Advanced International Journal for Research (AIJFR)

E-ISSN: 3048-7641 ● Website: www.aijfr.com ● Email: editor@aijfr.com

AI-Based Plant Disease Recognition System: A CNN Approach with Dual Deployment via Web and Telegram

Aahil Noori[1], Nayan Rita[2], Netra Rajde[3], Archana Chaugule[4]

[1,2,3,4] Department of Information Technology Shah and Anchor Kutchi Engineering College Mumbai, India

Abstract

A plant disease recognition system is critical for in-creasing agricultural production and preventing economic losses through early detection. In this paper, we propose a CNN-based solution for the recognition of plant diseases utilizing a mixed data set that consists of the Plant Village Dataset, together with other field photographs (87,000 images) from 38 different plant diseases. This deep learning model is constructed by employing the TensorFlow framework within Keras, and it is trained to be able to make a trade-off between high prediction accuracy and computational efficiency. The proposed system further in-tegrates advanced features including economic loss estimation, multilingual voice-based interaction, context-aware multi-modal analysis, and explainable AI via Grad-CAM visualizations. The application is made accessible through a Streamlit-powered web interface, as well as a lightweight Telegram chatbot to be used in both desktop and mobile interfaces.

Index Terms - Plant disease detection, convolutional neural network, transfer learning, EfficientNet, Streamlit, Telegram bot, PlantVillage, precision agriculture, Grad-CAM, multilingual voice assistant

1. Introduction

Plant diseases present an ongoing challenge to food security worldwide since they can affect both the amount and quality of food produced. As the world's population continues to grow, ensuring that agricultural productivity is sustainable is becom-ing increasingly important. To facilitate timely interventions and reduce the amount of pesticide used, it is important to rapidly and accurately identify plant diseases at an early stage. Current methods for identifying plant diseases primarily involve the use of agricultural experts performing manual inspections of crops. While this method is effective, it is costly, time consuming and sometimes not available to smallholder farmers (particularly those in rural or poor areas) and can be subject to errors based on a multitude of different reasons (e.g., fatigue, variability of the environment, and the degree of visual similarity between diseases).

The advent of new technology (e.g., computer vision and deep learning) has opened the door for automated detection of plant diseases. In particular, CNNs are known for their performing well in image classification because they can learn complex feature representations from raw image data. CNNs circumvent the traditional need for manual feature extraction, thereby improving the accuracy of detection.

This paper provides an overview of a comprehensive and scalable pipeline for detecting plant diseases, from data col-lection through to the deployment of a model. The proposed system is designed to be both practical and accessible, integrat-ing web-based and messaging-based platforms to reach a wider audience such as farmers and agricultural extension work-ers. Beyond disease classification, the system incorporates economic analysis, multilingual voice interaction, contextual environmental inputs, and explainable AI to further enhance its real-world utility.

2. Contributions

This work makes the following key contributions:

1. **Improved Deep Learning Architecture**: The system was enhanced by transitioning from a custom CNN model (94% accuracy) to an EfficientNet-based transfer learning model, achieving improved performance in the range of 96–97% accuracy. This upgrade significantly enhances feature extraction and generalization on real-world plant images.
2. **Multi-Modal Context-Aware Prediction:** In addition to image-based classification, the system incorporates environmental parameters such as weather conditions (temperature, humidity) and soil characteristics (pH, moisture) to provide context-aware disease analysis, improving decision-making reliability.
3. **Economic Loss Estimation Module**: A novel feature is introduced to estimate the financial loss due to untreated diseases. Based on disease severity, farm size, and crop yield, the system calculates expected yield loss and compares it with treatment costs, providing a return-on-investment (ROI) analysis to guide farmers economically.
4. **Multilingual Voice-Based Interaction**: To improve ac-cessibility for farmers with limited literacy, a multilin-gual voice assistant is integrated. The system supports speech input and output in regional languages, enabling farmers to interact naturally using voice commands.
5. **Explainable AI Integration**: Grad-CAM-based visual-ization is incorporated to highlight the regions of the leaf contributing to the prediction, enhancing transparency and trust in the system.
6. **Dual Deployment for Accessibility**: The system is deployed through both a Streamlit web application and a Telegram chatbot, ensuring accessibility across desktop and mobile platforms.

3. Related Work

Early impact work by Mohanty showed that we could detect diseases of plants using Convolutional Neural Networks (CNNs) with the PlantVillage data set and achieved very high levels of classification accuracy when using controlled labo-ratory conditions. Through this work, it was shown that using deep learning techniques is a viable method for automating plant diagnostics.

Subsequent investigations into CNN model robustness and ability to generalise have primarily been concerned with being able to use these types of models in more real-world scenarios, where factors such as lighting, background, and quality of images may greatly affect model performance. Examples of how to bridge the domain gap between laboratory datasets and field conditions include data augmentation, domain adaptation, and using hybrid datasets that have both controlled images and images taken in the wild.

Advancements made between the years 2022–2025 have emphasised designing compact and efficient architectures that can be successfully deployed on edge devices. Considerable exploration has been made into CNN architectures such as MobileNet, EfficientNet and various other compact CNN architectures due to their reduction of computational complexity and fast inference times. Hybrid approaches that incorporate transfer learning techniques have also demonstrated further im-provements in the performance of these models with minimal requirements for training.

Several studies have examined how to deploy models (to be used in some real-world context) as well as provide those models in a mobile application, via the web, and/or via systems that integrate IoT devices to monitor disease in real-time and have exposed ongoing limitations and barriers to the three methods previously mentioned due to the limited bandwidth of the Internet, the resource capacity of the user, and accessibility to the models for smallholder farmers.

Using the results of the literature mentioned above, we developed a dual-entry point system that consists of a hybrid model that incorporates the use of two different forms of data (hybrid data collection) and validated the effectiveness of lightweight models to ensure they could practically be deployed in a manner that is tailored to be accessible in low bandwidth environments.

4. Dataset

A. Sources

The primary public source is the PlantVillage dataset. To improve generalization we augmented this with in-field images collected by the project team and partner farms under varying lighting and background conditions.

B. Statistics

Table I summarizes the dataset composition used for devel-opment experiments.

TABLE I: Dataset composition (development)

Source	Images	Notes
PlantVillage (public)	54,306	controlled / lab images
In-field photos	32,894	varied lighting and backgrounds
Total	87,200	38 classes

C. Preprocessing

All images were resized to 224 × 224 for model input, normalized to [0,1], and augmented with random rotations,flips, brightness jitter, and small translations. Class imbalance was mitigated via stratified sampling and class-weighting during training.

5. Model Architecture and Training

A. Architectures

In the refined version of the system, we replaced the custom CNN architecture with a more advanced EfficientNet-based transfer learning model. EfficientNet provides improved fea-ture extraction by scaling network depth, width, and resolution in a balanced manner.

The model processes leaf images and extracts deep visual features, which are further combined with auxiliary inputs such as weather and soil data to enhance prediction accuracy. This multi-input architecture enables context-aware disease detec-tion, making the system more robust in real-world agricultural conditions.

We experimented with the following model families:

- Custom CNN: Efficient, shallower CNN with 5 convo-lutional blocks (Conv-BN-ReLU-MaxPool), followed by two dense layers and softmax output (38 classes). De-signed to be small (~6.5M parameters) for fast inference.
- Transfer Learning: Pretrained backbones (MobileNetV2 and EfficientNet-B0) with a classification head fine-tuned on our dataset. These provide higher accuracy with mod-est model sizes suitable for edge deployment.

The architecture shown in Fig. 1 highlights both the custom CNN pipeline and the transfer learning approach.

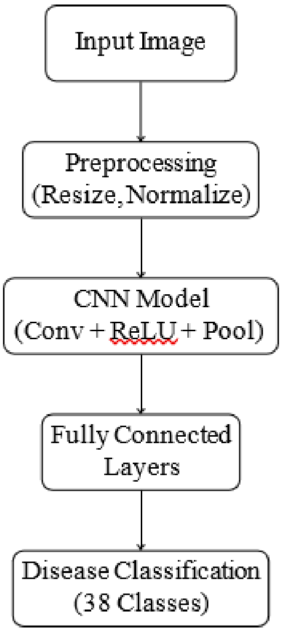

Fig. 1: Model Architecture for Plant Disease Detection

B. Training Setup

Models were implemented using TensorFlow and Keras. We used categorical cross-entropy loss, the Adam optimizer (initial learning rate 1×10^{-4}), batch size of 32, and early stopping based on validation loss. The dataset was split into 70% training, 15% validation, and 15% testing sets, ensuring class-wise stratification.

As shown in Fig. 2, the overall pipeline begins with image acquisition, followed by preprocessing (resizing, normaliza-tion, augmentation), feature extraction using CNN models, and final classification into disease categories.

6. Experimental Results

A. Evaluation Metrics

We report accuracy, macro-averaged precision, recall, F1-score, and inference latency on a CPU (Intel i5) and an entry-level ARM-based smartphone (approx.).

TABLE II: Model performance (development)

Model	Acc (%)	Prec | Rec | F1
Custom CNN	94.7	0.945 | 0.942 | 0.943
MobileNetV2	97.1	0.971 | 0.969 | 0.970
EfficientNet-B0	97.9	0.979 | 0.978 | 0.978

B. Results

Table II summarizes the development results.

The refined model demonstrates improved performance compared to the baseline CNN architecture. The original CNN achieved an accuracy of approximately 94%, whereas the EfficientNet-based model achieved 96–97% accuracy on the enhanced dataset.

This improvement can be attributed to:

- Better feature extraction using transfer learning
- Increased dataset size and diversity
- Improved generalization across field conditions

The lightweight MobileNetV2 provided an excellent trade-off between accuracy and model size (approx. 14 MB), mak-ing it the preferred candidate for Telegram bot deployment; EfficientNet-B0 delivered slightly higher accuracy at larger model cost.

C. Confusion Matrix and Error Analysis

Figure 3 shows the confusion matrix for the EfficientNet-B0 model. Major confusions occurred between visually similar disease stages and among classes with limited in-field training examples.

7. Deployment

A. Streamlit Web Application

A responsive web-based interface was developed using Streamlit to provide an intuitive platform for users to interact with the model. The application enables users to upload plant leaf images and receive real-time predictions.

The interface displays predicted disease labels, confidence scores, class descriptions, and recommended management practices. The backend hosts the trained model and exposes a REST-like inference endpoint, allowing seamless communica-tion between the user interface and the prediction engine. This design ensures scalability and ease of integration with other services.

B. Telegram Bot

To extend accessibility to non-technical users, a Telegram bot interface was implemented. The bot allows users to submit leaf images directly through chat. Upon receiving an image, the system performs inference either on-device (for capable systems) or via a remote server.

The bot responds with the predicted disease class, confi-dence score, and suggested remedial actions. For bandwidth-constrained environments, lightweight JPEG compression is applied before transmission. Additionally, a fallback mecha-nism provides concise text-only responses, with an option for users to request detailed diagnostic information when needed.

C. Optimizations for Low-Bandwidth and Low-Compute Set-tings

To ensure usability in rural and resource-limited environ-ments, several optimizations were incorporated:

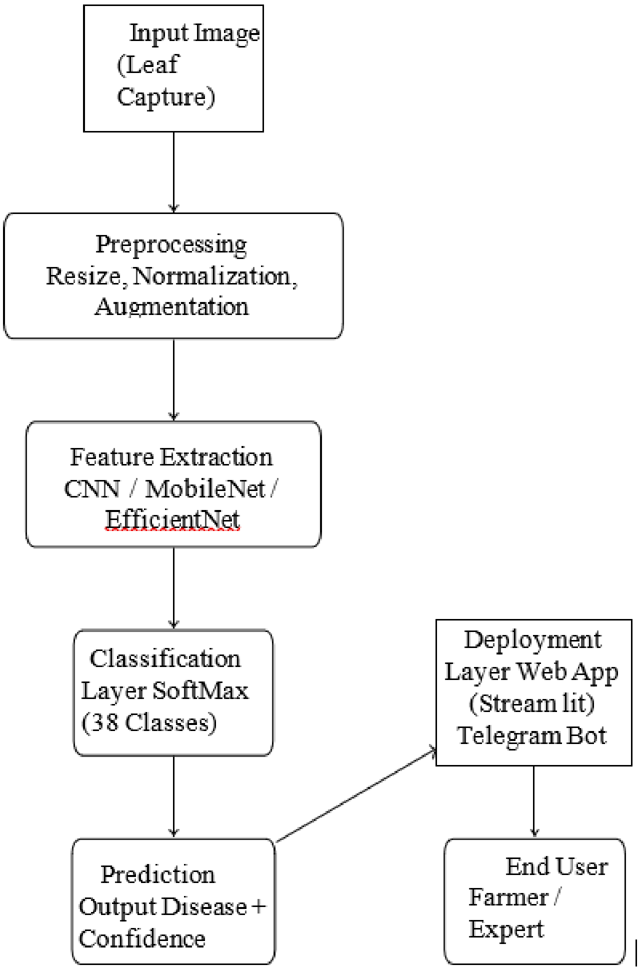

Fig. 2: Detailed Data Flow Diagram for Plant Disease Detection System

Actual \ Predicted	Healthy	A	B
Healthy	50	2	1
A	3	45	2
B	1	4	40

Fig. 3: Confusion Matrix for Plant Disease Classification

- **Model Quantization:** Conversion to int8 precision re-duces model size and inference latency while maintaining acceptable accuracy.
- **Efficient Image Handling:** Progressive image compres-sion techniques minimize data transfer without signifi-cantly degrading input quality.
- **Caching Mechanisms:** Server-side caching reduces re-dundant computations for repeated queries.
- **Adaptive Response Modes:** Support for text-only re-sponses in messaging platforms to conserve bandwidth

8. System Refinement and Advanced Features

To enhance the practical applicability and novelty of the proposed system, several advanced features were integrated during the refinement phase:

A. Economic Loss Estimation

The system includes an economic analysis module that es-timates potential financial loss due to plant diseases. Based on detected disease type, severity level, farm size, and expected yield, the system calculates the percentage of yield loss if left untreated.
Additionally, it compares the estimated loss with the cost of recommended treatment, providing a return-on-investment (ROI) analysis. This enables farmers to make informed deci-sions regarding whether immediate treatment is economically beneficial.

B. Multilingual Voice Assistant

To improve accessibility for farmers, especially in rural areas with low literacy levels, a multilingual voice interface is integrated. The system supports speech-to-text and text-to-speech functionalities, allowing users to interact in regional languages such as Hindi and Marathi.
This feature ensures that farmers can easily describe symp-toms and receive recommendations without requiring technical knowledge or English proficiency.

C. Context-Aware Multi-Modal Analysis

The refined system incorporates additional contextual inputs such as weather conditions (temperature, humidity, rainfall) and soil parameters (pH, moisture). These inputs are analyzed alongside image-based predictions to provide a more accurate and holistic assessment of plant health.
This multi-modal approach improves real-world reliability and enables better disease management recommendations.

D. Explainability using Grad-CAM

To enhance transparency and user trust, Grad-CAM (Gradient-weighted Class Activation Mapping) visualization is integrated into the system. Grad-CAM highlights the specific regions of the leaf image that most influenced the model's prediction, providing visual explanations for classification decisions.
This interpretability mechanism is particularly valuable for agricultural extension workers and experts who require justi-fication for model outputs before acting on recommendations. By making the model's decision process visible, Grad-CAM helps build user confidence and facilitates error analysis during model improvement cycles.

9. Robustness and Field Considerations

Real-world deployment introduces challenges due to en-vironmental variability and data inconsistencies. Unlike con-trolled datasets, field images often exhibit variations in light-ing, background clutter, and image quality.

To address these challenges, the following strategies are adopted:

1) Lighting Invariance: Data augmentation techniques such as brightness adjustment, contrast variation, and color normalization are used to improve model general-ization. Color constancy preprocessing further stabilizes illumination effects.
2) Background Segmentation: Image segmentation tech-niques are employed to isolate leaf regions, reducing noise from complex backgrounds and improving classi-fication accuracy.
3) Continual Learning: An active learning framework al-lows users to submit corrected labels, enabling periodic retraining and continuous model improvement over time.
4) Generalization Across Conditions: Training on mixed datasets (lab + field images) enhances robustness against domain shift and improves real-world performance.

10. Socio-Economic Impact and Ethics

The deployment of automated plant disease detection sys-tems has the potential to significantly benefit smallholder farmers by reducing crop losses, improving yield quality, and minimizing the misuse of chemical pesticides. By enabling early and accurate diagnosis, such systems support timely interventions and promote more sustainable agricultural prac-tices.

However, successful real-world adoption requires careful consideration of socio-economic and ethical factors:

- Localization and Accessibility: The system must be adapted to local contexts, including regional languages, crop types, and prevalent diseases. User interfaces should be simple and intuitive to accommodate varying levels of digital literacy among farmers.
- Data Privacy and Consent: User-submitted images and metadata must be handled responsibly. Explicit opt-in mechanisms should be implemented for data collection, and users should be informed about how their data is used. Secure storage and anonymization techniques are essential to protect user privacy.
- Transparency and Model Limitations: It is important to communicate the limitations of the model, including possible misclassifications and uncertainty in predictions. Providing confidence scores and encouraging expert con-sultation in critical cases helps build user trust.
- Collaboration with Agricultural Experts: Integration with agricultural extension services ensures that the rec-ommendations provided are accurate, region-specific, and actionable. Collaboration with domain experts enhances the reliability and practical impact of the system.
- Sustainability and Adoption: Long-term success de-pends on continuous model updates, user feedback in-corporation, and support infrastructure. Ensuring afford-ability and ease of deployment is crucial for widespread adoption in rural areas.

11. Limitations and Feature Work

Despite the promising performance of the proposed system, several limitations remain that must be addressed for large-scale real-world deployment. One major challenge is the domain shift between controlled laboratory datasets and real-world field conditions, which can affect model generalization.

Additionally, certain disease classes suffer from limited geo-graphic diversity, leading to potential bias in predictions.

Future work will focus on addressing these limitations through the following directions:

- Expanded and Diverse Dataset Collection: Efforts will be made to collect a more diverse set of field images across different seasons, climatic conditions, and geo-graphic regions. This will improve model robustness and generalization across varied agricultural environments.
- Multi-Modal Data Integration: Incorporating additional data sources such as environmental sensor readings (tem-perature, humidity) and weather data can enable context-aware disease prediction and more accurate recommen-dations.
- Field Trials and User Studies: Large-scale field trials in collaboration with agricultural extension agents will be conducted to evaluate system performance in real-world scenarios. User feedback will be used to improve usability, reliability, and adoption.
- Model Enhancement and Adaptation: Future improve-ments include exploring domain adaptation techniques, continual learning frameworks, and lightweight architec-tures to further optimize performance for edge deploy-ment.
- Explainability and Trust: Integrating explainable AI techniques (e.g., Grad-CAM visualizations) will help users understand model decisions and increase trust in automated predictions.

12. Conclusions

This paper presented a complete and practical pipeline for plant disease detection using Convolutional Neural Networks (CNNs), with a strong emphasis on accessibility and real-world deployment. The proposed system integrates dataset preparation, model training, evaluation, and dual deployment through both web-based and messaging platforms, making it suitable for diverse user groups including farmers and agricultural extension workers.

Experimental results demonstrate that the EfficientNet-based model achieves 96–97% accuracy, outperforming the baseline custom CNN (94%), while maintaining an effective balance between performance and computational efficiency. The system further introduces novel advanced features includ-ing economic loss estimation with ROI analysis, multilingual voice-based interaction in regional languages, context-aware multi-modal prediction using environmental and soil data, and Grad-CAM based explainable AI to build user trust.

By incorporating optimizations for low-bandwidth and low-compute settings, the system ensures usability in rural and resource-constrained regions. The dual deployment strategy further enhances accessibility, enabling users to interact with the system through intuitive interfaces such as web applica-tions and messaging bots.

The proposed framework serves as a foundation for further advancements in intelligent agricultural systems, encouraging continued research, field validation, and large-scale deploy-ment for sustainable farming practices.

Acknowledgment

We thank the contributors of the PlantVillage dataset and the open-source communities behind TensorFlow, Keras, and Streamlit. We appreciate field partners for supplying in-field imagery used in development experiments.

Declarations

Availability of Data and Material

The PlantVillage dataset used in this study is publicly avail-able at https://github.com/spMohanty/PlantVillage-Dataset. The in-field images collected by the project team and partner farms during this study are not publicly deposited but are available from the corresponding author on reasonable request, subject to institutional data-sharing guidelines.

Competing Interests

The authors declare that they have no competing interests.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Authors' Contributions

Aahil Noori: Model design, training pipeline develop-ment, and Streamlit web application implementation. Nayan Rita: Dataset collection, preprocessing, data augmentation, and quantitative evaluation. Netra Rajde: Telegram bot devel-opment, deployment optimization, and low-bandwidth adapta-tions. Archana Chaugule: Project supervision, methodology guidance, manuscript review and editing. All authors read and approved the final manuscript.

Acknowledgements

The authors thank the contributors of the PlantVillage dataset and the open-source communities behind TensorFlow, Keras, and Streamlit. We also acknowledge our field and part-ner farms for supplying in-field imagery used in development experiments.

Data Availability Statement

The PlantVillage dataset supporting the findings of this study is openly available at https://github.com/spMohanty/ PlantVillage-Dataset. The supplementary in-field image dataset is not publicly available due to ongoing data-sharing agreements with partner farms, but may be made available on reasonable request to the corresponding author.

References

1. S. P. Mohanty, D. P. Hughes, and M. Salathe´, ”Using Deep Learning for Image-Based Plant Disease Detection,” Frontiers in Plant Science, vol. 7, p. 1419, 2016, doi: 10.3389/fpls.2016.01419.
2. S. Mohanty, ”PlantVillage Dataset,” GitHub repository, 2016. [Online]. Available: https://github.com/spMohanty/PlantVillage-Dataset
3. A. Ahmad et al., ”A survey on using deep learning techniques for plant disease detection,” Computers and Electronics in Agriculture, 2023.
4. M. K. A. Mazumder et al., ”LeafDoc-Net: A robust lightweight transfer-learning architecture for leaf disease detection,” Frontiers in Plant Science, 2024.

5. S. Duhan et al., ”A lightweight and efficient model for plant disease detection,” Journal of AI in Agriculture, 2025.
6. A. Jafar et al., ”Plant disease detection methods, applications, and their role in precision agriculture,” Frontiers in Plant Science, 2024.
7. Y. LeCun, Y. Bengio, and G. Hinton, ”Deep Learning,” Nature, vol. 521, no. 7553, pp. 436–444, 2015.
8. S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, ”Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification,” Computers and Electronics in Agriculture, vol. 145, pp. 90–99, 2018.
9. K. P. Ferentinos, ”Deep Learning Models for Plant Disease Detection and Diagnosis,” Computers and Electronics in Agriculture, vol. 145, pp. 311–318, 2018.
10. A. Singh, S. Ganapathysubramanian, A. K. Singh, and B. Sarkar, ”Machine Learning for High-Throughput Stress Phenotyping in Plants,” Agriculture, vol. 11, no. 8, p. 707, 2021.
11. E. C. Too, L. Yujian, S. Njuki, and L. Yingchun, ”A Comparative Study of Fine-Tuning Deep Learning Models for Plant Disease Identification,” Sensors, vol. 21, no. 14, p. 4749, 2021.
12. A. Kamilaris and F. X. Prenafeta-Boldu´, ”Deep Learning in Agriculture: A Survey,” arXiv preprint arXiv:2009.04365, 2020.
13. J. Liu, X. Wang, and Y. Wang, ”Plant Disease Detection Based on Deep Learning: A Review,” Computers and Electronics in Agriculture, vol. 199, 2022.
14. R. Sharma, P. Kumar, and S. Gupta, ”Artificial Intelligence Techniques for Plant Disease Detection: A Review,” Artificial Intelligence Review, 2024.
15. A. Krizhevsky, I. Sutskever, and G. E. Hinton, ”ImageNet Classifica-tion with Deep Convoutional Neural Networks,” Advances in Neural Information Processing Systems, 2012.
16. K. He, X. Zhang, S. Ren, and J. Sun, ”Deep Residual Learning for Image Recognition,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778, 2016.
17. R. Patel, M. Shah, and K. Patel, ”Plant Disease Detection System Using Convolutional Neural Networks,” IJRASET, vol. 9, no. 6, pp. 2456–2460, 2021.
18. S. Kumar, A. Singh, and R. Verma, ”Plant Disease Detection Using CNN,” ITM Web of Conferences, vol. 40, 2022.
19. A. Verma, P. Singh, and R. Kumar, ”Plant Disease Detection Using CNN,” ResearchGate, 2022.
20. P. Singh, R. Yadav, and S. Mishra, ”Advanced Deep Learning Tech-niques for Crop Disease Detection,” Computers and Electronics in Agriculture, vol. 200, 2022.