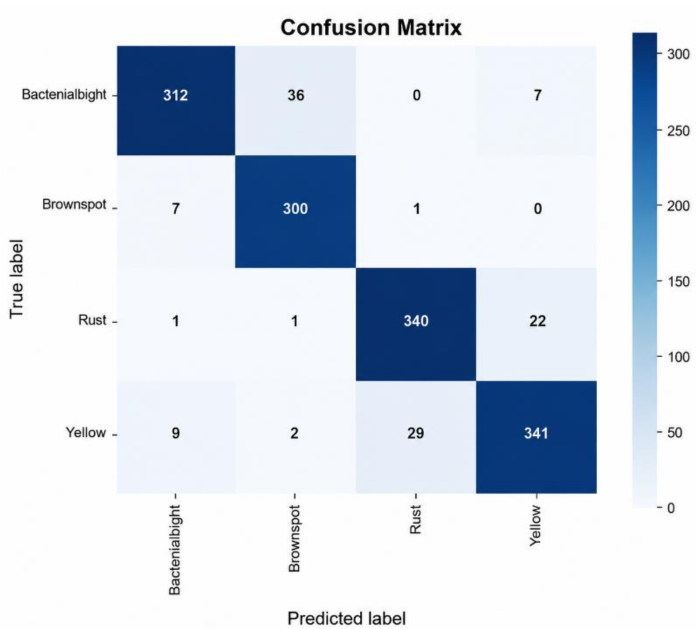

Figure 4. Confusion matrix of the proposed CNN model for classifying Rice, and Sugarcane Diseases.

4.1.1. Class-Wise Performance Metrics Calculation

The performance of the proposed CNN model was evaluated using standard classification metrics, including Accuracy, Precision, Recall, and F1-Score. These metrics were computed separately for each disease class based on the confusion matrix to provide a comprehensive assessment of the model's classification capability. Accuracy measures the proportion of correctly classified samples, Precision evaluates the reliability of positive predictions, Recall measures the model's ability to correctly identify instances of a specific class, and the F1-Score represents the harmonic mean of Precision and

Recall. For each class, True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN) were derived from the confusion matrix and used to calculate the performance metrics according to Eq. (1)–(4).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{1}$$

$$Precision = \frac{TP}{TP+FP} \tag{2}$$

$$Recall = \frac{TP}{TP+FN} \tag{3}$$

$$F1-Score = \frac{2\times(Precision \times Recall)}{Precision + Recall} \tag{4}$$

These metrics provide complementary insights into the model's performance and enable a detailed evaluation of disease-specific classification effectiveness across the Rice Bacterial Blight, Rice Brown Spot, Sugarcane Rust, and Sugarcane Yellow Disease classes. The results of these computations are summarized in Table 2.

Table 2. Class-wise performance metrics.

Disease Class	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Rice Bacterial Blight	95.74	94.83	87.89	91.23
Rice Brown Spot	96.66	88.50	97.40	92.74
Sugarcane Rust	96.16	91.89	93.41	92.64
Sugarcane Yellow	95.10	92.16	89.50	90.81
Average Performance	**95.92**	**91.85**	**92.05**	**91.86**

Among the disease categories, Rice Brown Spot exhibited the highest recall score of 97.40%, showing that the intelligent diagnosis system is highly proficient in identifying infected samples of this class with minimal false negatives. Rice Bacterial Blight achieved the highest precision value of 94.83%, indicating that predictions for this disease are highly trustworthy and contain very few false positives. All classes attained F1-scores exceeding 90%, illustrating a harmonious balance between precision and recall and substantiating the robustness of the trained deep learning model.

The class-level assessment metrics (Table 2) show that the designed CNN architecture produced stable and high-performing outcomes across every disease type. Among the four classes, Rice Brown Spot attained the utmost overall performance with an accuracy of 96.66% and recall of 97.40%, demonstrating excellent disease detection capability. The Rice Bacterial Blight class also showed strong classification performance, achieving an accuracy of 95.74%, precision of 94.83%, and an F1-score of 91.23%, indicating reliable prediction performance. Similarly, the Sugarcane Rust disease category maintained balanced results with an accuracy of 96.16%, precision of 91.89%, recall of 93.41%, and F1-score of 92.64%, confirming stable and effective classification capability. In comparison, the Sugarcane Yellow disease class achieved slightly lower metrics, with an accuracy of 95.10%, precision of 92.16%, recall of 89.50%, and F1-score of 90.81%, mainly due to visual similarity with Sugarcane Rust disease that caused a few misclassifications.

Overall, the proposed model accomplished an average class-wise accuracy of **95.92%,** precision of **91.85%,** recall of **92.05%,** and F1-score of **91.86%.** These results confirm the robustness, reliability, and effectiveness of the proposed plant disease diagnosis system, making it suitable for real-time smart agriculture applications.

4.2. Comparative Evaluation

To evaluate the effectiveness of the proposed system, the PyTorch-based model was benchmarked against alternative deep learning approaches, as presented in Table 3. For a fair comparison, all benchmark models and the proposed CNN were trained under identical experimental conditions, including dataset split, image preprocessing, augmentation techniques, input image size (224 × 224), Adam optimizer (learning rate = 0.001), batch size of 32, and the same number of training epochs. This ensured that the reported performance differences resulted solely from architectural variations among the models.

Table 3. Comparative evaluation of the proposed system.

Model	Accuracy	Model Size	Inference Time
PyTorch Custom CNN	95.92%	15.2MB	~210 ms
MobileNetV2 (transfer learning)	95.1%	9.8 MB	~180 ms
ResNet18 (transfer learning)	96.2%	45 MB	~250 ms

The comparative evaluation presented in Table 3 demonstrates the performance of three deep learning models for the proposed Plant Disease Detection System by analyzing their accuracy, model size, and inference time. The comparative evaluation revealed that ResNet18 (transfer learning) achieved the greatest classification accuracy of 96.2%, narrowly surpassing the PyTorch Custom CNN with 95.92%, whereas MobileNetV2 (transfer learning) attained 95.1%. Although ResNet18 produced slightly better accuracy, the difference compared to the proposed Custom CNN was minimal, indicating that the custom-designed architecture was highly effective for plant leaf disease classification and capable of extracting relevant disease features efficiently.

Regarding storage requirements, MobileNetV2 emerged as the most compact architecture with a model size of 9.8 MB, making it well suited for implementation on mobile platforms and edge devices that possess limited memory and processing capability. The proposed PyTorch Custom CNN maintained a moderate model size of 15.2 MB, providing a good balance among performance and deployability. In contrast, ResNet18 had the largest model size of 45 MB, which may increase memory requirements and reduce efficiency in resource-constrained environments.

Regarding prediction efficiency, MobileNetV2 recorded the fastest processing rate, requiring around 180 ms per image for inference, whereas the suggested Custom CNN achieved an inference time of nearly 210 ms. ResNet18 required the highest inference time of approximately 250 ms, indicating greater computational complexity due to its deeper architecture. Although MobileNetV2 offered faster inference and a smaller model size, its slightly lower accuracy suggests a trade-off between speed and detection performance. Overall, the proposed PyTorch Custom CNN achieved an effective balance among classification accuracy, computational efficiency, and model size, making it a practical and reliable solution for real-time plant disease detection applications in smart agriculture systems.

4.3. Training Loss and Accuracy

The training and validation evaluation graphs illustrated in Figure 5 reveal the strong learning effectiveness of the proposed CNN architecture throughout the model training phase. The framework was developed using PyTorch and trained for 40 iterations, with an early stopping mechanism employed based on validation error to minimize overfitting. As observed in Figure 5, both the training error and validation error progressively declined with the increase in epochs, reflecting continuous enhancement in the model's learning behavior. The training error decreased rapidly during the early epochs and gradually converged after nearly 25 epochs, attaining a minimal value toward the completion of training.

Likewise, the validation error exhibited a similar downward pattern without notable variations, indicating that the model maintained good generalization on previously unseen samples and avoided significant overfitting.

With respect to prediction performance, both the training accuracy and validation accuracy improved significantly during the initial training epochs before stabilizing at values exceeding 90%. The validation accuracy consistently remained close to the training accuracy throughout the learning phase, indicating stable network behavior and effective generalization capability. The absence of a significant gap between the training and validation outcomes further confirms that the CNN framework achieved well-balanced optimization without adapting excessively to the training data. In summary, the performance curves demonstrate that the developed CNN architecture achieved steady convergence, effective representation learning, and dependable classification capability for crop disease recognition.

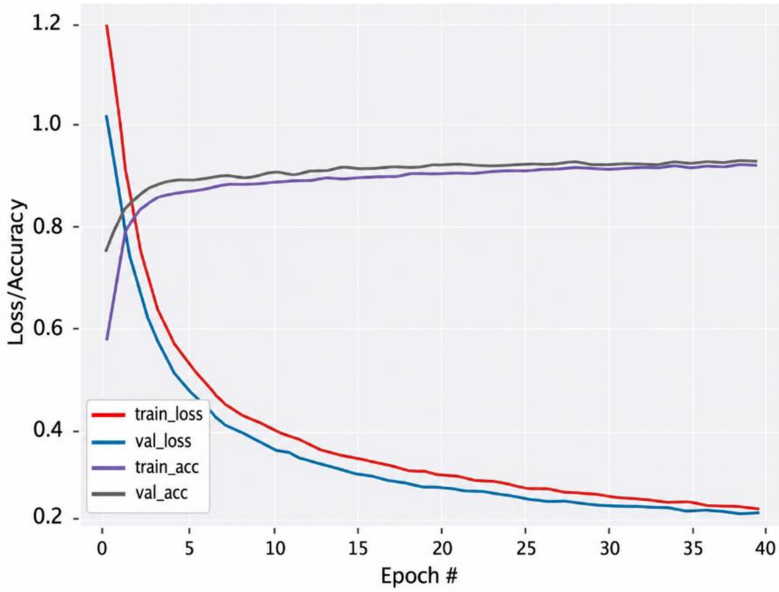

Figure 5. Training and validation loss and accuracy curves of the proposed CNN model over 40 epochs.

5. Conclusion and Future Works

This investigation proposed an intelligent crop disease diagnosis framework that integrates a deep learning-based image recognition architecture with a MERN stack web platform to facilitate interactive and real-time implementation. The developed PyTorch-based Custom CNN exhibited robust and dependable capability in recognizing crop diseases, attaining an overall class-wise average accuracy of 95.92%, precision of 91.85%, recall of 92.05%, and F1-score of 91.86% across multiple disease categories. The evaluation findings indicated that the architecture achieved exceptional results for the Brown Spot disease category, recording the highest accuracy of 96.66% along with a recall of 97.40%, while also preserving consistent performance for Bacterial Blight, Rust, and Yellow disease classes. The comparatively reduced performance observed for the Yellow disease category was primarily attributed to its visual resemblance to Rust disease, leading to a limited number of incorrect predictions.

The training and validation performance graphs additionally verified the efficiency of the developed CNN architecture, illustrating steady convergence with very limited overfitting throughout the learning phase. Performance comparison with transfer learning architectures revealed that the developed Custom CNN provided an effective compromise between prediction accuracy, architecture size, and processing speed. Although ResNet18 obtained a slightly superior accuracy of 96.2%, its increased architecture size (45 MB) and higher inference latency (~250 ms) limited its effectiveness for instant deployment scenarios. Conversely, the developed Custom CNN preserved competitive prediction capability with a balanced architecture size of 15.2 MB and an inference duration of nearly 210 ms, making it appropriate for intelligent farming systems and edge-computing implementation.

The developed system integrates a React.js frontend, Node.js backend, Flask API for model inference, and MongoDB for data management, providing a scalable, responsive, and accessible platform for farmers, researchers, and agricultural experts. The end-to-end framework enables efficient disease diagnosis through image upload and real-time prediction, contributing to early disease detection and improved crop management.

For future research, the proposed framework may be augmented with real-world agricultural image datasets exhibiting variations in illumination, complicated backgrounds, visual obstructions, and partially infected leaves to increase resilience in practical farming environments. The integration of disease severity estimation, pesticide or treatment recommendations, and crop health monitoring features would further increase the usefulness of the system for precision agriculture. In addition, developing a multilingual mobile application with offline prediction capability can improve accessibility for farmers in rural and remote regions with limited internet connectivity. Future research may also explore advanced deep learning architectures such as attention-based CNNs, EfficientNet, and Vision Transformers (ViTs) to further improve classification performance, particularly for diseases with highly similar visual symptoms. Collectively, the suggested architecture serves as a strong groundwork for the development of intelligent, adaptable, and AI-based agricultural systems that encourage efficient and sustainable farming methods.

Declarations

Author Contributions: L.K.: Formal analysis, review and editing, supervising. B.J.P.: Conceptualization, methodology, formal analysis, validation, writing original manuscript. S.K.: Conceptualization, software validation, editing. S.G.V.: Data curation, writing original manuscript.

Conflicts of Interest: The authors declare that they have no conflicts of interest.

Data Availability: The data that support the findings of this study are available from the corresponding author upon reasonable request.

Use of Artificial Intelligence (AI) Tools Statement: The authors declare that no AI tools were used in the preparation of this manuscript.

References

[1] FAO, Plant pests and diseases. Food and Agriculture Organization of the United Nations. [Online]. Available: https://www.fao.org [Accessed: Jun. 22, 2025].

[2] S. P. Mohanty, D. P. Hughes, and M. Salathé, “Using Deep Learning for Image-Based Plant Disease Detection,” *Front. Plant Sci.*, vol. 7, Sep. 2016. https://doi.org/10.3389/fpls.2016.01419.

[3] K. P. Ferentinos, “Deep learning models for plant disease detection and diagnosis,” *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, Feb. 2018. https://doi.org/10.1016/j.compag.2018.01.009.

[4] E. C. Too, L. Yujian, S. Njuki, and L. Yingchun, “A comparative study of fine-tuning deep learning models for plant disease identification,” *Computers and Electronics in Agriculture*, vol. 161, pp. 272–279, Jun. 2019. https://doi.org/10.1016/j.compag.2018.03.032.

[5] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, “Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification,” *Computational Intelligence and Neuroscience*, vol. 2016, pp. 1–11, 2016. https://doi.org/10.1155/2016/3289801.

[6] N. Zhang, M. Wang, and N. Wang, “Precision agriculture—a worldwide overview,” *Computers and Electronics in Agriculture*, vol. 36, no. 2–3, pp. 113–132, Nov. 2002. https://doi.org/10.1016/s0168-1699(02)00096-0.

[7] Md. M. Islam *et al.*, “DeepCrop: Deep learning-based crop disease prediction with web application,” *Journal of Agriculture and Food Research*, vol. 14, p. 100764, Dec. 2023. https://doi.org/10.1016/j.jafr.2023.100764.

[8] M. H. Saleem, J. Potgieter, and K. M. Arif, “Plant Disease Detection and Classification by Deep Learning,” *Plants*, vol. 8, no. 11, p. 468, Oct. 2019. https://doi.org/10.3390/plants8110468.

[9] A. G. Howard *et al.*, “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications,” 2017, *arXiv*. https://doi.org/10.48550/ARXIV.1704.04861.

[10] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT: IEEE, Jun. 2018, pp. 4510–4520. https://doi.org/10.1109/CVPR.2018.00474.

[11] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017. https://doi.org/10.1109/TPAMI.2016.2577031.

[12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 779–788. https://doi.org/10.1109/CVPR.2016.91.

[13] K. He, G. Gkioxari, P. Dollar, and R. Girshick, “Mask R-CNN,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice: IEEE, Oct. 2017, pp. 2980–2988. https://doi.org/10.1109/ICCV.2017.322.

[14] J. G. A. Barbedo, “Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification,” *Computers and Electronics in Agriculture*, vol. 153, pp. 46–53, Oct. 2018. https://doi.org/10.1016/j.compag.2018.08.013.

[15] C. Shorten and T. M. Khoshgoftaar, “A survey on Image Data Augmentation for Deep Learning,” *J Big Data*, vol. 6, no. 1, p. 60, Dec. 2019. https://doi.org/10.1186/s40537-019-0197-0.

[16] I. J. Goodfellow *et al.*, “Generative Adversarial Nets,” in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Eds., Curran Associates, Inc., 2014. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2014/file/f033ed80deb0234979a61f95710dbe25-Paper.pdf [Accessed: Oct. 24, 2025].

[17] A. Kamilaris and F. X. Prenafeta-Boldú, “Deep learning in agriculture: A survey,” *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, Apr. 2018. https://doi.org/10.1016/j.compag.2018.02.016.

[18] K. Liakos, P. Busato, D. Moshou, S. Pearson, and D. Bochtis, “Machine Learning in Agriculture: A Review,” *Sensors*, vol. 18, no. 8, p. 2674, Aug. 2018. https://doi.org/10.3390/s18082674.

[19] S. Balasubramanian, S. Rajaram, S. Selvabharathy, V. Krishnamurthy, V. Sakthivel, A. Ramprasad, “Mobile Application for Leaf Disease Classification,” in *Lecture Notes in Networks and Systems*, Singapore: Springer Nature Singapore, 2023, pp. 297–307. https://doi.org/10.1007/978-981-99-5085-0_29.

[20] L. Jain, H. V. M.A., N. M.L., and S. S.S., “Cloud-Based System for Supervised Classification of Plant Diseases Using Convolutional Neural Networks,” in *2017 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*, Bangalore: IEEE, Nov. 2017, pp. 63–68. https://doi.org/10.1109/ccem.2017.22.

[21] M. Manerikar, K. Maski, Y. Sasane, and O. Pawar, “Journeying through Query Verse: A MERN-Stack Implementation for Dynamic Web-based Inquiry,” *IJRASET*, vol. 12, no. 2, pp. 89–98, Feb. 2024. https://doi.org/10.22214/ijraset.2024.58264.

[22] P. Porter, S. Yang, and X. Xi, “The Design and Implementation of a RESTful IoT Service Using the MERN Stack,” in *2019 IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems Workshops (MASSW)*, Monterey, CA, USA: IEEE, Nov. 2019, pp. 140–145. https://doi.org/10.1109/massw.2019.00035.

[23] R. Lohith, K. E. Cholachgudda, and R. C. Biradar, “PyTorch Implementation and Assessment of Pre-Trained Convolutional Neural Networks for Tomato Leaf Disease Classification,” in *2022 IEEE Region 10 Symposium (TENSYMP)*, Mumbai, India: IEEE, Jul. 2022, pp. 1–6. https://doi.org/10.1109/tensymp54529.2022.9864390.

[24] A.A. Ali, B. Chramcov, R. Jasek, R. Katta, S. Krayem, “Classification of Plant Diseases Using Convolutional Neural Networks,” in *Lecture Notes in Networks and Systems*, Cham: Springer International Publishing, 2021, pp. 268–275. https://doi.org/10.1007/978-3-030-77445-5_24.

[25] A. Bruno *et al.*, “Improving plant disease classification by adaptive minimal ensembling,” *Front. Artif. Intell.*, vol. 5, Sep. 2022. https://doi.org/10.3389/frai.2022.868926.

[26] D. A. Kumar, P. S. Chakravarthi, and K. S. Babu, “Multiclass Support Vector Machine based Plant Leaf Diseases Identification from Color, Texture and Shape Features,” in *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, Tirunelveli, India: IEEE, Aug. 2020, pp. 1220–1226. https://doi.org/10.1109/icssit48917.2020.9214142.

[27] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization,” *Int J Comput Vis*, vol. 128, no. 2, pp. 336–359, Feb. 2020. https://doi.org/10.1007/s11263-019-01228-7.

[28] A. Dosovitskiy *et al.*, “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale,” 2020. *arXiv*. https://doi.org/10.48550/ARXIV.2010.11929.

[29] Z. Salman, A. Muhammad, and D. Han, “Plant disease classification in the wild using vision transformers and mixture of experts,” *Front. Plant Sci.*, vol. 16, p. 1522985, Jun. 2025. https://doi.org/10.3389/fpls.2025.1522985.

[30] S. M. Alhammad, D. S. Khafaga, W. M. El-hady, F. M. Samy, and K. M. Hosny, “Deep learning and explainable AI for classification of potato leaf diseases,” *Front. Artif. Intell.*, vol. 7, p. 1449329, Feb. 2025. https://doi.org/10.3389/frai.2024.1449329.

[31] H. Chung, J.-H. Kim, J. Ahn, Y. Chung, E. Kim, and W. Heo, “Resource-Constrained Edge AI Solution for Real-Time Pest and Disease Detection in Chili Pepper Fields,” *Agriculture*, vol. 16, no. 2, p. 223, Jan. 2026. https://doi.org/10.3390/agriculture16020223.