

DEEP LEARNING FOR SUSTAINABLE AGRICULTURE: OPTIMIZED MOBILENETV2 FOR MULTI-CLASS CROP DISEASE IDENTIFICATION

Daisy Wadhwa

School of Engineering and Technology, CT University, Ludhiana, Punjab, India

Arvind Kumar

School of Engineering and Technology, CT University, Ludhiana, Punjab, India

Kamal Malik

Computer Science and Engineering, Maharishi Markandeshwar (Deemed to be University),
Mullana- Ambala, Haryana, India

ABSTRACT

The prevalence of crop diseases presents a major challenge to global food security and agricultural sustainability, causing significant yield losses and economic damage. Conventional disease detection methods, which rely on manual inspection, are often inaccurate, time-consuming, and impractical for large-scale implementation. While deep learning, especially Convolutional Neural Networks, has shown promise in automating disease classification, the practical deployment of these models is hindered by high computational demands. This study proposes an efficient MobileNetV2-based deep learning model tailored for classifying 36 healthy and unhealthy categories across 16 plant species. The dataset used in this research combines real-field and lab-controlled images from multiple public sources, enhancing the model's generalizability. The model was trained with six different optimizers, and Nadam was identified as the most effective, yielding 93.51% test accuracy. To further enhance performance, Optuna-based hyperparameter tuning was employed. The fine-tuned model attained 98.82% test accuracy, with precision, recall, and F1-score of 0.9882 and ROC AUC of 0.9999, reflecting a 5.68% improvement over the baseline model. The findings emphasize the feasibility of deploying a lightweight, high-performance model for real-time crop disease detection. By offering a scalable and computationally efficient approach, this study advances sustainable agriculture, enabling timely disease identification and improved crop management.

KEYWORDS:

Deep learning, convolutional neural networks, lightweight model, transfer learning, sustainable agriculture, plant disease

1. INTRODUCTION

Crop diseases induced by various factors like environmental conditions and pests, severely impact agricultural productivity, farmer livelihoods, national economies, and the overall global food security. According to the United Nations Food and Agriculture Organization (FAO), crop diseases cause up to 40% of global crop production losses annually, making them a critical barrier to achieving sustainable agriculture and meeting the nutritional demands of a growing global population [1]. These losses not only reduce farmer incomes but also affect national Gross Domestic Product (GDP), export potential, and food supply chains, emphasizing the pressing need for effective crop disease management strategies [2].

Timely detection and accurate diagnosis of crop diseases are essential to mitigate their adverse effects. Farmers usually employ visual analysis techniques based on their knowledge and experience to detect diseases. This method is subjective, labor-intensive, and prone to diagnostic errors, often resulting in delayed interventions. Such delays result in disease spread, increase crop losses, and necessitate greater reliance on chemical inputs, which can have long-term environmental consequences [3]. These limitations highlight the urgent need for innovative solutions that facilitate early, accurate, and efficient crop disease detection.

Artificial Intelligence (AI) technologies like Machine Learning (ML), and Deep Learning (DL), are increasingly being used for automating disease detection, including crop diseases [4-7].

Among these, Convolutional Neural Networks (CNNs) are widely being employed for their ability to extract hierarchical features from images, making them highly effective for plant disease classification [8-11]. Several studies have demonstrated that CNN-based models achieve high accuracy in identifying crop diseases, establishing them as a cornerstone of precision agriculture [12-16]. However, significant challenges remain in deploying automated disease detection systems in practical settings, leaving critical gaps in the literature.

First, most models are computationally heavy (e.g., VGG, ResNet), making them impractical for smallholder farms and edge devices. Although lightweight models such as MobileNetV2 and EfficientNet have been proposed, many studies remain restricted to single-crop disease detection [17-20]. For example, Pineda Medina et al. [21] applied MobileNetV2 for potato leaf disease detection and achieved high accuracy but focused only on three disease classes. Similarly, Rashid et al. [22] used MobileNetV3 for tomato classification in a mobile application, but its scope was limited to a single crop and lab-quality images.

Second, there is a generalizability gap in datasets. Most prior studies rely on lab-controlled datasets [23-25], which fail to capture the variability of real-field conditions, including background clutter, lighting variations, and overlapping leaves. Studies, such as Lu et al. [26] and Tambe et al. [27] that utilized MobileNetV2, emphasize the potential of lightweight architectures but also note challenges in robustness when tested in diverse, uncontrolled field environments.

Third, while optimization techniques significantly affect CNN performance, their systematic use remains limited. For instance, Dogo et al. [28] observed Nadam's superiority in convergence for CNNs, and Bhuyian et al. [29] showed that Bayesian optimization improves performance for lightweight models. Yet, automated hyperparameter tuning frameworks are underexplored in multi-crop disease classification tasks, where balancing performance and efficiency is critical.

1.1. Research Hypothesis

Based on these gaps, this study hypothesizes that:

H_0 (Null Hypothesis): A lightweight CNN model, fine-tuned with automated hyperparameter optimization, does not outperform baseline CNN models in terms of classification accuracy and generalizability across multiple crop species.

H_1 (Alternative Hypothesis): A lightweight CNN model, fine-tuned with automated hyperparameter optimization, outperforms baseline CNN models in terms of classification accuracy and generalizability across multiple crop species.

1.2. Research objectives

This study is guided by the following objectives:

1. To develop a lightweight CNN model using transfer learning with MobileNetV2 for multi-crop disease classification.
2. To systematically evaluate optimizers and identify the most effective one for model training through comparative experiments.
3. To apply automated hyperparameter tuning to optimize hyperparameters for improved performance.
4. To validate the proposed model on a comprehensive dataset combining lab-controlled and real-field images across 36 classes and 16 crop species.

1.3. Contributions of the study

Unlike earlier MobileNetV2-based studies [21-22], which are constrained by single-crop focus, limited optimization, or exclusive reliance on lab datasets, this study makes the following contributions:

- It proposes a generalized lightweight model capable of classifying diseases across multiple crops, moving beyond single-crop limitations.
- It integrates automated hyperparameter tuning (Optuna) with MobileNetV2, an approach underexplored in multi-crop disease detection.
- It evaluates multiple optimizers under controlled experiments, statistically validating performance differences.
- It utilizes a diverse dataset that combines real-field and lab-controlled images, thereby enhancing robustness in practical agricultural settings.

1.4. Organization of the paper

The subsequent sections of this paper are structured as follows. Section 2 provides a detailed review of the literature on deep learning applications in crop disease detection. Section 3 describes the methodology employed to develop the proposed model. Section 4 presents the results and discusses the findings in relation to existing studies. Finally, Section 5 concludes the study and offers directions for future research.

2. REVIEW OF LITERATURE

Deep learning has significantly advanced the automatic identification and classification of crop diseases. This section reviews the recent literature, organized into four themes: lightweight architectures for crop disease detection, transfer learning and multi-crop generalization, optimization and automated hyperparameter tuning, and dataset considerations. The review emphasizes the need of compact, generalizable, and field-deployable models.

2.1. Lightweight architectures for crop disease detection

Early works in the area of crop disease classification such as Mohanty et al. [30] and Ferentinos [15] employed deep CNNs like GoogLeNet and VGG, achieving high accuracies (>99%) on PlantVillage [31] dataset. However, these models suffered from high computational costs and poor generalization to real-field conditions. To address these issues, lightweight CNNs have emerged.

MobileNet and its successors dominate this space. Fan and Guan [32] and Ullah et al. [33] demonstrated MobileNetV2 and ShuffleNetV2 models exceeding 98% accuracy for tomato and corn diseases. Duhan et al. [34] proposed RTR_Lite-MobileNetV2, specifically designed

for low-resource edge deployment, achieving comparable performance while reducing Floating Point Operations Per Second (FLOPs). Similarly, Rashid et al. [22] developed a MobileNetV3 variant for tomato leaf disease detection, emphasizing real-time applicability.

Other variants include EfficientNet and SqueezeNet derivatives. Ali et al. [35] presented an ensemble of EfficientNet and ResNet models, reporting >99% accuracy across multiple crops, but at the cost of parameter growth. By contrast, Lu et al. [26] proposed MobileNetV2 modifications that reduced FLOPs while maintaining >99% accuracy. These developments highlight the centrality of lightweight CNNs in balancing accuracy with resource constraints.

2.2. Transfer learning and multi-crop generalization

Transfer learning has been pivotal in crop disease identification. Pineda Medina et al. [21] and Kini et al. [36] leveraged pretrained CNNs for feature extraction, achieving high accuracy (>97%) across multiple disease classes. However, these approaches are often single-crop oriented.

Recent studies have begun exploring cross-crop and multi-crop generalization. Tambe et al. [27] developed a MobileNetV2 based model for classifying diseases in multiple crops, achieving an accuracy of 91.98%. Ashurov et al. [37] highlighted transfer learning strategies across multiple crop species. These findings align with emerging trends that emphasize not just accuracy but cross-species adaptability, which is essential for practical agricultural adoption.

2.3. Optimization and automatic hyperparameter tuning

Optimizers and hyperparameter tuning significantly influence CNN performance. While Adam optimizer [38] is widely used, studies stress the importance of context-specific selection. Dogo et al. [28] found Nadam superior in convergence and accuracy, while Saleem et al. [39] highlighted the synergy between optimizers and model types, with Adam-Xception pairing yielding a 0.9978 F1-score.

More recently, automated methods have been deployed. Ashwinkumar et al. [40] and Bhuyian et al. [29] applied Bayesian optimization for lightweight models for tomato and banana disease detection, achieving 96-98% accuracy. These findings reinforce the importance of the current study's integration of Optuna-based tuning in achieving state-of-the-art performance.

2.4. Datasets and real-field challenges

Dataset choice strongly influences model performance. Early reliance on controlled datasets (e.g., PlantVillage) yielded inflated accuracies, but failed in uncontrolled field environments [15], [30]. To address this, field-based datasets such as PlantDoc [41] and custom multi-crop datasets have gained traction.

Salka et al. [42] surveyed field dataset challenges, including varying lighting, occlusion, and mixed symptoms. Ferdi [43] emphasized the benefit of background-removal augmentation to reduce noise. Still, most lightweight models remain evaluated on single crops or lab settings [21], [27].

This study uniquely addresses this limitation by combining both real-field and controlled images across 16 crops and 36 classes. Unlike studies like [27], [44], which focus exclusively on lab datasets, this study integrates diverse data sources to maximize robustness.

2.5.Comparative analysis of existing lightweight models

To contextualize the proposed approach, **Table 1** provides a comparative summary of representative lightweight models reported in recent studies, highlighting differences in crops, dataset scale, number of classes, model complexity (parameters and FLOPs), and performance.

Table 1. Comparative analysis of existing lightweight CNN models for plant disease detection

Study	Model	Crop s	Datase t Size	Cla sses	Paramet ers (Millions)	FLOPs (Millions)	Accura cy (%)	Remarks
Duhan et al. [34]	Mobil eNetV 2	Multi ple crops	61,486 images	39	1.05	526	99.92	Achieved the highest accuracy with augmentation; precision, recall, and F1 > 99.85%.
Pineda Medina et al. [21]	Mobil eNetV 2	Potato	3,000 images	3	2.34	–	98.70	Effective in real-time applications with small dataset.
Tambe et al. [27]	Mobil eNetV 2	Multi ple crops	87,867 images	38	–	–	91.98	Struggles in multi-leaf, crop-specific disease identification.
Lu et al. [26]	Mobil eNetV 2	Multi ple crops	30,644 images	25	0.91	268	99.53	High accuracy under controlled settings; limited field applicability.
Zhu and Gao [45]	Shuffl eNetV 2	Maize	10,845 images	6	0.87	1.75	99.86	High accuracy but limited to maize; narrow disease range.
Ullah et al. [33]	Custo m CNN	Toma to	10,000 images	10	0.69	–	99.34	Extremely lightweight, but limited by lab-captured dataset.

As shown in Table 1, MobileNetV2 is a widely adopted lightweight backbone. Duhan et al. [34] achieved an exceptional accuracy of 99.92% on a large multi-crop dataset, demonstrating that MobileNetV2, when carefully augmented, can outperform heavier CNNs while maintaining only 1.05M parameters. Similarly, Lu et al. [26] reported strong results (99.53%) across 25 PlantVillage classes, though this performance was obtained under controlled conditions, limiting its field applicability. Likewise, Tambe et al. [27] highlighted

the challenge of handling complex multi-leaf images as their model was trained using lab-controlled images with a uniform background.

In a crop-specific context, Pineda Medina et al. [21] demonstrated the suitability of MobileNetV2 for potato disease detection, achieving 98.7% accuracy with just 2.34M parameters, making it highly suitable for real-time use.

Beyond MobileNet, other lightweight models have also been explored. Zhu and Gao [45] developed a ShuffleNetV2 model for maize disease classification, achieving 99.86% accuracy with only 0.87M parameters and 1.75M FLOPs, but its narrow focus on six maize diseases limited generalizability. Ullah et al. [33] proposed a custom lightweight CNN with just 0.69M parameters, reaching 99.34% accuracy for tomato leaf diseases, though its reliance on lab-captured images reduces its robustness in real-world conditions.

Overall, these comparisons highlight two consistent trends. First, MobileNetV2 is a dominant and versatile lightweight backbone, but its performance depends heavily on dataset diversity and robustness to real-field variability. Second, lightweight models such as ShuffleNetV2 show impressive compactness, but their generalizability across crops and disease types remains constrained in the current body of research.

2.6. Research gaps

Based on the review of the literature, the following research gaps have been identified. Most lightweight models focus on single-crop classification, limiting their usefulness for farmers managing diverse crop species [32], [46]. Optimization techniques, though effective for single crops, are rarely adapted for the added complexity of multi-crop classification, where both accuracy and efficiency must be balanced [29], [47]. Additionally, many models are trained on lab-controlled datasets, reducing their robustness in real-field conditions where diseases vary with environment and crop type [23], [27], [48]. Transfer learning remains underutilized for multi-species classification, with challenges in preserving both performance and compactness.

Together, these gaps highlight the need for research that moves beyond narrow single-crop models, integrates both lab and real-field data, adopts systematic optimization techniques, and validates improvements through rigorous statistical analysis. Addressing these gaps is essential for advancing crop disease detection models that are both accurate and practically deployable in resource-constrained agricultural settings.

3. MATERIALS AND METHODS

This section outlines the methodology followed to develop a lightweight deep learning model for multi-crop disease classification. It covers dataset preparation, model training using multiple optimizers, automated hyperparameter tuning, and performance evaluation. An overview of the process is illustrated in **Figure 1**.

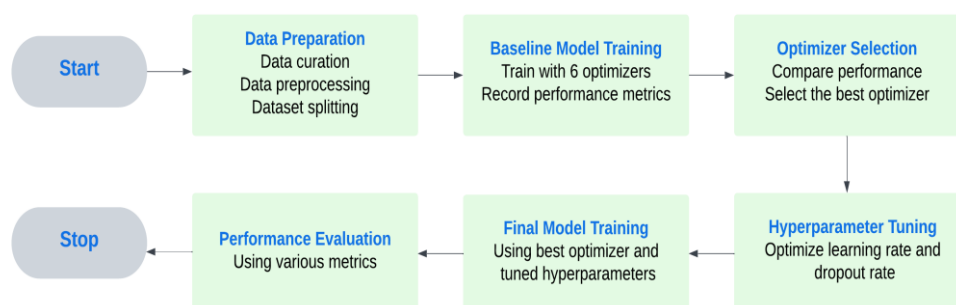


Figure 1. Overview of research methodology

3.1.Computational environment

All experiments were conducted in Python (version 3.11.7) using the TensorFlow deep learning framework, with supporting libraries including NumPy, Pandas, Matplotlib, and Scikit-learn. Training was executed on a workstation equipped with an NVIDIA RTX 3050 GPU (6 GB VRAM), 13th Gen Intel(R) Core (TM) i7-13700HX (2.10 GHz) (16 cores, 2.1 GHz), and 16 GB RAM, running a 64-bit Windows 11 operating system with CUDA 11.8 for GPU acceleration.

3.2.Dataset curation

To support generalization across crop types and environmental conditions, a diverse dataset was created by combining images from five publicly available sources: PlantVillage [31], PlantDoc [41], Bean Leaf Diseases Dataset [49], Coffee Leaf Diseases Dataset [50], and Banana Leaf Diseases Dataset [51]. These datasets represent a mix of real-field and lab-controlled images across 16 plant species and 36 healthy/diseased classes. The final curated dataset comprised 56,044 images, split into training (39,214), validation (7,843), and testing (16,830) subsets using a stratified 70-30 split, followed by an 80-20 training-validation split. Sample images from the dataset are shown in **Figure 2**.

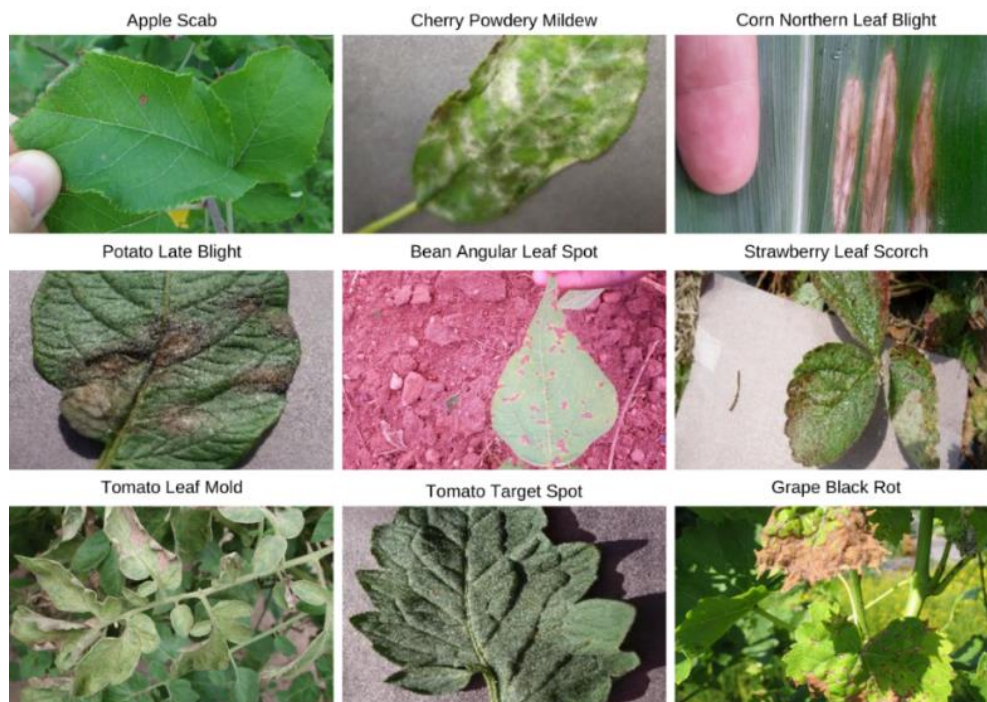


Figure 2. A sample of images from the combined dataset

3.3.Data preprocessing and class imbalance handling

The curated dataset exhibited class imbalance, with some disease classes significantly underrepresented. To address this, data augmentation was applied to minority classes using techniques such as random rotations, flips, shifts, zooming, and shear transformations. This increased sample size and also improved the model's robustness to real-world variations. Additionally, class weights were computed based on inverse class frequency to ensure fair learning across all categories. All images were resized to 224×224 pixels and normalized to a $[0, 1]$ scale to meet the input requirements of the proposed model.

3.4. Model architecture

MobileNetV2 was selected as the backbone model for its strong balance between accuracy and efficiency, making it suitable for deployment in resource-constrained agricultural environments. Unlike other lightweight models, MobileNetV2 integrates features such as inverted residuals and linear bottlenecks, offering an optimal trade-off between performance and complexity. To contextualize this choice, **Table 2** provides a comparison with other widely used lightweight architectures, including MobileNetV1, EfficientNetB0, NASNetMobile, ShuffleNet, and SqueezeNet. The comparison includes parameter count, computational complexity (FLOPs), and Top-1 ImageNet accuracy, which together highlight the trade-offs between efficiency and predictive performance. As shown, MobileNetV2 achieves higher accuracy than most models of comparable size and complexity, reinforcing its suitability as the foundation for the present study.

Table 2. Comparison of MobileNetV2 with other lightweight models

Model	Parameter count	FLOPs (Millions)	Top-1 Accuracy on ImageNet	Key features	Suitability for this study
MobileNetV2	~3.4M	~300	~ 71.8%	Inverted residuals, linear bottlenecks	High efficiency and accuracy for multi-class classification
MobileNetV1	~4.2M	~569	~ 70.9%	Depthwise separable convolutions	Lacks advanced features of MobileNetV2
EfficientNetB0	~5.3M	~390	~ 76.7%	Compound scaling	Higher computational cost
NASNetMobile	~5.3M	~564	~ 74.0%	Neural architecture search	Complex and less efficient
ShuffleNet	~1.4M	~150	~ 69.4%	Group convolutions, channel shuffling	Lower accuracy on diverse datasets
SqueezeNet	~1.2M	~831	~ 57.5%	Fire modules	Compact but insufficient for complex tasks

To tailor MobileNetV2 for the 36-class crop disease classification task, a customized classification head was added. The original fully connected layers were replaced with a global average pooling layer, followed by a dropout layer (rate = 0.2) and a dense softmax layer for multi-class output. The base MobileNetV2, initialized with ImageNet weights, was frozen during initial training to retain its learned features. The customized model architecture is illustrated in **Figure 3**.

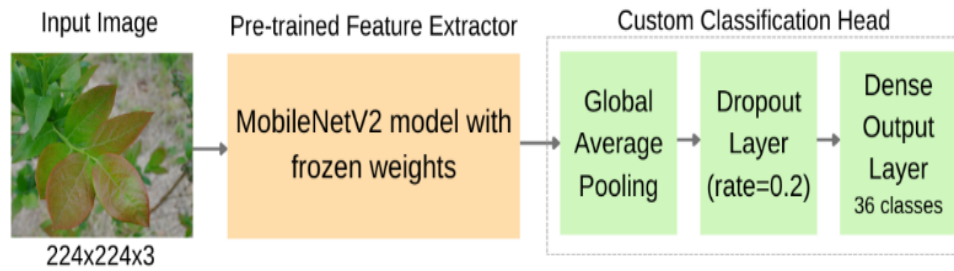


Figure 3. Customized baseline MobileNetV2 model architecture

3.5.Optimizer selection

To ensure optimal performance of the proposed model, the MobileNetV2 architecture was trained using six different optimizers: Adam, Stochastic Gradient Descent (SGD), RMSprop, Adagrad, Nadam, and Adamax. These optimizers were chosen for their established effectiveness in deep learning applications and their diverse approaches to gradient-based optimization. Each optimizer was applied using its default learning rate settings to maintain consistency, and the model was trained with the configuration setting (epochs=30, steps per epoch=300, validation steps=100). For each trained baseline model, performance metrics were recorded, providing a basis for comparing the effectiveness of the different optimizers.

The need for comparing these optimizers arises from the fact that their performance can vary significantly based on the nature of the dataset and the architecture of the model. Optimizers not only influence convergence speed but also impact the ability of the model to generalize well to unseen data. In the context of this study, selecting the right optimizer was particularly critical, as the classification task involved a diverse dataset of crop disease images, requiring high accuracy, precision, recall, and generalization capacity.

By systematically evaluating the performance of the model trained with each optimizer over the specified training conditions, the most suitable optimizer for the study was identified based on a range of evaluation metrics.

3.6.Automatic hyperparameter tuning

Following optimizer selection, key hyperparameters, dropout rate and learning rate, were tuned using Optuna [52], an efficient hyperparameter optimization framework. Automatic tuning was chosen over manual or grid search methods to reduce bias and improve efficiency. In this study, dropout rate and learning rate were targeted, as these are critical factors influencing model generalization, convergence stability, and training dynamics.

Optuna was configured to maximize the weighted F1-score on the validation set as the objective function, ensuring that improvements were not limited to accuracy alone but extended to balanced classification performance across all classes. For each trial, hyperparameter values were sampled by Optuna within the ranges summarized in **Table 3**.

Table 3. Search space for automatic hyperparameter tuning

Parameter	Range	Sampling Strategy
Dropout rate	0.1 to 0.9 (step=0.1)	Uniform
Learning rate	10^{-6} to 10^{-2}	Log Uniform

The learning rate was sampled on a logarithmic scale between $1e^{-6}$ and $1e^{-2}$, while dropout rate was sampled uniformly between 0.1 and 0.9 in increments of 0.1. The Optuna hyperparameter optimization study was performed with a fixed random seed of 42 to control trial sampling. A total of 30 trials were executed. Each trial consisted of the following steps:

1. Data was split into training and validation sets (80:20, stratified).
2. Image augmentation (rotation, shifting, shearing, zooming, flipping) was applied to improve robustness.
3. A MobileNetV2 model, fine-tuned with a Nadam optimizer, was trained with the candidate hyperparameters.
4. Training incorporated early stopping (patience = 20) and ReduceLROnPlateau (factor = 0.5, patience = 3, minimum learning rate = $1e^{-6}$) to prevent overfitting and improve convergence.
5. Class weights were computed to mitigate imbalance across crop disease classes.
6. The model was evaluated on the validation set, and the weighted F1-score was returned to Optuna as the performance metric.

Bayesian optimization with pruning was employed by Optuna, and poorly performing trials were terminated early to reduce computation. After completing 30 trials, the best-performing configuration was selected based on the highest validation weighted F1-score.

3.7. Model development and performance evaluation

After selecting the best-performing optimizer and identifying the optimal hyperparameters (learning rate and dropout rate), the MobileNetV2 model was trained with these configurations to achieve optimal performance. The model was trained for 50 epochs with 300 steps per epoch for the training set and 100 validation steps for the validation set. Early stopping was implemented to prevent overfitting and ensure the best model weights were retained. Additionally, a learning rate reduction mechanism was incorporated. Class weights were calculated and applied during training to address the class imbalance in the dataset, ensuring that minority classes contributed proportionately to the model's learning process.

To comprehensively assess the performance of the proposed model and the baseline models created using different optimizers, a diverse set of evaluation metrics was employed on the test dataset. Since the dataset exhibited class imbalance, weighted precision, weighted recall, and weighted F1-score were calculated along with the accuracy metric. The equations of these metrics are listed in **Table 4**.

Table 4. Metrics for performance evaluation

Metric	Equation	Description
Accuracy	$\frac{TP + TN}{TP + FP + TN + FN}$	TP, TN, FP, and FN denote True Positives, True Negatives, False Positives, and False Negatives, respectively
Weighted Precision	$P_w = \frac{\sum_{i=1}^C N_i \cdot P_i}{\sum_{i=1}^C N_i}$	N_i is the number of instances in class i , P_i is the precision for class i , and C is the total number of categories
Weighted Recall	$R_w = \frac{\sum_{i=1}^C N_i \cdot R_i}{\sum_{i=1}^C N_i}$	R_i is the recall for class i .
Weighted F1-score	$F1_w = \frac{\sum_{i=1}^C N_i \cdot F1_i}{\sum_{i=1}^C N_i}$ $= \frac{2 \cdot P_i \cdot R_i}{P_i + R_i}$	

In addition to these metrics, the overall weighted Receiver Operating Characteristic- Area Under Curve (ROC AUC) score was calculated using the one-vs-rest (OVR) method, as per the multi-class classification nature of the current study. This metric evaluates the model's ability to distinguish between classes. A deeper analysis of the model's performance was conducted by examining the confusion matrix. Additionally, the performance of the baseline models and the fine-tuned proposed model was statistically tested.

3.8 Statistical Analysis

To rigorously evaluate model performance, multiple statistical tests were incorporated into the experimental design. The goal was twofold:

1. To examine whether the choice of optimizer significantly influenced the baseline MobileNetV2 performance
2. To validate whether hyperparameter tuning with Optuna led to statistically significant improvements over the baseline models.

For the baseline analysis, MobileNetV2 model with each of the six optimizers was trained with different random seeds (42, 123, 999). This repeated-seed design has been recommended in prior deep learning studies to account for stochastic variations during training [53-54]. The resulting test accuracies were analyzed using a one-way repeated measures ANOVA to determine whether performance differences among optimizers were statistically significant. Because ANOVA assumes normality and equal variances, the non-parametric Friedman test was additionally applied, as suggested in comparative deep learning research [55], to provide a distribution-free confirmation.

Following the baseline evaluation, the tuned proposed model- optimized using Optuna for dropout rate and learning rate- was compared against its baseline counterpart. Statistical validation was performed using a paired t-test. To quantify the magnitude of improvements, Cohen's d was calculated as an effect size measure, where values above 0.8 indicate large effects [56]. Additionally, 95% confidence intervals (CIs) for the mean difference in accuracy were computed to provide an interval estimate of the performance gain.

4. Results and Discussion

This section presents a comprehensive evaluation of the proposed fine-tuned MobileNetV2 model, including its classification performance, computational efficiency, and a comparative analysis with existing lightweight models and prior studies in the literature.

4.1. Optimizer comparison results

The baseline MobileNetV2 model was trained using six different optimizers- Adam, SGD, RMSprop, Adagrad, Adamax, and Nadam- and their performances were compared based on key evaluation metrics such as test accuracy, weighted precision, weighted recall, weighted F1-score, and overall ROC AUC. The results are summarized in **Table 5**, which presents a comprehensive comparison of the optimizers' effectiveness in training the model. The best performing optimizer is highlighted in boldface in Table 5.

Table 5. Baseline MobileNetV2 model performance comparison using different optimizers

Optimizer	Test Accuracy	Weighted Precision	Weighted Recall	Weighted F1-score	Overall ROC AUC
Adam	92.43%	0.9149	0.9219	0.9146	0.9979
SGD	91.72%	0.9038	0.9161	0.9059	0.9974

RMSprop	92.46%	0.9125	0.9189	0.9127	0.9973
Adagrad	83.31%	0.8159	0.8195	0.8101	0.9908
Adamax	92.37%	0.9102	0.9216	0.9133	0.9976
Nadam	93.51%	0.9231	0.9333	0.9257	0.9982

From the results in Table 5, Nadam emerged as the best-performing optimizer, achieving the highest test accuracy (93.51%), weighted precision (0.9231), weighted recall (0.9333), and weighted F1-score (0.9257). Additionally, its overall ROC AUC of 0.9982 further highlights its superior performance in distinguishing between classes across the test dataset.

The superior performance of Nadam can be attributed to its incorporation of Nesterov momentum on top of Adam's adaptive gradient methodology. Nesterov momentum predicts the future position of the gradient and adjusts accordingly, which results in smoother and faster convergence. This capability is particularly advantageous in complex tasks like multi-class crop disease classification, where the loss landscape can be non-convex and noisy. Nadam's ability to maintain stability while achieving faster convergence is evident in its higher accuracy.

In contrast, while optimizers like Adam and RMSprop performed well, they lacked the additional momentum enhancements provided by Nadam, which likely explains their marginally lower performance. SGD, while effective in some cases, showed slower convergence due to its lack of adaptiveness, evident in its slightly lower accuracy (91.72%). Adagrad, which adapts learning rates individually for parameters, underperformed significantly (test accuracy: 83.31%), likely due to its tendency to diminish learning rates excessively during training, leading to suboptimal convergence.

The results also highlight that Adamax closely followed Nadam in performance but fell slightly short in terms of test accuracy and weighted F1-score. This indicates its effectiveness but suggests a limitation in convergence speed compared to Nadam. Nadam's superior handling of the optimization process justifies its selection for further training of the MobileNetV2 model.

4.2.Results of automatic hyperparameter tuning

The hyperparameters, learning rate and dropout rate, were optimized using Optuna, which identified the best values as a learning rate of 0.0001837 and a dropout probability of 0.2. The model was trained using this best found hyperparameter values. The model's performance before and after hyperparameter tuning is detailed in **Table 6**, along with the percentage change in each evaluation metric.

Table 6. Model performance before and after hyperparameter tuning

Metric	Before tuning	After tuning	Percentage change (%)
Test Accuracy	93.51%	98.82%	+5.68
Test Loss	0.2234	0.0442	-80.21
Weighted Precision	0.9231	0.9883	+7.06
Weighted Recall	0.9333	0.9882	+5.88
Weighted F1 Score	0.9257	0.9882	+6.75
Overall ROC AUC	0.9982	0.9999	+0.17

It is apparent from Table 6 that hyperparameter tuning resulted in a significant improvement in model performance across all metrics. The test accuracy increased from 93.51% to 98.82%, representing a substantial 5.68% enhancement. Similarly, the test loss dropped dramatically by 80.21%, indicating more stable and accurate predictions with reduced errors.

Improvements in weighted precision, recall, and F1 score highlight enhanced class-wise prediction capability, reducing misclassification across all categories. The slight increase in overall ROC AUC (0.17%) suggests near-perfect separability between classes, which is crucial for multi-class classification tasks.

The optimized learning rate and dropout probability played pivotal roles in achieving these improvements. The fine-tuned learning rate ensured smoother and more precise gradient updates, allowing the model to converge effectively without overshooting the minima. The optimized dropout rate balanced regularization and overfitting control, particularly important for the diverse and complex crop disease dataset used in this study.

The substantial improvements further validate the efficacy of automatic hyperparameter optimization techniques like Optuna, which streamline the search for ideal parameters compared to manual tuning or grid search. By systematically exploring the parameter space, Optuna identified hyperparameters that maximized performance while maintaining computational efficiency. To validate the performance improvements of the proposed model, statistical analysis was conducted, the results of which are presented in the next section.

4.3. Results of statistical analysis

To evaluate whether the choice of optimizer had a statistically significant effect on model performance, test accuracies of baseline MobileNetV2 models trained with six different optimizers were compared across three independent runs for each optimizer. The mean and standard deviation of test accuracies are summarized in **Table 7**. Among all optimizers, Nadam achieved the highest average accuracy (90.58%), followed closely by SGD (88.20%), while Adagrad produced the lowest performance (78.54%).

Table 7. Performance of MobileNetV2 models across six optimizers

Optimizer	Test Accuracy (%)				Standard Deviation
	Seed 42	Seed 123	Seed 999	Mean Accuracy	
Adagrad	83.26	76.19	76.17	78.54	0.0334
Adamax	91.85	88.19	57.75	79.26	0.1529
RMSprop	92.54	89.07	75.39	85.67	0.0740
Adam	93.42	83.97	83.87	87.09	0.0448
SGD	91.40	88.24	84.96	88.20	0.0263
Nadam	93.80	89.95	88.00	90.58	0.0241

A one-way repeated measures ANOVA was conducted to assess whether these observed differences were statistically significant. The results indicated no significant differences among optimizers, $F(5,12) = 0.87$, $p = 0.528$. To further validate this finding, the non-parametric Friedman test was applied, yielding $\chi^2 = 9.48$ with $p = 0.092$, which similarly failed to demonstrate statistical significance. Thus, as shown in Table 7, although Nadam outperformed the other optimizers on average, the improvements were not statistically significant. This outcome is likely influenced by the small sample size of three runs per optimizer, which limits the power of the statistical tests. Future work could incorporate a larger number of independent runs to provide higher statistical power and more robust conclusions.

Having established Nadam as the best-performing optimizer in terms of mean accuracy, the next stage of analysis compared the baseline Nadam model against its tuned counterpart obtained through Optuna-based hyperparameter optimization. The test results from three

independent runs with different seeds (42, 123, 999) are presented in **Table 8**. The tuned Nadam model consistently outperformed the baseline, with accuracies exceeding 98.74% across all runs.

Table 8. Performance comparison of baseline vs. tuned Nadam models (mean $\pm$ standard deviation across three runs)

Model	Test Accuracy (%)
Baseline Nadam	90.58 $\pm$ 0.030
Tuned Nadam (Optuna)	98.74 $\pm$ 0.001

To statistically validate these improvements, paired *t*-tests were applied. The paired *t*-test demonstrated that the tuned Nadam model significantly outperformed the baseline ($t = 5.04$, $p = 0.037$), with a 95% confidence interval for the difference of [0.01, 0.15]. This suggests that the accuracy improvements introduced by hyperparameter optimization are statistically meaningful. Importantly, the effect size estimated using Cohen's *d* was 2.91, which corresponds to an extremely large effect, reinforcing the practical importance of the observed improvements.

These findings support rejection of the null hypothesis (H_0) and provide evidence in favor of the alternative hypothesis (H_1), namely that automated hyperparameter optimization improves classification accuracy and generalizability across multiple crop species. The statistically significant improvements suggest that farmers and extension workers could rely on the model to provide more consistent diagnostic outputs, even when deployed on diverse data sources.

4.4. Proposed model performance analysis

In order to obtain deeper insights into the tuned-model performance, its accuracy and loss plots, and confusion matrix were analyzed.

The accuracy and loss curves of the proposed model demonstrate consistent performance across training and validation phases (**Figure 4**). The training and validation accuracy steadily improved during the initial epochs, surpassing 90% within the first 10 epochs. Both training and validation accuracy converge close to 99% towards the later epochs, indicating a high-performing model. The training and validation loss exhibited a rapid decline during the early epochs, stabilizing at minimal values (below 0.05) after approximately 20 epochs. Notably, the validation loss closely follows the training loss curve, suggesting no overfitting or underfitting issues. This minimal gap between training and validation accuracy/loss throughout the training process highlights the model's ability to generalize well to unseen data.

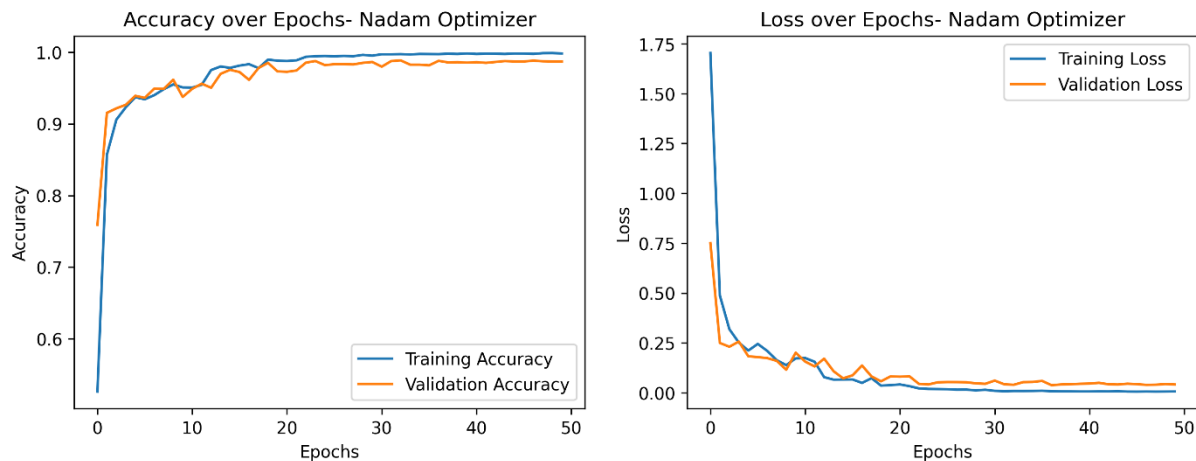


Figure 4. Accuracy and loss plots of the proposed model

The proposed model's confusion matrix offers a thorough analysis of the classification performance of the model (**Figure 5**). The concentration of predictions along the diagonal reflects the model's strong generalization capability and its ability to discriminate between distinct classes effectively. This is particularly notable given the diversity of the dataset, which includes healthy and diseased leaves from multiple crops with varying visual characteristics.

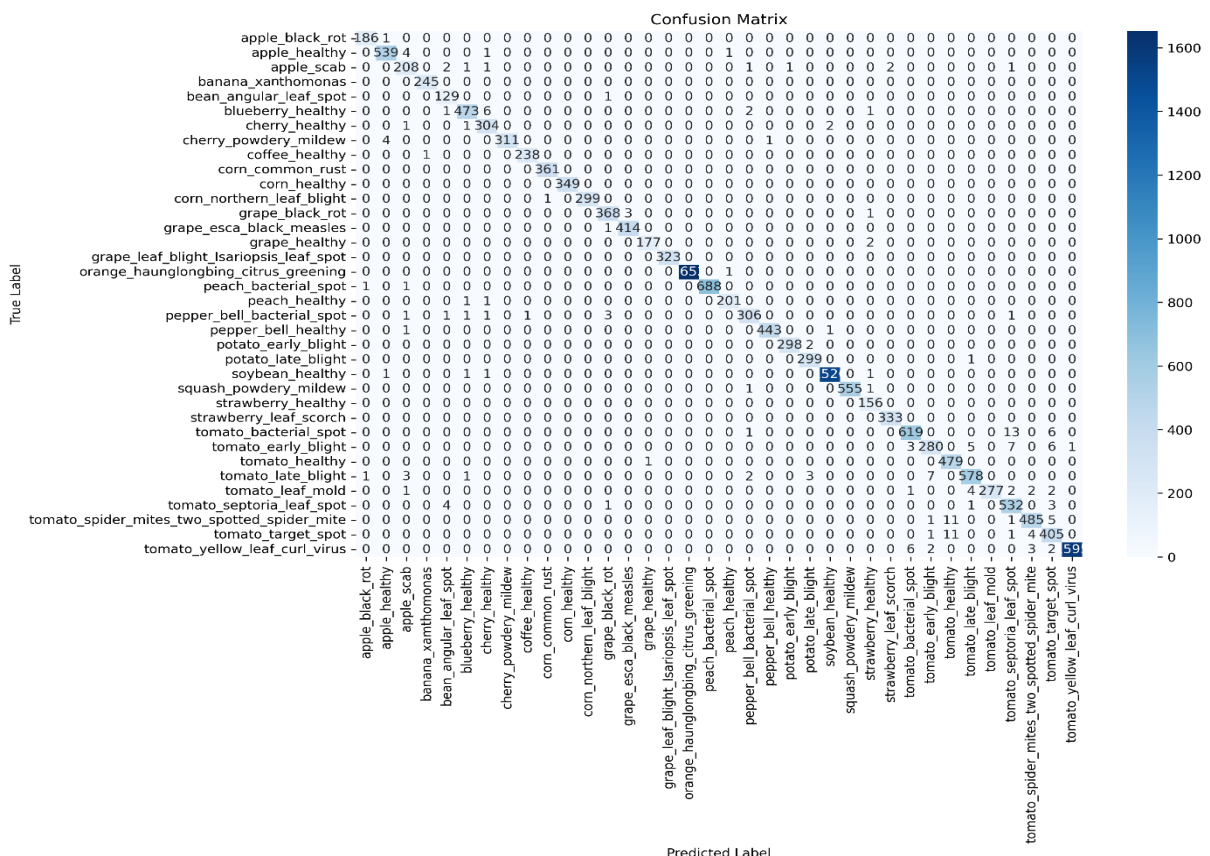


Figure 5. Confusion matrix of the proposed model

For most classes, including high-support categories such as Orange Haunglongbing, Soybean Healthy, and Tomato Yellow Leaf Curl Virus, the model achieved nearly perfect predictions, with minimal or no misclassifications. Classes with lower support, such as Bean Angular

Leaf Spot and Strawberry Healthy, also exhibited strong performance, with very few misclassifications observed.

The off-diagonal misclassifications are sparse and primarily limited to visually similar classes or classes with shared morphological features. For instance, Tomato Early Blight and Tomato Late Blight show minor confusion, as a few instances of the former were misclassified as the latter. Similarly, some misclassifications occurred between Apple Scab and Apple Black Rot, likely due to overlapping symptoms such as leaf discoloration.

The model's ability to handle high-support and low-support classes with consistent accuracy demonstrates the efficacy of the Nadam optimizer and the tuned hyperparameters. These elements contributed to the balanced learning of features across classes, preventing overfitting to dominant categories and underrepresentation of minority ones.

This level of class discrimination is essential in practice, as visually similar diseases often require distinct treatments; hence, misclassification could lead to costly or ineffective interventions.

4.5. Computational efficiency of the proposed model

The computational efficiency of the proposed model demonstrates significant improvements in both parameter count and FLOPs. Specifically, the proposed model has approximately 2.3 million trainable parameters and requires ~599 million FLOPs for inference. A comparative analysis of these values against other well-known lightweight models, presented in **Table 9**, reveal the following insights.

Table 9. Computational efficiency of proposed model versus other lightweight models

Model	Parameter count	FLOPS (Millions)
MobileNetV2	~3.4M	~300
MobileNetV1	~4.2M	~569
EfficientNetB0	~5.3M	~390
NASNetMobile	~5.3M	~564
ShuffleNet	~1.4M	~150
SqueezeNet	~1.2M	~831
Proposed Model	~2.3M	~599

Compared to MobileNetV1, the proposed model reduces parameter count by ~45% while requiring slightly more FLOPs. It is also more parameter-efficient than EfficientNetB0 and NASNetMobile (~57% fewer parameters), both of which are computationally demanding.

Despite having higher FLOPs than MobileNetV2, the proposed model outperforms it in classification performance. The efficiency of this MobileNetV2-based model, enhanced through optimal hyperparameters and the Nadam optimizer, makes it suitable for resource-constrained environments.

While ShuffleNet and SqueezeNet have lower parameter counts, their simplified architectures often sacrifice accuracy, especially for complex datasets. The proposed model bridges this gap by offering a strong balance between lightweight design and high performance.

Though its FLOPs (~599M) are comparable to MobileNetV1 and NASNetMobile, this increase is a result of intentional architectural and training improvements.

These enhancements, including optimizer choice and hyperparameter tuning, contribute to the model's superior accuracy, justifying the computational cost. It further highlights that the

model can be integrated into smartphone-based tools or IoT devices for on-field disease monitoring, making precision agriculture more accessible to smallholder farmers.

4.6.Comparative analysis with existing models in literature

The proposed MobileNetV2-based model achieved a classification accuracy of 98.82%, outperforming existing models in literature when considering both model performance and dataset characteristics. A comparison of the proposed model with other recent studies is presented in the **Table 10**.

Table 10. Comparison between proposed model and existing crop disease multi-classifiers

Study	Model Architecture	Crop Species	Classes	Dataset Size	Performance (Accuracy)	Dataset Characteristics
Y. Liu et al. [57]	SqueezeNext	10	59	36,258	91.94%	Lab-setting images
Tambe et al. [27]	MobileNetV2	14	38	87,867	91.98%	Lab-setting images
Macdonald et al. [58]	Custom CNN	14	38	54,306	96.75%	Lab-setting images
Barman et al. [44]	EfficientNet B0	3	7	7000	97.29%	Lab-setting images
Proposed Model	Fine-tuned MobileNetV2	16	36	56,044	98.82%	Both real-world and lab-setting images

From the comparison with the existing studies, it is apparent that the proposed model demonstrates an advantage over existing lightweight crop diseases multi-classifiers. This high performance stems from the model's carefully fine-tuned architecture, the use of the Nadam optimizer, and the hyperparameter optimization. In contrast, other studies, such as those employing MobileNetV2 [27] reported lower accuracies, despite using lab-controlled datasets. Y. Liu et al. [57] employed SqueezeNext to classify 59 classes across 10 crop species, however, its accuracy was limited to 91.94% due to the inherent simplicity of the model and the sole reliance on lab-setting images. Likewise, Macdonald et al. [58] employed a custom CNN achieving higher accuracy than Y. Liu et al. [57] but still falling short of the proposed model's performance. Notably, Barman et al. [44] achieved a high accuracy of 97.29% with EfficientNetB0. However, this study was focused on a much smaller dataset of just 7,000 images and included only three crop species and seven classes. While this study highlights EfficientNetB0's capability for smaller-scale problems, its limited crop diversity and dataset size make it unsuitable for generalized crop disease classification tasks. In contrast, the proposed model was trained on 16 crop species and 36 classes, making it significantly more versatile and applicable to real-world agricultural scenarios.

Another critical aspect of the proposed model is its dataset composition. Unlike existing studies that primarily rely on lab-setting images, the proposed model incorporates both real-world and lab-setting images, making it robust for practical deployment in diverse agricultural environments. This improvement addresses a critical gap in existing literature, where models trained solely on lab-controlled datasets may struggle with noisy and variable conditions encountered in real-world applications.

From a model design perspective, the proposed MobileNetV2-based architecture strikes an effective balance between computational efficiency and accuracy. Unlike models like SqueezeNext or EfficientNetB0, which are designed to minimize parameters and FLOPs at the cost of performance, the proposed model integrates a lightweight architecture with advanced training optimizations, enabling it to outperform the existing models. Additionally, while Tambe et al. [27] also utilized MobileNetV2, their model achieved only 91.98% accuracy, indicating the significant impact of the proposed model's tailored optimizations.

The key findings of this study highlight the practicality of employing lightweight CNN architectures for multi-class crop disease classification, especially in resource-limited agricultural settings. The model's ability to process diverse datasets highlights its potential for real-world applications, such as early disease detection and crop health monitoring. This can aid farmers and agricultural professionals in timely decision-making and ultimately contribute to improving crop yields and reducing losses.

4.7.Limitations of the study

While the proposed model achieved high classification accuracy and demonstrated computational efficiency, certain limitations must be acknowledged. First, although the dataset combined both lab-controlled and real-field images, it may still be subject to bias due to uneven representation of some crop-disease classes. The generalizability of the model to unseen datasets from different geographies, crop varieties, or image acquisition devices remains to be validated. Second, variations in image resolution, illumination, occlusion, and natural background clutter were only partially captured in the dataset, which may limit performance under highly diverse real-world conditions. Third, the study focused on tuning two key hyperparameters (dropout rate and learning rate), while other hyperparameters (e.g., batch size, weight decay) were not explored, leaving open the possibility of further optimization. Addressing these limitations will be essential for enhancing the applicability of the model in large-scale agricultural deployments.

5. CONCLUSION AND FUTURE WORK

This study presented an optimized MobileNetV2-based deep learning framework for the classification of 36 healthy and diseased crop classes across 16 plant species. Through systematic optimizer benchmarking and Optuna-based hyperparameter tuning, the proposed model achieved 98.82% accuracy with only 2.3M parameters and 599 MFLOPs, making it suitable for deployment on edge devices in resource-constrained agricultural settings. By integrating both lab-controlled and real-field images, the model demonstrated improved generalizability compared to prior lightweight approaches.

Looking ahead, three promising research directions emerge. First, expanding the dataset to include more crop-disease classes from diverse geographies would strengthen robustness and ensure wider applicability. Second, deploying the model in real-time scenarios, such as smartphone applications or IoT-enabled systems would provide practical validation of its field readiness. Third, future work may explore advanced efficiency techniques such as pruning, quantization, or lightweight ensemble learning to further enhance performance without increasing computational cost.

This study advances the technical frontier of lightweight deep learning for crop disease classification as well as holds broader societal relevance. By enabling early and accurate disease detection in real-world conditions, it contributes to achieving Sustainable Development Goal 2 (Zero Hunger) through improved food security and sustainable agricultural practices.

REFERENCES

1. FAO, "About | Plant Production and Protection | Food and Agriculture Organization of the United Nations," Plant-Production-and-Protection. Accessed: May 04, 2024. [Online]. Available: <https://www.fao.org/plant-production-protection/about/en>
2. S. Shukla, D. Upadhyay, A. Mishra, T. Jindal, and K. Shukla, "Challenges Faced by Farmers in Crops Production Due to Fungal Pathogens and Their Effect on Indian Economy," in *Fungal diversity, ecology and control management*, V. R. Rajpal, I. Singh, and S. S. Navi, Eds., Singapore: Springer Nature, 2022, pp. 495–505. doi: 10.1007/978-981-16-8877-5_24.
3. A. Khakimov, I. Salakhutdinov, A. Omolikhov, and S. Utaganov, "Traditional and current-prospective methods of agricultural plant diseases detection: A review," *IOP Conf. Ser.: Earth Environ. Sci.*, vol. 951, no. 1, p. 012002, Jan. 2022, doi: 10.1088/1755-1315/951/1/012002.
4. [4] images using convolutional neural networks: A systematic review," *Computers and Electronics in Agriculture*, vol. 185, p. 106125, Jun. 2021, doi: 10.1016/j.compag.2021.106125.
5. G. Mohyuddin, M. A. Khan, A. Haseeb, S. Mahpara, M. Waseem, and A. M. Saleh, "Evaluation of Machine Learning Approaches for Precision Farming in Smart Agriculture System: A Comprehensive Review," *IEEE Access*, vol. 12, pp. 60155–60184, 2024, doi: 10.1109/ACCESS.2024.3390581.
6. H. N. Ngugi, A. E. Ezugwu, A. A. Akinyelu, and L. Abualigah, "Revolutionizing crop disease detection with computational deep learning: a comprehensive review.," *Environ Monit Assess*, vol. 196, no. 3, p. 302, Feb. 2024, doi: 10.1007/s10661-024-12454-z.
7. T. Saranya, C. Deisy, S. Sridevi, and K. S. M. Anbananthen, "A comparative study of deep learning and Internet of Things for precision agriculture," *Engineering Applications of Artificial Intelligence*, vol. 122, p. 106034, Jun. 2023, doi: 10.1016/j.engappai.2023.106034.
8. B. Gülmez, "Advancements in rice disease detection through convolutional neural networks: A comprehensive review," *Heliyon*, vol. 10, no. 12, Jun. 2024, doi: 10.1016/j.heliyon.2024.e33328.
9. D. S. Joseph, P. M. Pawar, and R. Pramanik, "Intelligent plant disease diagnosis using convolutional neural network: a review," *Multimed Tools Appl*, vol. 82, no. 14, pp. 21415–21481, Jun. 2023, doi: 10.1007/s11042-022-14004-6.
10. Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," *IEEE Trans. Neural Netw. Learning Syst.*, vol. 33, no. 12, pp. 6999–7019, Dec. 2022, doi: 10.1109/TNNLS.2021.3084827.
11. G. W. Lindsay, "Convolutional Neural Networks as a Model of the Visual System: Past, Present, and Future," *Journal of Cognitive Neuroscience*, vol. 33, no. 10, pp. 2017–2031, Sep. 2021, doi: 10.1162/jocn_a_01544.
12. A. O. Anim-Ayeko, C. Schillaci, and A. Lipani, "Automatic blight disease detection in potato (*Solanum tuberosum* L.) and tomato (*Solanum lycopersicum*, L. 1753) plants

- using deep learning,” *Smart Agricultural Technology*, vol. 4, p. 100178, Aug. 2023, doi: 10.1016/j.atech.2023.100178.
13. P. Bedi and P. Gole, “Plant disease detection using hybrid model based on convolutional autoencoder and convolutional neural network,” *Artificial Intelligence in Agriculture*, vol. 5, pp. 90–101, Jan. 2021, doi: 10.1016/j.aiia.2021.05.002.
14. A. Carraro, G. Saurio, A. López-Maestresalas, S. Scardapane, and F. Marinello, “Convolutional Neural Networks for the Detection of Esca Disease Complex in Asymptomatic Grapevine Leaves,” in *Image Analysis and Processing - ICIAP 2023 Workshops*, G. L. Foresti, A. Fusiello, and E. Hancock, Eds., Cham: Springer Nature Switzerland, 2024, pp. 418–429. doi: 10.1007/978-3-031-51023-6_35.
15. K. P. Ferentinos, “Deep learning models for plant disease detection and diagnosis,” *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, Feb. 2018, doi: 10.1016/j.compag.2018.01.009.
16. M. Ji and Z. Wu, “Automatic detection and severity analysis of grape black measles disease based on deep learning and fuzzy logic,” *Computers and Electronics in Agriculture*, vol. 193, p. 106718, Feb. 2022, doi: 10.1016/j.compag.2022.106718.
17. N. Aishwarya, N. G. Praveena, S. Priyanka, and J. Pramod, “Smart farming for detection and identification of tomato plant diseases using light weight deep neural network,” *Multimed Tools Appl*, vol. 82, no. 12, pp. 18799–18810, May 2023, doi: 10.1007/s11042-022-14272-2.
18. C. Gao *et al.*, “A fast and lightweight detection model for wheat fusarium head blight spikes in natural environments,” *Computers and Electronics in Agriculture*, vol. 216, p. 108484, Jan. 2024, doi: 10.1016/j.compag.2023.108484.
19. M. J. Karim, M. O. F. Goni, M. Nahiduzzaman, M. Ahsan, J. Haider, and M. Kowalski, “Enhancing agriculture through real-time grape leaf disease classification via an edge device with a lightweight CNN architecture and Grad-CAM,” *Sci Rep*, vol. 14, no. 1, p. 16022, Jul. 2024, doi: 10.1038/s41598-024-66989-9.
20. G. Singh, G. Kasana, and K. Singh, “Improved Potato Crop Disease Classification Using Ensembled Convolutional Neural Network,” *Potato Res.*, Sep. 2024, doi: 10.1007/s11540-024-09787-0.
21. D. Pineda Medina, I. Miranda Cabrera, R. A. de la Cruz, L. Guerra Arzuaga, S. Cuello Portal, and M. Bianchini, “A Mobile App for Detecting Potato Crop Diseases,” *Journal of Imaging*, vol. 10, no. 2, Art. no. 2, Feb. 2024, doi: 10.3390/jimaging10020047.
22. R. Rashid, W. Aslam, R. Aziz, and G. Aldehim, “A Modified MobileNetv3 Coupled With Inverted Residual and Channel Attention Mechanisms for Detection of Tomato Leaf Diseases,” *IEEE Access*, vol. 13, pp. 52683–52696, 2025, doi: 10.1109/ACCESS.2025.3550205.
23. A. Abbas, S. Jain, M. Gour, and S. Vankudothu, “Tomato plant disease detection using transfer learning with C-GAN synthetic images,” *Computers and Electronics in Agriculture*, vol. 187, p. 106279, Aug. 2021, doi: 10.1016/j.compag.2021.106279.
24. Z. Tang, J. Yang, Z. Li, and F. Qi, “Grape disease image classification based on lightweight convolution neural networks and channelwise attention,” *Computers and*

- Electronics in Agriculture*, vol. 178, p. 105735, Nov. 2020, doi: 10.1016/j.compag.2020.105735.
25. D. Zhang, Y. Huang, C. Wu, and M. Ma, "Detecting tomato disease types and degrees using multi-branch and destruction learning," *Computers and Electronics in Agriculture*, vol. 213, p. 108244, Oct. 2023, doi: 10.1016/j.compag.2023.108244.
26. J. Lu, X. Liu, X. Ma, J. Tong, and J. Peng, "Improved MobileNetV2 crop disease identification model for intelligent agriculture," *PeerJ Comput. Sci.*, vol. 9, p. e1595, Sep. 2023, doi: 10.7717/peerj-cs.1595.
27. U. Y. Tambe, Dr. A. Shanthini, and P.-A. Hsiung, "Integrated Leaf Disease Recognition Across Diverse Crops through Transfer Learning," *Procedia Computer Science*, vol. 233, pp. 22–34, Jan. 2024, doi: 10.1016/j.procs.2024.03.192.
28. E. M. Dogo, O. J. Afolabi, N. I. Nwulu, B. Twala, and C. O. Aigbavboa, "A Comparative Analysis of Gradient Descent-Based Optimization Algorithms on Convolutional Neural Networks," in *2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS)*, Dec. 2018, pp. 92–99. doi: 10.1109/CTEMS.2018.8769211.
29. Md. A. B. Bhuiyan, H. M. Abdullah, S. E. Arman, S. Saminur Rahman, and K. Al Mahmud, "BananaSqueezeNet: A very fast, lightweight convolutional neural network for the diagnosis of three prominent banana leaf diseases," *Smart Agricultural Technology*, vol. 4, p. 100214, Aug. 2023, doi: 10.1016/j.atech.2023.100214.
30. 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, doi: 10.3389/fpls.2016.01419.
31. D. P. Hughes and M. Salathe, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," Apr. 11, 2016, *arXiv*: arXiv:1511.08060. doi: 10.48550/arXiv.1511.08060.
32. X. Fan and Z. Guan, "VGNet: A Lightweight Intelligent Learning Method for Corn Diseases Recognition," *Agriculture*, vol. 13, no. 8, Art. no. 8, Aug. 2023, doi: 10.3390/agriculture13081606.
33. N. Ullah *et al.*, "A Lightweight Deep Learning-Based Model for Tomato Leaf Disease Classification," *Computers, Materials and Continua*, vol. 77, no. 3, pp. 3969–3992, Dec. 2023, doi: 10.32604/cmc.2023.041819.
34. S. Duhan, P. Gulia, N. S. Gill, and E. Narwal, "RTR_Lite_MobileNetV2: A lightweight and efficient model for plant disease detection and classification," *Current Plant Biology*, vol. 42, p. 100459, Jun. 2025, doi: 10.1016/j.cpb.2025.100459.
35. A. H. Ali, A. Youssef, M. Abdelal, and M. A. Raja, "An ensemble of deep learning architectures for accurate plant disease classification," *Ecological Informatics*, vol. 81, p. 102618, Jul. 2024, doi: 10.1016/j.ecoinf.2024.102618.
36. A. S. Kini, K. V. Prema, and S. N. Pai, "Early stage black pepper leaf disease prediction based on transfer learning using ConvNets," *Sci Rep*, vol. 14, no. 1, p. 1404, Jan. 2024, doi: 10.1038/s41598-024-51884-0.
37. A. Ashurov *et al.*, "Enhancing plant disease detection through deep learning: a Depthwise CNN with squeeze and excitation integration and residual skip

- connections,” *Frontiers in Plant Science*, vol. 15, Jan. 2025, doi: 10.3389/fpls.2024.1505857.
38. D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” Jan. 30, 2017, *arXiv*: arXiv:1412.6980. doi: 10.48550/arXiv.1412.6980.
39. M. H. Saleem, S. Khanchi, J. Potgieter, and K. M. Arif, “Image-Based Plant Disease Identification by Deep Learning Meta-Architectures,” *Plants*, vol. 9, no. 11, Art. no. 11, Nov. 2020, doi: 10.3390/plants9111451.
40. S. Ashwinkumar, S. Rajagopal, V. Manimaran, and B. Jegajothi, “Automated plant leaf disease detection and classification using optimal MobileNet based convolutional neural networks,” in *Materials Today: Proceedings*, in CMAE’21, vol. 51. Jan. 2022, pp. 480–487. doi: 10.1016/j.matpr.2021.05.584.
41. D. Singh, N. Jain, P. Jain, P. Kayal, S. Kumawat, and N. Batra, “PlantDoc: A Dataset for Visual Plant Disease Detection,” in *Proceedings of the 7th ACM IKDD CoDS and 25th COMAD*, in CoDS COMAD 2020. New York, NY, USA: Association for Computing Machinery, Jan. 2020, pp. 249–253. doi: 10.1145/3371158.3371196.
42. T. D. Salka, M. B. Hanafi, S. M. S. A. A. Rahman, D. B. M. Zulperi, and Z. Omar, “Plant leaf disease detection and classification using convolution neural networks model: a review,” *Artif Intell Rev*, vol. 58, no. 10, p. 322, Jul. 2025, doi: 10.1007/s10462-025-11234-6.
43. Y. Ferdi, “Data Augmentation through Background Removal for Apple Leaf Disease Classification Using the MobileNetV2 Model,” Nov. 29, 2024, *arXiv*: arXiv:2412.01854. doi: 10.48550/arXiv.2412.01854.
44. S. Barman *et al.*, “Optimized Crop Disease Identification in Bangladesh: A Deep Learning and SVM Hybrid Model for Rice, Potato, and Corn,” *Journal of Imaging*, vol. 10, no. 8, Art. no. 8, Aug. 2024, doi: 10.3390/jimaging10080183.
45. S. Zhu and H. Gao, “MC-ShuffleNetV2: A lightweight model for maize disease recognition,” *Egyptian Informatics Journal*, vol. 27, p. 100503, Sep. 2024, doi: 10.1016/j.eij.2024.100503.
46. P. Lv *et al.*, “An improved lightweight ConvNeXt for rice classification,” *Alexandria Engineering Journal*, vol. 112, pp. 84–97, Jan. 2025, doi: 10.1016/j.aej.2024.10.098.
47. C. H. Praharsha, A. Poullose, and C. Badgujar, “Comprehensive Investigation of Machine Learning and Deep Learning Networks for Identifying Multispecies Tomato Insect Images,” *Sensors*, vol. 24, no. 23, Art. no. 23, Jan. 2024, doi: 10.3390/s24237858.
48. S. Quan, J. Wang, Z. Jia, M. Yang, and Q. Xu, “MS-Net: a novel lightweight and precise model for plant disease identification,” *Front. Plant Sci.*, vol. 14, Oct. 2023, doi: 10.3389/fpls.2023.1276728.
49. [49] Makerere AI Lab, “Bean disease dataset,” Bean disease dataset. Accessed: Nov. 10, 2024. [Online]. Available: <https://huggingface.co/datasets/AI-Lab-Makerere/beans>
50. J. Parraga-Alava, K. Cusme, A. Loor, and E. Santander, “RoCoLe: A robusta coffee leaf images dataset,” vol. 2, May 2019, doi: 10.17632/c5yvn32dzg.2.

51. Y. Hailu, "Banana Leaf Disease Images," vol. 1, Sep. 2021, doi: 10.17632/rjykr62kdh.1.
52. T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A Next-generation Hyperparameter Optimization Framework," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, in KDD '19. New York, NY, USA: Association for Computing Machinery, Jul. 2019, pp. 2623–2631. doi: 10.1145/3292500.3330701.
53. N. Reimers and I. Gurevych, "Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, M. Palmer, R. Hwa, and S. Riedel, Eds., Copenhagen, Denmark: Association for Computational Linguistics, Sep. 2017, pp. 338–348. doi: 10.18653/v1/D17-1035.
54. X. Bouthillier, "Accounting for variance and hyperparameter optimization in machine learning benchmarks," Jun. 2022, Accessed: Sep. 14, 2025. [Online]. Available: <http://hdl.handle.net/1866/27457>
55. J. Demsar and J. Demsar, "Statistical Comparisons of Classifiers over Multiple Data Sets," *Journal of Machine Learning Research*, vol. 7.
56. J. Cohen, "Statistical power analysis for the behavioral sciences, 2nd edn. 12 Lawrence Erlbaum Associates Inc," *Hillsdale, New Jersey*, vol. 13, 1988.
57. Y. Liu, G. Gao, and Z. Zhang, "Crop Disease Recognition Based on Modified Light-Weight CNN With Attention Mechanism," *IEEE Access*, vol. 10, pp. 112066–112075, 2022, doi: 10.1109/ACCESS.2022.3216285.
58. W. Macdonald, Y. A. Sari, and M. Pahlevani, "Grow-light smart monitoring system leveraging lightweight deep learning for plant disease classification," *Artificial Intelligence in Agriculture*, vol. 12, pp. 44–56, Jun. 2024, doi: 10.1016/j.aiia.2024.03.003.