Efficient Convolutional Neural Network for Real-Time Crop Disease Detection in Precision Agriculture

Akash Shivadas Chatake[*,1] Arthika Birajdar[2]
Vaishnavi Dhuttarge[2] [*†‡]

Received: March 11, 2026 Revision: March 11, 2026 Published: March 11, 2026

Abstract

Early detection of crop diseases is essential for minimizing agricultural losses and optimizing pesticide application in precision agriculture. Traditional crop monitoring methods rely heavily on manual inspection by agricultural experts, which can be time-consuming, inconsistent, and often inaccessible to small-scale farmers.

This study proposes a lightweight Convolutional Neural Network (CNN) architecture for real-time crop disease detection using leaf images. The model is trained on a balanced subset of the PlantVillage dataset containing images of healthy and diseased crop leaves. Data preprocessing and augmentation techniques are applied to improve model robustness and generalization.

Experimental results demonstrate that the proposed model achieves **90.50% test accuracy**, with **90.54% precision** and **90.50% recall** for binary classification (healthy vs. diseased crops). The architecture consists of three convolutional layers followed by fully connected layers, containing approximately 900,000 trainable parameters.

The trained model has a compact size of **3.6 MB** and achieves inference times between **45 ms and 120 ms** on standard CPU hardware. These characteristics make the system suitable for deployment on edge devices such as IoT cameras, smartphones, and embedded agricultural monitoring systems.

The results demonstrate that carefully designed lightweight CNN architectures can deliver reliable crop disease detection while maintaining low computational requirements, enabling practical AI-driven solutions for real-world agricultural environments.

[*] [*] Corresponding author: akash@chatakeinnoworks.com

[†1] Department of Information Technology, Shri Siddheshwar Polytechnic, Solapur, Maharashtra 413002, India; Founder & Research Director, Chatake Innoworks Pvt. Ltd., India.

[‡2] Department of Information Technology, Shri Siddheshwar Polytechnic, Solapur, Maharashtra 413002, India.

Keywords: Convolutional Neural Networks; Crop Disease Detection; Deep Learning; Precision Agriculture; Edge Computing; Plant Disease Classification; Computer Vision.

1 Introduction

Global agriculture faces significant challenges related to crop diseases, resulting in approximately 20–40% yield losses for major crops annually [1]. The Food and Agriculture Organization (FAO) reports that plant pests and diseases have substantial economic impact on global agricultural production. Early detection of crop diseases is therefore essential for minimizing economic losses, reducing excessive pesticide use, preventing disease spread across fields, and improving crop yield through timely intervention.

Traditional crop monitoring methods rely on visual inspection by trained agronomists or farmers. While effective in controlled environments, these approaches present several limitations. Manual inspection across large agricultural areas is time-consuming and labor-intensive. Furthermore, disease identification depends heavily on individual expertise, which can lead to inconsistent assessments. In many rural and resource-constrained regions, access to trained agricultural specialists is limited, resulting in delayed detection and increased crop damage.

1.1 Deep Learning for Crop Disease Detection

Recent advances in computer vision and deep learning have enabled automated crop disease detection systems. Convolutional Neural Networks (CNNs) have become the dominant approach for image-based classification tasks due to their ability to learn hierarchical feature representations directly from image data.

Previous studies have demonstrated strong performance of CNN-based approaches in agricultural applications. For example, CNN models have achieved classification accuracies between 85% and 95% on crop disease datasets [2]. Modern deep learning architectures also enable real-time inference on image data [3]. In addition, techniques such as transfer learning allow models trained on large-scale datasets to be adapted for agricultural tasks with limited labeled data [4,5]. These models have shown strong generalization capabilities across different crop varieties and growing conditions [6].

Despite these advances, deploying AI-based agricultural monitoring systems introduces several practical challenges. Edge devices such as IoT sensors and mobile platforms often have limited computational resources and memory. Real-time decision-making requires low-latency inference, typically under 100 milliseconds per image. In addition, battery-powered agricultural devices must operate using energy-efficient models while maintaining acceptable classification accuracy.

1.2 Research Contributions

This work addresses the trade-off between model accuracy and computational efficiency by proposing a lightweight CNN architecture designed for real-time crop disease detection. The main contributions of this study are summarized as follows:

1. Development of a lightweight three-layer CNN architecture containing approximately 900,000 parameters while achieving 90.50% classification accuracy.
2. Implementation of a complete training pipeline using PyTorch with data augmentation techniques to improve model generalization.
3. Experimental evaluation using a balanced subset of the PlantVillage dataset containing 1,000 crop disease images.
4. Comprehensive performance evaluation including precision, recall, F1-score, and confusion matrix analysis.
5. Demonstration of practical edge deployment feasibility through analysis of model size and inference time.

2 Related Work

2.1 Plant Disease Detection Using Deep Learning

Recent research has demonstrated the effectiveness of deep learning techniques for crop disease classification. Convolutional Neural Networks (CNNs) have been widely adopted for image-based plant disease detection due to their ability to automatically learn hierarchical visual features. Mohanty et al. [7] reported classification accuracies between 90–95% on plant disease datasets using deep CNN architectures with transfer learning. Their work highlighted the potential of deep learning models for agricultural diagnostics using leaf images.

However, many existing approaches rely on large pre-trained models that require substantial computational resources. These models are often designed for high-performance GPU environments and are typically evaluated using batch-processing workflows rather than real-time inference systems. Additionally, many studies rely on models pre-trained on ImageNet without domain-specific optimization for agricultural applications, which may limit their efficiency in real-world deployments.

2.2 Lightweight Neural Networks for Edge Deployment

With the increasing adoption of IoT devices and edge-based monitoring systems, lightweight neural network architectures have become an important research focus. Architectures such as MobileNet and EfficientNet utilize depthwise separable convolutions to significantly reduce the number of model parameters while maintaining competitive performance [8]. These techniques allow deep learning models to operate efficiently on resource-constrained devices.

Additional model compression techniques have also been explored. Knowledge distillation enables smaller neural networks to learn from the outputs of larger, more complex models [9]. Similarly, quantization methods reduce model precision to lower memory requirements and improve computational efficiency with minimal impact on classification accuracy [10]. These approaches have shown promise in enabling deep learning inference on mobile and embedded platforms.

2.3 Identified Research Gap

Although significant research has focused on improving classification accuracy using large neural networks or improving efficiency using compressed models, relatively few studies systematically explore the trade-off between accuracy and computational efficiency for agricultural deployment. Many existing implementations assume cloud-based inference, which introduces latency and connectivity challenges in rural environments.

Furthermore, some lightweight approaches sacrifice classification accuracy in order to reduce model size, resulting in performance levels below 80%. Other approaches require specialized hardware such as GPUs or high-performance edge accelerators, which may not be accessible in developing agricultural regions.

This research aims to address this gap by proposing a lightweight and reproducible CNN architecture that maintains competitive classification accuracy while significantly reducing computational complexity. The proposed model achieves approximately 90.5% accuracy while maintaining low inference latency, making it suitable for deployment on practical edge-based agricultural monitoring systems.

3 Proposed Methodology

3.1 CNN Architecture

The proposed model consists of three convolutional blocks followed by two fully connected layers for binary classification of crop leaf images. The network processes an input image $\mathbf{x}$ through a sequence of convolution, activation, and pooling operations.

$$\mathbf{c}_1 = \text{ReLU}(\text{Conv}_1(\mathbf{x})) \tag{1}$$
$$\mathbf{p}_1 = \text{MaxPool}(\mathbf{c}_1) \tag{2}$$
$$\mathbf{c}_2 = \text{ReLU}(\text{Conv}_2(\mathbf{p}_1)) \tag{3}$$
$$\mathbf{p}_2 = \text{MaxPool}(\mathbf{c}_2) \tag{4}$$
$$\mathbf{c}_3 = \text{ReLU}(\text{Conv}_3(\mathbf{p}_2)) \tag{5}$$
$$\mathbf{p}_3 = \text{MaxPool}(\mathbf{c}_3) \tag{6}$$

The resulting feature maps are flattened and passed through fully connected layers:

$$\mathbf{f} = \text{Flatten}(\mathbf{p}_3) \tag{7}$$
$$\mathbf{h} = \text{ReLU}(\text{FC}_1(\mathbf{f})) \tag{8}$$
$$\hat{\mathbf{y}} = \text{Softmax}(\text{FC}_2(\mathbf{h})) \tag{9}$$

Architecture Specification

- Input size: $224 \times 224 \times 3$ RGB image
- Conv Block 1: $\text{Conv2D}(3 \to 32, 3 \times 3, \text{padding} = 1) + \text{ReLU} + \text{MaxPool}(2 \times 2)$
- Conv Block 2: $\text{Conv2D}(32 \to 64, 3 \times 3, \text{padding} = 1) + \text{ReLU} + \text{MaxPool}(2 \times 2)$
- Conv Block 3: $\text{Conv2D}(64 \to 128, 3 \times 3, \text{padding} = 1) + \text{ReLU} + \text{MaxPool}(2 \times 2)$
- Flatten layer: 100,352 features
- Fully Connected Layer: $\text{Linear}(100{,}352 \to 128) + \text{ReLU} + \text{Dropout}(0.5)$
- Output layer: $\text{Linear}(128 \to 2) + \text{Softmax}$
- Total parameters: $\sim 900{,}000$

3.2 Loss Function and Optimization

The model is trained using cross-entropy loss for binary classification:

$$\mathcal{L} = -\sum_{i=1}^{2} y_i \log(\hat{y}_i) \tag{10}$$

where y_i represents the true class label and $\hat{y}_i$ denotes the predicted probability.

Model parameters are optimized using the Adam optimizer with learning rate $\alpha = 0.001$:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \tag{11}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \tag{12}$$

$$\theta_t = \theta_{t-1} - \alpha \frac{m_t}{\sqrt{v_t} + \epsilon} \tag{13}$$

where $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$.

3.3 Data Augmentation

To improve generalization and reduce overfitting, several data augmentation techniques are applied during training:

- Random horizontal flip (probability = 0.5)
- Random rotation within $\pm 15°$
- Color jitter for brightness, contrast, and saturation (± 0.2)
- Image normalization using ImageNet statistics (mean: [0.485, 0.456, 0.406], std: [0.229, 0.224, 0.225])

These transformations effectively increase dataset diversity and improve model robustness.

3.4 Training Configuration

The model is trained using the following configuration:

- Batch size: 32
- Number of epochs: 20
- Learning rate: 0.001
- Weight decay: 0
- Loss function: Cross-entropy
- Optimizer: Adam
- Weight initialization: Kaiming initialization for convolutional layers
- Regularization: Dropout with probability 0.5

4 Experimental Methodology

4.1 Dataset Description

The experiments are conducted using a subset of the publicly available PlantVillage dataset from Kaggle (emmarex/plantdisease). The full dataset contains more than 54,000 labeled images of healthy and diseased crop leaves across multiple plant species.

For controlled experimentation, a balanced subset of 1,000 images is selected covering three crop categories. The dataset is divided into training, validation, and testing sets using a 70% / 15% / 15% split. All images are resized to 224×224 pixels before being provided as input to the CNN model.

Dataset Composition

Crop	Healthy	Diseased	Total
Tomato	150	150	300
Potato	150	150	300
Pepper	100	100	200
Total	**500**	**500**	**1,000**

Table 1: Dataset composition and class distribution

4.2 Evaluation Metrics

Model performance is evaluated using standard classification metrics commonly used in binary classification tasks.

1. **Accuracy**

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

2. **Precision**

$$\text{Precision} = \frac{TP}{TP + FP}$$

Precision represents the proportion of correctly predicted positive cases among all predicted positives.

3. **Recall**

$$\text{Recall} = \frac{TP}{TP + FN}$$

Recall measures the ability of the model to correctly identify diseased crops.

4. **F1-Score**

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

The F1-score balances precision and recall and is particularly useful when evaluating classification performance.

5. **Confusion Matrix**

The confusion matrix summarizes prediction outcomes:

$$\begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix}$$

where TP represents true positives, TN true negatives, FP false positives, and FN false negatives.

5 Results

5.1 Training Performance

The CNN model was trained for 20 epochs with periodic validation to monitor learning behavior and detect potential overfitting.

Epoch	Train Loss	Train Acc (%)	Val Loss	Val Acc (%)
1	0.6220	75.62	0.3786	91.50
2	0.1964	94.88	0.2789	89.50
4	0.0832	96.38	0.2432	92.50
10	0.0511	98.00	0.2458	93.50
20	0.0718	97.62	0.5251	90.50

Table 2: Epoch-by-epoch training and validation performance

The training process shows rapid convergence during early epochs. Validation accuracy reaches approximately 91.5% in the first epoch and peaks at 93.5% around epoch 10. The final validation accuracy stabilizes at approximately 90.5%, indicating good generalization performance.

5.2 Test Set Evaluation

The model achieves balanced classification performance across all evaluation metrics, indicating reliable discrimination between healthy and diseased crop samples.

Metric	Value
Accuracy	90.50%
Precision	90.54%
Recall	90.50%
F1-Score	0.9050
Loss	0.5251

Table 3: Test set performance metrics

5.3 Per-Class Performance

Class	Precision	Recall	F1	Support
Healthy	0.91	0.90	0.90	100
Diseased	0.90	0.91	0.90	100
Weighted Avg	**0.9050**	**0.9050**	**0.9050**	**200**

Table 4: Per-class classification metrics

The results demonstrate balanced performance between the two classes, with slightly higher recall for diseased crops.

5.4 Confusion Matrix

	Predicted Healthy	Predicted Diseased
Actual Healthy	89	11
Actual Diseased	8	92

Table 5: Confusion matrix for test set predictions

The model correctly classifies 181 out of 200 samples, corresponding to an overall accuracy of 90.5%. The false negative rate (missed disease cases) is relatively low, which is desirable in agricultural disease detection systems where missing a disease instance can have significant economic impact.

5.5 Inference Performance

These results demonstrate that the proposed CNN model can operate efficiently on CPU-based systems without requiring specialized hardware acceleration. The compact model size and low inference latency make the system suitable for deployment on edge devices such as Raspberry Pi, Jetson Nano, and mobile platforms.

Metric	Value	Notes
Model Size	3.6 MB	PyTorch .pth model
Memory Footprint	150 MB	During inference (batch size = 1)
Inference Time (CPU)	45–120 ms	Single image prediction
Throughput	8–22 images/sec	Real-time capable

Table 6: Computational efficiency metrics

6 Discussion

6.1 Summary of Findings

The experimental results demonstrate that a lightweight three-layer CNN can achieve competitive performance for binary crop disease classification. The proposed model achieves an overall accuracy of approximately 90.5% while maintaining a compact architecture with fewer than one million parameters.

Unlike many deep learning approaches that rely on complex ensemble models or large pre-trained networks, the proposed architecture achieves reliable performance using a relatively simple design. The model is trained using only a balanced subset of 1,000 images while still generalizing effectively to unseen test samples. Validation accuracy remains stable within the range of 90–93% during training, indicating good convergence behavior.

6.2 Comparison with Related Work

Approach	Accuracy	Model Size	Inference Time	Deployment
ResNet50	92–94%	98 MB	200–300 ms	GPU-based systems
EfficientNetB3	91–93%	12 MB	80–150 ms	Edge-capable
Proposed CNN	90.50%	3.6 MB	45–120 ms	Edge deployment

Table 7: Comparison with representative deep learning architectures

Although the proposed model achieves slightly lower classification accuracy than large-scale architectures such as ResNet50 and EfficientNet, it provides significant advantages in terms of computational efficiency and model size. The reduced inference latency and compact model footprint make the proposed approach particularly suitable for edge-based agricultural monitoring systems.

6.3 Practical Applications

The lightweight design of the proposed model enables several practical deployment scenarios:

1. **Mobile Field Monitoring:** Farmers can capture leaf images using smartphones and obtain immediate disease predictions directly on the device.

2. **IoT Camera Networks:** Low-power IoT cameras can continuously monitor crop health and detect early signs of disease.

3. **Autonomous Agricultural Robots:** Robots used for spraying or weeding can incorporate the CNN model to detect diseased crops during field navigation.

4. **Early Warning Systems:** Automated monitoring platforms can identify early disease symptoms and trigger preventive interventions.

In addition to operational advantages, such systems may also provide economic benefits by reducing crop losses and minimizing excessive pesticide usage.

6.4 Limitations

Despite promising results, several limitations should be considered. First, the dataset used in this study includes only three crop categories (tomato, potato, and pepper), and further validation is required to confirm generalization to other crops. Second, the PlantVillage dataset largely consists of images captured under controlled conditions, which may differ from real-world agricultural environments. Third, the current model performs binary classification (healthy versus diseased) and does not identify specific disease types. Finally, the training dataset contains only 1,000 images, and larger datasets may further improve model robustness.

6.5 Future Research Directions

Several directions can be explored to extend this work. Future research may focus on multi-class classification to identify specific plant diseases, incorporating additional crop species to improve generalization, and evaluating model performance under real-world field conditions. Additional improvements could include attention mechanisms to highlight disease-affected regions of leaves, integration with temporal data to model disease progression, and combining visual information with multispectral or environmental sensor data for more comprehensive crop monitoring systems.

7 Conclusion

This paper presents a lightweight Convolutional Neural Network (CNN) architecture for real-time crop disease detection designed for deployment in precision agriculture environments. The proposed model prioritizes computational efficiency while maintaining competitive classification performance, enabling practical use on edge devices without requiring cloud infrastructure or specialized hardware.

Experimental evaluation demonstrates that the model achieves **90.50% classification accuracy** for binary crop disease detection using a balanced subset of the PlantVillage dataset. The trained network contains approximately 900,000 parameters and requires only **3.6 MB** of storage. In addition, the model achieves inference times between **45 ms and 120 ms** on standard CPU hardware, demonstrating suitability for real-time agricultural monitoring applications.

The results indicate that carefully designed lightweight CNN architectures can achieve reliable disease detection performance while maintaining low computational requirements. This balance between accuracy and efficiency is particularly important for agricultural environments where computational resources, connectivity, and specialized hardware may be limited.

Future work will focus on extending the proposed model to multi-class disease identification, expanding the dataset to include additional crop species and field conditions, and integrating the model with edge-based agricultural monitoring platforms for real-world deployment.

Overall, this work contributes toward practical AI-assisted crop monitoring systems that support early disease detection and improved agricultural management practices.

References

[1] FAO, *The State of Food and Agriculture 2019*. Food and Agriculture Organization of the United Nations, Rome, 2019.

[2] A. Ramcharan, E. Baranowski, P. McCloskey, et al., "Deep learning for image-based cassava disease detection," *Frontiers in Plant Science*, vol. 10, p. 1852, 2019.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 25, pp. 1097–1105, 2012.

[4] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in Neural Information Processing Systems*, vol. 27, pp. 3320–3328, 2014.

[5] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 97–105, 2013.

[6] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 6105–6114, 2019.

[7] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers in Plant Science*, vol. 7, p. 1419, 2016.

[8] A. G. Howard, M. Zhu, B. Chen, et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[9] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[10] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding," *arXiv preprint arXiv:1510.00149*, 2015.

Acknowledgments

The authors would like to acknowledge the creators of the PlantVillage dataset and the Kaggle community for providing publicly available crop disease datasets that enabled this research. The authors also thank the PyTorch and scikit-learn open-source development communities for providing the software frameworks used in this work. Institutional support from Shri Siddheshwar Women's Polytechnic, Solapur, is gratefully acknowledged. The authors further recognize the broader open-source agricultural AI community whose contributions make research in this domain possible.