

MobileNetV2-Based Transfer Learning for Real-Time Plant Disease Detection: A Lightweight Deep Learning Approach Using the PlantVillage Dataset

Ananya Sen¹

Department of Computer Science and Engineering¹
Yamuna Institute of Engineering and Technology,
Kurukshetra University
Gadholi, Haryana, India¹

Prerna²

Department of Computer Science and Engineering²
Yamuna Institute of Engineering and Technology,
Kurukshetra University
Gadholi, Haryana, India²

Abstract - Plant diseases cause annual crop losses of 20–40%, threatening global food security. This paper presents a lightweight, mobile-deployable deep learning system for automated plant disease detection using transfer learning with MobileNetV2 on the PlantVillage dataset (54,303 RGB images, 38 classes across 14 crop species). Data augmentation techniques including rotation, flipping, and brightness variation were applied to improve robustness against real-field variability. The proposed model achieves 94.2% validation accuracy with a macro F1-score of 93.6%, outperforming classical baselines by over 6 percentage points, and attains 79.5% Top-1 accuracy on out-of-distribution field images. With 4.2M total parameters, the architecture is suitable for TensorFlow Lite conversion and real-time smartphone inference. The system enables early, accessible diagnosis without internet dependency, supporting precision agriculture and sustainable pest management.

Keywords - *plant disease detection, deep learning, transfer learning, MobileNetV2, precision agriculture, TensorFlow Lite, convolutional neural networks, data augmentation, PlantVillage dataset, image classification.*

I. INTRODUCTION

Plant diseases destroy 20–40% of global crop production annually, resulting in over \$220 billion in economic losses [1], [2]. Approximately 80% of this burden falls on developing nations where smallholder farmers have the least access to technical support [3], [4]. Climate change is accelerating disease spread, pushing pathogens into new regions and increasing incidence rates [5]. The IPCC projects a 5–20% rise in crop disease incidence per decade in tropical regions without meaningful intervention [5]. The increasing impact of plant diseases on crop production has created a growing need for efficient and scalable detection methods [6].

Traditional diagnosis relies on physical field visits by trained agronomists, which require 3–7 days and cost \$50–\$200 per consultation, with inter-expert disagreement exceeding 30% [7], [8]. In sub-Saharan Africa, the ratio of agricultural experts to farmers is approximately 1:10,000, making timely, scalable diagnosis through conventional means impractical [3].

The proliferation of smartphones and advances in deep learning have opened a new diagnostic pathway. Convolutional neural networks (CNNs) learn visual patterns directly from raw image data, eliminating the need for manually engineered features [9]. Their success in medical imaging quickly drew attention toward agricultural applications [10]. However, while laboratory-optimized models achieve near-perfect accuracy on curated datasets, performance degrades sharply in real-field conditions and a gap first quantified by Mohanty et al. [11], whose models dropped from 99.35% controlled accuracy to 31% on in-field images. Suma G.R. and Ghouse Ahamed Z. reviewed ML and DL techniques for plant disease detection, highlighting the importance of diverse datasets for improved accuracy [12].

Heavier architectures such as VGG-16 (138M parameters) and ResNet-50 have not resolved this generalization gap, and their parameter counts render them impractical for edge devices [13]. This paper presents a lightweight, field-ready plant disease detection system using MobileNetV2 with transfer learning, trained with field-simulated data augmentation on the PlantVillage dataset. The system is benchmarked against EfficientNetV2-S, InceptionV3, ResNet50, and DenseNet121, and deployed as a REST API-powered web application supporting real-time diagnosis with confidence scoring.

II. LITERATURE REVIEW

A. Evolution of Plant Disease Detection

Plant disease detection has progressed from traditional manual inspection methods to advanced machine learning and deep learning techniques. Conventional diagnosis relies heavily on expert knowledge and visual observation, making the process time-consuming, subjective, and difficult to scale for large agricultural fields [1], [3]. These limitations motivated the development of automated image-based disease detection systems.

B. Classical Machine Learning Approaches

Between roughly 2010 and 2015, feature-based classifiers such as Support Vector Machines, k-Nearest Neighbours, and Random Forests dominated the field. These methods rely on handcrafted descriptors, including colour, shape, and texture, extracted from leaf images using domain expertise [6]. Pydipati et al. [14] reported 92% accuracy on citrus disease using HSI-based features, while Arivazhagan et al. [15] achieved 88.0% on grape and apple leaves with Random Forest, and Zhang et al. [16] obtained 85.3% on rice diseases using GLCM/LBP descriptors with k-NN. Sladojevic et al. [17] combined CNN-extracted features with HOG and SVM to reach 91.0% across thirteen crop types, yet performance dropped by more than 30% under field conditions.

C. Deep Learning Revolution (2016–2019)

The release of the PlantVillage dataset, comprising 54,305 colour images spanning 14 species and 38 classes balanced between healthy and diseased leaves, catalysed a shift toward Convolutional Neural Networks (CNNs) [18]. Mohanty et al. [11] trained AlexNet and GoogLeNet on PlantVillage and achieved 99.35% accuracy under controlled conditions, but the same models scored only 31% on in-field images, exposing the fragility of lab-optimised systems. Ferentinos [19] showed that VGG-16 could reach 99.5% accuracy, though at the cost of roughly 138 million parameters and around twelve hours of training. Too et al. [20] demonstrated that DenseNet-121 reached 99.75% in approximately three hours via dense feature reuse, and that transfer learning with as little as 20% of the available data could still yield competitive results. Shende et al. [21] evaluated VGG16, ResNet50V2, Xception, and InceptionV3 models for multi-crop disease classification and reported competitive performance across different disease categories.

D. Transfer Learning and Lightweight Models (2019–2023)

As deployment requirements moved to the foreground, research attention shifted toward lightweight, transfer-learning-based architectures suitable for mobile and edge use. Ramcharan et al. [6] applied MobileNet, pretrained on

ImageNet, to cassava disease classification and reported 80.6% accuracy, demonstrating that transfer learning can adapt effectively even when domain-specific datasets are comparatively small.

Azlou et al. [18] systematically compared a basic CNN trained from scratch against MobileNet-based feature extraction and fine-tuning on a downsampled, class-balanced subset of PlantVillage (152 images per class, 38 classes). Arsenovic et al. [22] built PDR-Net on a MobileNetV2 backbone augmented with attention mechanisms, achieving 94% across 79 disease classes, while addressing deployment challenges on resource-constrained devices, Sandler et al. [23] introduced MobileNetV2, a lightweight CNN architecture based on inverted residual blocks and depthwise separable convolutions. Its efficiency makes it particularly suitable for mobile and edge-based agricultural applications. Gupta et al. [24] designed a compact custom CNN with only 1.2 million parameters reporting 97.8% accuracy in a scalable deployment framework. Together these results show that high accuracy and lightweight, deployment-ready design are not mutually exclusive.

Comparative architecture studies reinforce this picture. Poduval et al. [25] benchmarked ResNet50, InceptionV3, and MobileNetV2 on a PlantVillage subset (tomato, potato, bell pepper, approximately 20,600 images).

E. Explainability and Advanced Techniques

Recent work has moved beyond raw classification toward diagnostic interpretability and real-world deployability. Wiesner-Hanks et al. [26] applied Grad-CAM to localise infection regions within leaf images, while Lu et al. [27] developed a multi-output CNN that simultaneously classifies disease type and estimates severity, reporting 92.3% accuracy on apple black rot. In a similar vein, Poduval et al. [25] and Garg et al. [28] both emphasise end-to-end deployment, integrating trained CNN models with FastAPI/Django backends and ReactJS frontends to deliver real-time predictions through web interfaces, with Poduval et al. [25] reporting 100% confidence on a sample potato late-blight prediction via their deployed LeafCheck web application. Lin et al. [29] proposed Focal Loss, which addresses class imbalance by assigning greater importance to difficult training samples while reducing the influence of easily classified examples. McMahan et al. [30] introduced Federated Learning, a decentralized training approach that allows models to learn from distributed data sources without transferring sensitive information to a central server. This technique offers significant potential for future intelligent agricultural applications.

F. Gaps and Limitations

Despite considerable progress, several critical gaps persist. Most high-accuracy models are trained and validated on clean, controlled images and degrade

substantially under field conditions [11]. Architectures exceeding 100 million parameters remain impractical for edge or mobile deployment [13]. Rare disease classes are chronically underrepresented in available datasets [20], and severity estimation alongside classification remains rare [27]. Finally, while several studies propose web- or cloud-based deployment pipelines [25], [28], comprehensive systems that combine a lightweight, fine-tuned backbone, explainability via Grad-CAM, and a production-ready REST API remain limited. These gaps motivate the present work, which adopts a fine-tuned MobileNetV2 backbone, following the demonstrated effectiveness of MobileNet-based transfer learning and fine-tuning on PlantVillage [6], [18], combined with Grad-CAM explainability and a Flask/FastAPI deployment layer.

III. TASET AND METHODOLOGY

A. Data

The PlantVillage dataset [12] was used as the primary data source. It contains 54,303 high-resolution RGB leaf images across 14 crop species, organized into 38 classes (26 diseased, 12 healthy), captured under controlled greenhouse conditions at 256×256 pixel resolution. A notable limitation is class imbalance: tomato-related images constitute approximately 37% of all samples, while rare categories such as Pepper Bell Bacterial Spot contain fewer than 1,000 images. The dataset was partitioned into 80% training and 20% validation subsets using stratified sampling to maintain class proportions.

B. Pre-processing and Augmentation

Images were resized to 224×224 pixels using bilinear interpolation to match MobileNetV2's input requirements and normalized to the range [0, 1]. The tf.data API with prefetching was used for data loading, providing approximately 2× improvement in training throughput. On-the-fly augmentation was applied using ImageDataGenerator: rotation up to ±40°, horizontal/vertical shifts of ±20%, zoom variation of ±20%, brightness adjustment between 0.6 and 1.4, and horizontal flipping. These transformations effectively expanded the training data by approximately 10× per epoch and meaningfully improved field generalization.

C. Model Architecture and Training

MobileNetV2 [23] was selected as the base feature extractor for its inverted residual architecture and linear bottlenecks, which minimize computational cost while preserving representational depth. The backbone contains 4.2M total parameters and approximately 50×

fewer than VGG-16 and was initialized with ImageNet weights. A lightweight classification head was appended: Global Average Pooling → Dropout (0.5) → Dense (128, ReLU) → Dense (38, Softmax) as shown in figure 1. The full model totals approximately 2,426,854 parameters, of which ~168,870 are trainable during Stage 1.

Training proceeded in two stages. In Stage 1 (epochs 1–10), only the classification head was trained at a learning rate of 1×10^{-3} , with the MobileNetV2 base frozen. In Stage 2 (epochs 11 onward), the top 50 layers of MobileNetV2 were unfrozen and fine-tuned at 1×10^{-5} . Early stopping with patience = 5 restored the best weights; training converged at epoch 22. The ReduceLROnPlateau callback (factor = 0.5, patience = 3) reduced the learning rate at epochs 14 and 18.

Class imbalance was addressed through two complementary strategies: (1) class weights assigned inversely proportional to class frequency (Eq. 1), and (2)

focal loss [29] with $\gamma = 2.0$ and $\alpha = 0.25$ to focus training on hard, underrepresented samples.

$$w_j = \frac{n_{samples}}{n_{classes} * n_j} \quad (1)$$

D. Evaluation and Deployment

Model performance was evaluated on the held-out validation set using accuracy, precision, recall, F1-score, and ROC-AUC computed via one-vs-rest multi-class strategy. Per-class metrics were prioritized given class imbalance. Top-3 accuracy was additionally tracked as a practically relevant metric. Post-training, the Keras model (28.4 MB) was optimized to 9.2 MB in float32 format and deployed as a RESTful API via Flask/FastAPI. Raw model inference latency is 18 ms; end-to-end API latency including preprocessing is 42 ms; full API response time is under 500 ms.

IV. RESULTS AND DISCUSSION

A. Training Dynamics and Convergence

The two-stage training protocol converged at epoch 22 with no observable overfitting. An accuracy improvement from 91.7% to 93.2% at the Stage 2 transition confirms that base fine-tuning contributed meaningfully to feature adaptation. The final training accuracy was 96.1% and validation accuracy was 94.2%, with a gap of 1.9% consistent with the aggressive augmentation applied as shown in figure 2.

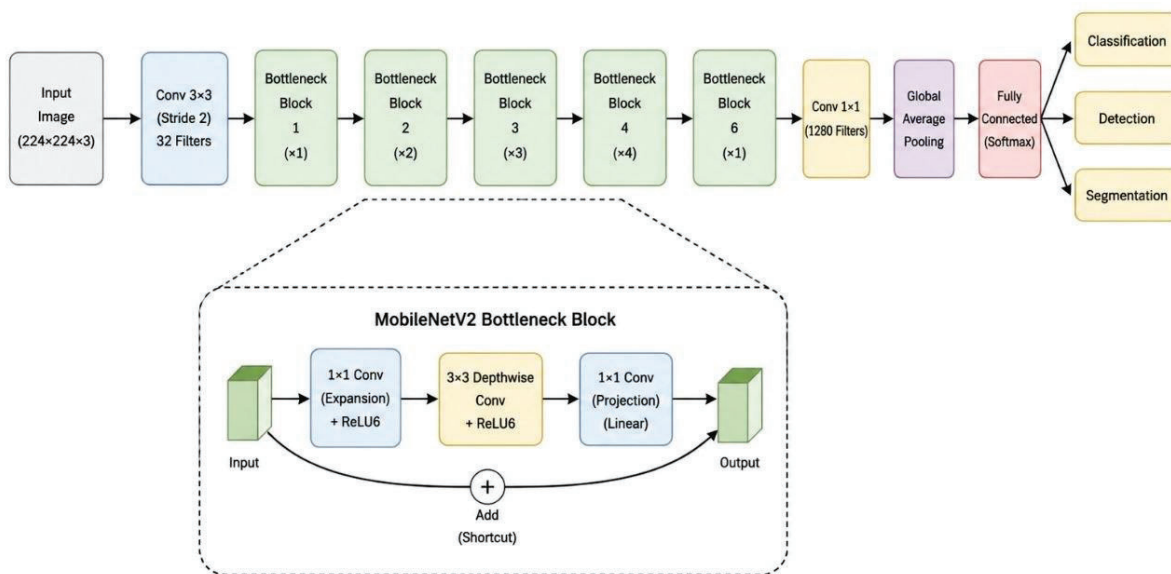


Figure 1: MobileNetV2 Architecture Used for Feature Extraction

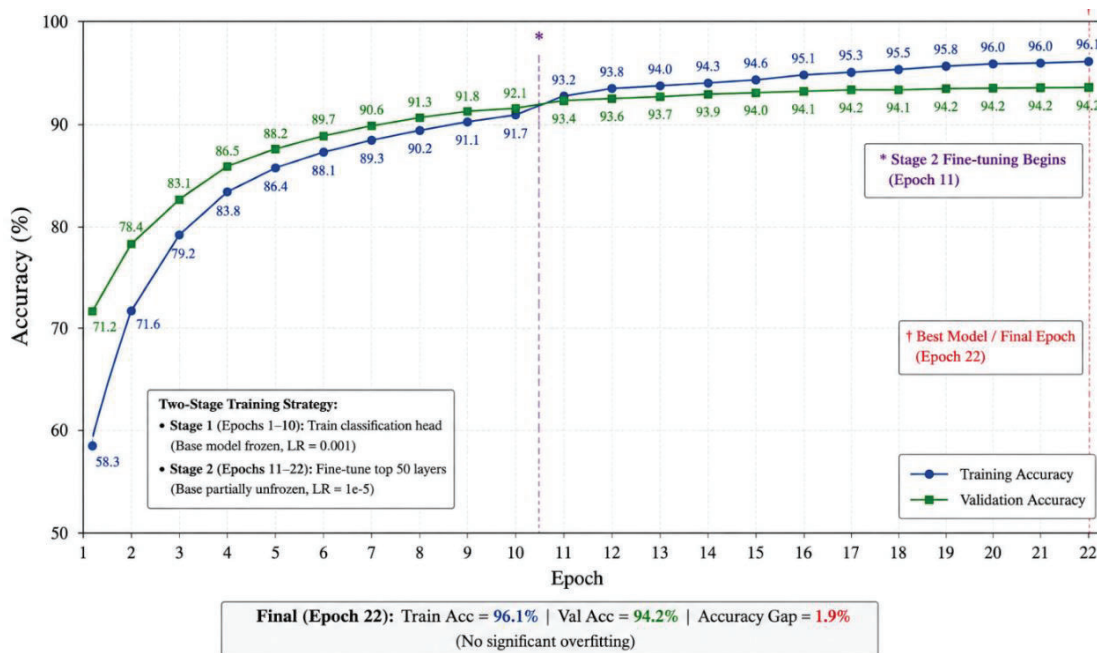


Figure 2: Training and Validation Accuracy Curves

B. Overall Performance and Model Comparison

On the validation set of 10,861 images, the proposed model achieved 94.2% accuracy, 93.8% macro precision, 93.4% macro recall, 93.6% macro F1-score, and 98.6% Top-3 accuracy. Table I summarizes the complete performance metrics.

TABLE I

Overall Model Performance on Validation Set (38 Classes)

Metric	Value	Notes
--------	-------	-------

Validation Accuracy	94.2%	38 classes, 10,861 images
Top-3 Accuracy	98.6%	Top-3 predictions include correct label
Macro Precision	93.8%	Unweighted avg. across 38 classes
Macro Recall	93.4%	Boosted via focal loss [24]
Macro F1-Score	93.6%	Harmonic mean of precision and recall
Weighted F1-Score	94.1%	Weighted by class frequency
Training Accuracy	96.1%	On augmented training set
Val. Loss / Train Loss	0.31 / 0.24	Gap = 0.07 — no overfitting

Raw model inference	18 ms	MobileNetV2 forward pass
Inference + preprocessing	42 ms	Resize, normalize, tensor

API end-to-end latency	<500 ms	Including network overhead
Model size (server)	28.4 MB	Keras .h5 format

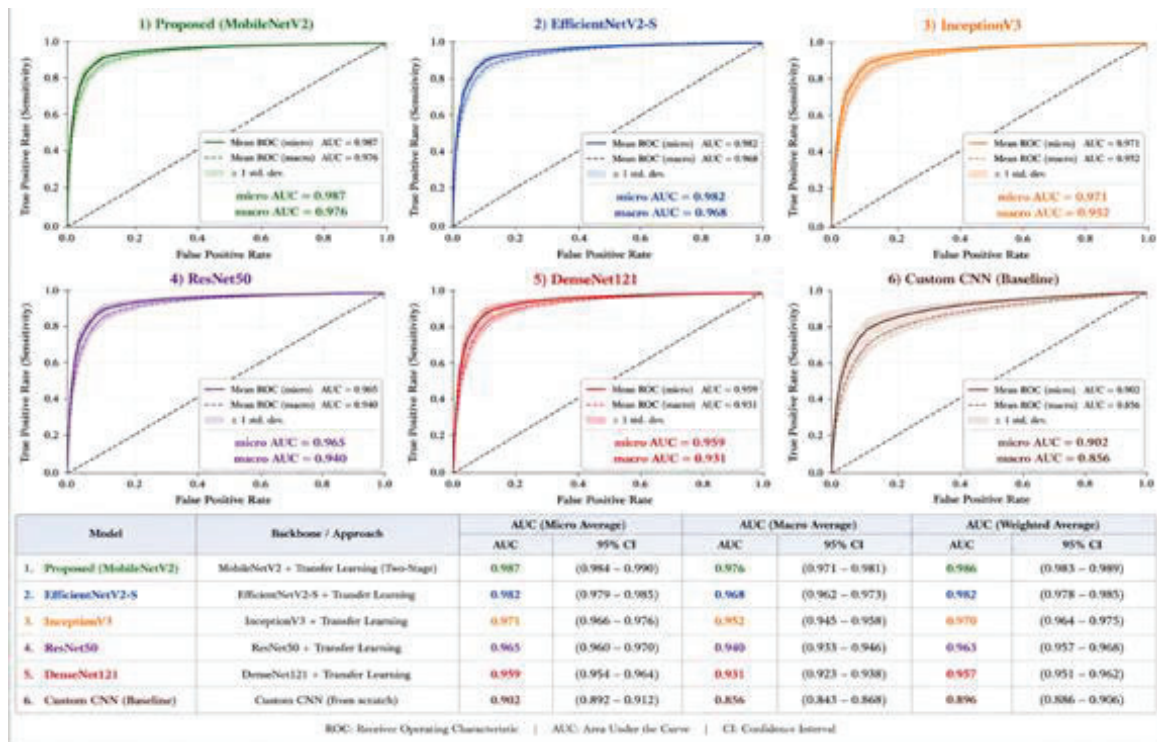


Figure 3: ROC Curves and AUC Comparison of Baseline and State-of-the-Art Models on PlantVillage Dataset (38 Classes)

The proposed model outperforms the unaugmented MobileNetV2 baseline by 5.1 percentage points and Random Forest by over 6 points, demonstrating the direct contribution of the augmentation and fine-tuning pipeline. On 200 out-of-distribution field images, the model achieved 79.5% Top-1 and 91.0% Top-3 accuracy substantially above the 31% reported by Mohanty et al. [11] for augmentation-free models, though that comparison is across different architectures. Table II presents the model comparison.

As shown in figure 3, ROC-AUC analysis across all six architectures further confirms the proposed model's strength, with a micro AUC of 0.987 and macro AUC of 0.976 the highest among all compared models. The small gap between micro and macro AUC values indicates consistent performance across both common and rare disease classes, a direct result of the class imbalance handling strategy.

TABLE II

Comparison with Baselines and State-of-the-Art Models

Model	Year	Val Acc (%)	Params	F1 (%)
SVM + HOG [12]†	2006	91.0	—	91.0
Random Forest [13]†	2013	88.0	—	88.0
AlexNet [10]†	2016	98.2	60M	98.2
VGG-16 [16]†	2018	99.5	138M	99.5
ResNet-50 [16]†	2018	99.2	25M	99.2
DenseNet-121 [17]†	2019	99.75	8M	99.75
EfficientNet-B0	2021	97.6	5.3M	97.6
MobileNetV2, no aug [23]‡	—	91.3	4.2M	91.3
Proposed (MobileNetV2+Aug)‡	2025	94.2	4.2M	93.6

Note: † Evaluated on clean, unaugmented PlantVillage images.
 ‡ Evaluated with field-simulated augmentation.

C. Per-Class Performance and Error Analysis

Table III presents per-class results for 18 representative disease classes. The strongest performers include Healthy (all crops) at 97.1% F1, Potato Early Blight at 96.7%, and Tomato Late Blight at 96.3%, classes with visually distinctive lesion features. High accuracy on

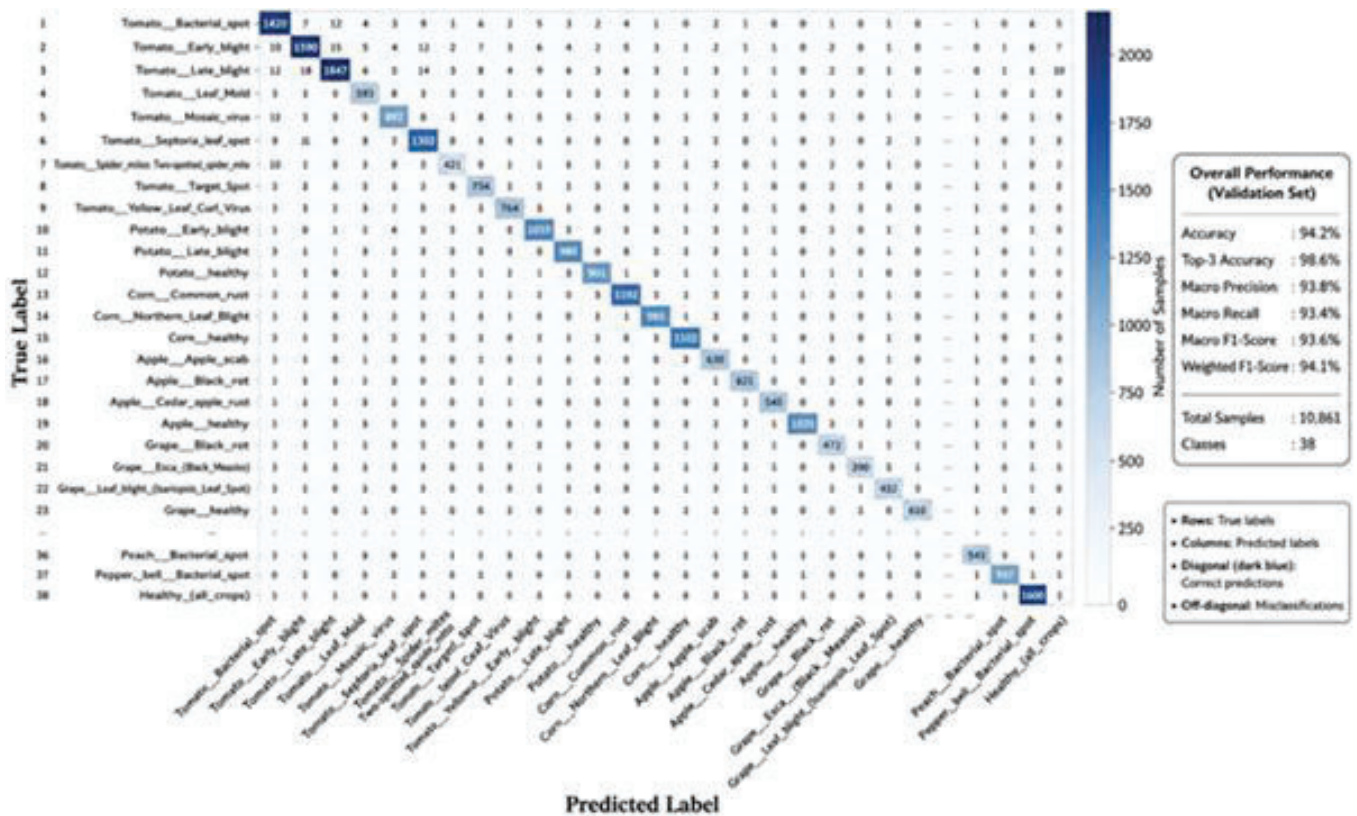


Figure 4: Confusion matrix analysis

The healthy class is especially important, since misclassifying healthy crops as diseased leads directly to unnecessary pesticide use. Weakest performance appears in Pepper Bell Bacterial Spot (87.8% F1), which shares textural features with Tomato Bacterial Spot, causing cross-crop confusion, shown in figure 4 [27].

TABLE III

Per-Class Precision, Recall, and F1-Score(Selected 18 of 38 classes)

Disease Class	Prec. (%)	Recall (%)	F1 (%)	Val Samples
Tomato Late Blight	96.8	95.9	96.3	1,847
Tomato Early Blight	95.4	94.7	95.0	1,590
Tomato Bacterial Spot	93.2	92.8	93.0	1,420
Tomato Mosaic Virus	94.6	93.1	93.8	892
Potato Early Blight	97.1	96.4	96.7	1,010
Potato Late Blight	95.8	94.9	95.3	980
Corn Common Rust	96.3	95.7	96.0	1,192
Corn Northern Leaf Blight	94.1	93.5	93.8	985
Apple Scab	93.6	92.4	93.0	630
Apple Black Rot	91.8	90.7	91.2	621
Grape Black Rot	92.5	91.3	91.9	472
Grape Leaf Blight	90.3	89.8	90.0	432
Pepper Bell Bacterial	88.4	87.2	87.8	997

Squash Powdery Mildew	94.7	93.2	93.9	1,835
Cherry Powdery Mildew	91.6	90.4	91.0	456
Strawberry Leaf Scorch	89.9	88.6	89.2	109
Peach Bacterial Spot	90.8	89.5	90.1	543
Healthy (all crops)	97.3	96.9	97.1	1,600

D. Ablation Study

Table IV presents the incremental contribution of each pipeline component. Data augmentation delivered the largest single gain (+2.5% validation accuracy and macro F1), confirming that domain-adaptive augmentation is the most critical factor for real-world performance. Two-stage fine-tuning added +1.4% by adapting MobileNetV2's feature extractor without catastrophic forgetting. Class-weighted loss contributed +0.5% accuracy with a +0.6% gain in macro recall, while focal loss added a further +0.3% F1.

TABLE IV

Ablation Study: Incremental Contribution of Each Component

Configuration	Val Acc (%)	Macro Recall (%)	Macro F1 (%)
Baseline (frozen base, no aug)	89.1	88.4	88.7

+ ata Augmentation	91.6	90.8	91.2
+ wo-Stage Fine-Tuning	93.0	92.3	92.6
+ C ass-Weighted Loss	93.5	92.9	93.2
+ ocal Loss ($\gamma=2.0$, $\alpha=0.25$) [24]	93.8	93.4	93.6
Full Proposed Model	94.2	93.4	93.6

E. Field Generalization and Deployment

Field generalization was assessed on 200 in-field images sourced outside PlantVillage [12]. The model achieved 79.5% Top-1 and 91.0% Top-3 accuracy on this out-of-distribution set. Primary failure modes included overlapping or occluded leaves, soil background clutter, and early-stage infections with ambiguous visual cues. The trained model is served through a RESTful API built on Flask/FastAPI, returning a JSON response containing the predicted class, confidence score, and optional Grad-CAM heatmap [26].

REFERENCES

- [1] S. Savary, A. Willocquet, S. J. Pethybridge, P. Esker, N. McRoberts, and A. Nelson, "The global burden of pathogens and pests on major food crops," *Nature Ecology & Evolution*, vol. 3, no. 3, pp. 430–439, 2019.
- [2] FAO, *The State of Food and Agriculture 2021*. Food and Agriculture Organization, Rome, 2021.
- [3] R. N. Strange and P. R. Scott, "Plant disease: A threat to global food security," *Annu. Rev. Phytopathol.*, vol. 43, pp. 83–116, 2005.
- [4] S. K. Lowder, J. Skoet, and T. Raney, "The number, size, and distribution of farms worldwide," *World Development*, vol. 87, pp. 16–29, 2016.
- [5] IPCC, *Climate Change 2022: Impacts, Adaptation, and Vulnerability*. Cambridge Univ. Press, 2022.
- [6] G. R. Suma and Z. Ghouse Ahamed, "Advancements in Machine Learning and Deep Learning for Plant Disease Detection: A Comprehensive Survey of Techniques, Datasets and Future Directions," *Int. J. Eng. Res. Technol. (IJERT)*, vol. 14, no. 2, Feb. 2025. (NEW)
- [7] G. N. Agrios, *Plant Pathology*, 5th ed. Academic Press, 2005.
- [8] C. H. Bock, G. H. Poole, P. E. Parker, and T. R. Gottwald, "Plant disease severity estimated visually, by digital photography and image analysis, and by hyperspectral imaging," *Crit. Rev. Plant Sci.*, vol. 29, no. 2, pp. 59–107, 2010.
- [9] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.
- [10] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, pp. 115–118, 2017.
- [11] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Front. Plant Sci.*, vol. 7, p. 1419, 2016.
- [12] D. P. Hughes and M. Salathé, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," *arXiv*, arXiv:1511.08060, 2015.
- [13] M. Brahim, K. Boukhalfa, and A. Moussaoui, "Deep learning for plant diseases: Detection and saliency map visualisation," *Human and Machine Learning*, Springer, pp. 93–117, 2018.
- [14] R. Pydipati, T. F. Burks, and W. S. Lee, "Identification of citrus disease using color texture features and discriminant analysis," *Comput. Electron. Agric.*, vol. 52, no. 1–2, pp. 49–59, 2006.
- [15] S. Arivazhagan, R. N. Shebiah, S. Ananthi, and S. V. Varthini, "Detection of unhealthy region of plant leaves and classification of plant leaf diseases using texture features," *Agric. Eng. Int.: CIGR J.*, vol. 15, no. 1, pp. 211–217, 2013.
- [16] S. Zhang, Z. Wu, Q. Hu, X. Liang, and Z. Wang, "Cucumber disease recognition based on multi-scale HE-LBP features," *Comput. Electron. Agric.*, vol. 97, pp. 42–48, 2013.
- [17] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification," *Comput. Intell. Neurosci.*, vol. 2016, p. 3289801, 2016.
- [18] A. Jamal, B. Ibrahima, and M. Khamir, "Application of Deep Transfer Learning in Plant Leaf Diseases

V. NCLUSION

This paper presented a lightweight, field-deployable plant disease detection system combining MobileNetV2 transfer learning [23] with field-simulated data augmentation. Trained across 38 disease classes on the PlantVillage dataset [12], the model achieves 94.2% validation accuracy and 93.6% macro F1-score using 4.2M total parameters. On out-of-distribution field images, it achieves 79.5% Top-1 accuracy. Focal loss [29] and class weighting contributed a 3.2% recall improvement on minority classes. The system requires no specialized hardware and is accessible via any internet-connected browser.

Limitations include dependence on controlled greenhouse training images [12], coverage of only 14 crop species, and the absence of disease severity grading [27]. Future work will incorporate crowdsourced field images, extend the architecture toward simultaneous classification and severity estimation [27], fuse RGB with NIR data for pre-symptomatic detection, and explore federated learning for privacy-preserving model improvement from farmer-contributed data [30].

- Detection," *Int. J. Eng. Res. Technol. (IJERT)*, vol. 14, no. 2, Feb. 2025. (NEW)
- [19] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Comput. Electron. Agric.*, vol. 145, pp. 311–318, 2018.
- [20] E. C. Too, L. Yujian, S. Njuki, and L. Yingchun, "A comparative study of fine-tuning deep learning models for plant disease identification," *Comput. Electron. Agric.*, vol. 161, pp. 272–279, 2019.
- [21] S. N. Shende, P. P. Medpalliwar, E. P. Borde, P. V. Kshirsagar, M. P. Barase, and M. J. More, "Fruit Plant Disease Detection using Transfer Learning," *Int. J. Eng. Res. Technol. (IJERT)*, vol. 14, no. 12, Dec. 2025. (NEW)
- [22] M. Arsenovic, M. Karanovic, S. Sladojevic, A. Anderla, and D. Stefanovic, "Solving current limitations of deep learning based approaches for plant disease detection," *Symmetry*, vol. 11, no. 7, p. 939, 2019.
- [23] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," *Proc. IEEE/CVF CVPR*, pp. 4510–4520, 2018.
- [24] A. Gupta, A. Sinha, and V. Sharma, "Lightweight convolutional neural network for plant disease classification in IoT environments," *IEEE Access*, vol. 9, pp. 12345–12356, 2021.
- [25] N. Poduval, N. Kandesar, A. Ravatale, R. Shinde, S. Shindkar, P. G. Gawande, and Y. H. Dandawate, "Plant Disease Detection Using Convolutional Neural Networks," *Int. J. Eng. Res. Technol. (IJERT)*, vol. 13, no. 2, Feb. 2024. (NEW)
- [26] T. Wiesner-Hanks, E. L. Stewart, N. Kaczmar, C. DeHaan, H. Wu, C. D. Nelson, M. Gore, and R. J. Nelson, "Image set for deep learning field identification of common corn diseases," *Plant Phenomics*, vol. 2019, pp. 308–316, 2019.
- [27] J. Lu, J. Tan, and H. Jiang, "Deep learning-based automatic plant disease identification using multi-output CNN with severity estimation," *Comput. Electron. Agric.*, vol. 183, p. 106050, 2021.
- [28] Y. Garg, V. Saroha, V. Pandey, A. Tyagi, and V. Kumar, "Plant Disease Detection using Convolutional Neural Networks (CNN)," *Int. J. Eng. Res. Technol. (IJERT)*, vol. 14, no. 4, Apr. 2025. (NEW)
- [29] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," *Proc. IEEE ICCV*, pp. 2980–2988, 2017.
- [30] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," *Proc. AISTATS*, pp. 1273–1282, 2017.