



RESEARCH ARTICLE

An ensemble of MobileNetV2 and DenseNet121 for tomato leaf disease detection

Rekha B¹, Vijayalaxmi Mekali², Kavya G³, Shreedevi S Ronihal^{4*} & Nandini G⁵

¹Department of Computer Science (Data Science), S J B Institute of Technology, Bengaluru 560 060, Karnataka, India

²Department of Computer Science, S J B Institute of Technology, Bengaluru 560 060, Karnataka, India

³Department of Computer Science and Engineering, K S Institute of Technology, Bengaluru 560 109, Karnataka, India

⁴Department of Information Science and Engineering, B N M Institute of Technology, Bengaluru 560 070, Karnataka, India

⁵Department of Computer Science and Engineering, B N M Institute of Technology, Bengaluru 560 070, Karnataka, India

*Correspondence email - shreedevironiha@gmail.com

Received: 06 December 2025; Accepted: 11 February 2026; Available online: Version 1.0: 06 May 2026

Cite this article: Rekha B, Vijayalaxmi M, Kavya G, Shreedevi SR, Nandini G. An ensemble of MobileNetV2 and DenseNet121 for tomato leaf disease detection. Plant Science Today. 2026; 13(sp2): 1-11. <https://doi.org/10.14719/pst.13105>

Abstract

Agriculture plays a crucial role in the global economy and crop production is heavily affected by phytopathological factors, especially plant leaf diseases. Early and precise detection of these diseases is essential because they directly affect crop yields, economic stability and the market value of agricultural produce. This research presents an advanced deep learning framework for accurate detection and classification of tomato leaf diseases. It employs a feature concatenation ensemble approach integrating MobileNetV2 and DenseNet121. MobileNetV2 provides efficient, lightweight feature extraction, while DenseNet121 offers rich hierarchical features through enhanced feature propagation and reuse. The features extracted by both models are concatenated to create a comprehensive representation, improving overall classification performance. The proposed ensemble achieves state-of-the-art accuracy of 99.45 %. This significantly improves disease diagnosis and enables effective crop management, ultimately boosting agricultural productivity.

Keywords: convolutional neural networks; deep learning framework; DenseNet121; MobileNetV2; tomato leaf disease detection

Introduction

Agriculture remains one of the most critical sectors underpinning the global economy and crop productivity is significantly affected by various phytopathological factors, with plant leaf diseases being a major contributor to yield losses. Timely and accurate disease detection is therefore essential to reduce crop damage, improve market value and ensure food security; however, conventional diagnosis relies largely on manual inspection by experts, which is time-consuming, subjective and difficult to scale. Recent advances in deep learning have enabled automated image-based disease detection as a promising alternative, with convolutional neural networks (CNNs) demonstrating strong performance in plant disease classification tasks. Nevertheless, single-model approaches often suffer from limited feature representation and reduced generalisation across diverse disease patterns and environmental conditions. In parallel, recent studies have explored attention mechanisms and transformer-based or hybrid CNN transformer architectures to capture global contextual information and long-range dependencies, further improving robustness in plant pathology applications. To address the limitations of single-model pipelines while retaining computational efficiency, this study proposes a feature concatenation-based ensemble approach integrating MobileNetV2 and DenseNet121,

effectively combining lightweight feature extraction with rich hierarchical feature representation to enhance tomato leaf disease detection accuracy.

Related work

Recent studies have applied CNN-based models for plant disease detection, leveraging architectures such as VGGNet, ResNet and Inception (1). MobileNetV2 is widely appreciated for its lightweight architecture and efficiency in feature extraction, making it suitable for deployment in resource-constrained environments (2). DenseNet121, with its dense connectivity, excels at capturing intricate hierarchical features by reusing previous layer outputs, improving gradient flow during training (3). Some prior works focus on model ensembling techniques, averaging predictions or stacking classifiers. However, few studies have explored direct feature concatenation to enhance representation capability. Our research aims to fill this gap by combining complementary features from MobileNetV2 and DenseNet121 into a unified framework for high-accuracy tomato leaf disease classification.

Tomato diseases threaten yield and quality, making early detection essential. A deep learning ensemble of MobileNetV2 and ResNet50 provides a scalable, precise solution for automated tomato disease diagnosis in smart agriculture (4). Tomato gray leaf spot, worsened by greenhouse expansion and new varieties,

causes major losses; early detection via deep learning and mobile apps can help farmers manage it in real time (5). LeafDoc-Net, a lightweight transfer-learning model combining DenseNet121 and MobileNetV2 with attention and enhanced layers, accurately detects cassava and wheat leaf diseases, outperforming existing models, even with limited data and its performance is validated using Grad-CAM++ (6). Early detection of tomato leaf diseases is vital for crop yield and economic sustainability. This study uses deep learning models like VGG19 and CNN to classify diseases, aiding effective crop management (7). The study proposes an ensemble of MobileNetV2 and Xception to improve feature extraction and disease classification, outperforming several CNN models while remaining lightweight for mobile deployment (8). Tomato leaf diseases threaten crop yield and economic viability. This study develops an optimised compact convolutional transformer (CCT) for accurate, scalable disease detection, outperforming VGG16, ResNet50 and vision transformers, providing a robust solution for real-time agricultural applications (9).

Cacao fruit diseases threaten cocoa yield in Indonesia. The study shows DenseNet-121 outperforms MobileNetV2 for automated disease classification, offering higher accuracy and robustness, making it suitable for web-based tools to assist farmers in timely detection (10). This study develops a MobileNet-based deep learning model for accurate plant disease detection in tomato, bean and chili plants, supporting precision farming and timely diagnosis (11). Table 1 compares the findings from various studies.

Methodology

The proposed methodology presents a hybrid deep learning framework that combines the efficiency of MobileNetV2 and the representational strength of DenseNet121 through an ensemble integration strategy for precise tomato leaf disease classification. Initially, tomato leaf images are pre-processed through resizing, normalisation and data augmentation to ensure balanced and consistent input quality. The processed images are then fed into two parallel feature extraction networks: MobileNetV2, which employs depth wise separable convolutions and inverted residual blocks to capture lightweight yet discriminative spatial features and DenseNet121, which utilises dense connectivity to promote feature reuse and gradient stability for complex texture representation. The feature maps obtained from both networks are concatenated and passed to a custom classification head comprising fully connected layers and a SoftMax activation to produce probabilistic outputs across 10 disease categories. The final ensemble model integrates the predictions of both architectures using a weighted averaging mechanism, effectively balancing accuracy and generalisation while minimising overfitting. Optimised with the Adam optimiser and categorical cross-entropy loss, this hybrid approach achieves high computational efficiency and superior diagnostic accuracy, making it well suited for real-time smart agriculture applications.

The architectural parameters of MobileNetV2 and DenseNet121, such as input resolution, depth, learning rate, batch size and optimiser configuration, were selected based on empirical validation and prior studies reported in recent literature to ensure a balance between computational efficiency and classification performance. The feature concatenation strategy is adopted to fuse complementary representations learned by both networks, enabling the model to capture both lightweight spatial features

and rich hierarchical patterns. This fusion enhances discriminative capability by providing a more comprehensive feature space to the classifier, thereby improving robustness and overall detection accuracy compared to individual backbone models.

A. MobileNetV

To classify tomato leaf images into 10 disease categories-tomato bacterial spot, tomato early blight, tomato late blight, tomato leaf mold, tomato septoria leaf spot, tomato spider mites, tomato target spot, tomato yellow leaf curl virus, tomato mosaic virus and tomato healthy the suggested model uses a MobileNetV2 backbone which is shown in Fig. 1. Using depth wise separable convolutions and bottleneck residual blocks for effective feature extraction, this architecture is tuned for both high performance and low computational cost.

Input layer: The input image, denoted as

$$X \in \mathbb{R}^{224 \times 224 \times 3} \quad [1]$$

It represents a colour tomato leaf image with RGB channels. Each pixel value is normalised to [0, 1] or standardised by the ImageNet mean and standard deviation for transfer learning.

Convolutional feature extraction (Conv2D Layer): The Conv2D layer extracts spatial and colour features using a learnable kernel W .

The operation is defined as:

$$Y_{i,j,o} = \sum_{u=-k}^k \sum_{v=-k}^k \sum_{c=1}^3 X_{i+u,j+v,c} \cdot W_{u,v,c,o} + b_o \quad [2]$$

where k is the kernel radius, b_o is the bias term and o indexes the output channels. This produces a feature map $Y \in \mathbb{R}^{H' \times W' \times o}$ capturing local texture variations like disease spots and discolorations.

Inverted residual block (depth wise + pointwise convolutions): MobileNetV2 replaces standard convolutions with depth wise separable convolutions, reducing computation by factorising convolution into two stages:

Depth wise convolution: Each channel is convolved independently:

$$Z_{i,j,c}^{(d)} = \sum_{u=-k}^k \sum_{v=-k}^k Y_{i+u,j+v,c} \cdot W_{u,v,c}^{(d)} \quad [3]$$

Pointwise (1×1) convolution: Combines outputs across channels:

$$X_{i,j,o}^{(p)} = \sum_{c=1}^C Z_{i,j,c}^{(d)} \cdot W_{c,o}^{(p)} \quad [4]$$

If input and output dimensions match, a residual connection is added

$$Z_{out} = Z^{(p)} + Y \quad [5]$$

This block is the core computational unit of MobileNetV2, enabling efficient nonlinear transformations while maintaining feature integrity.

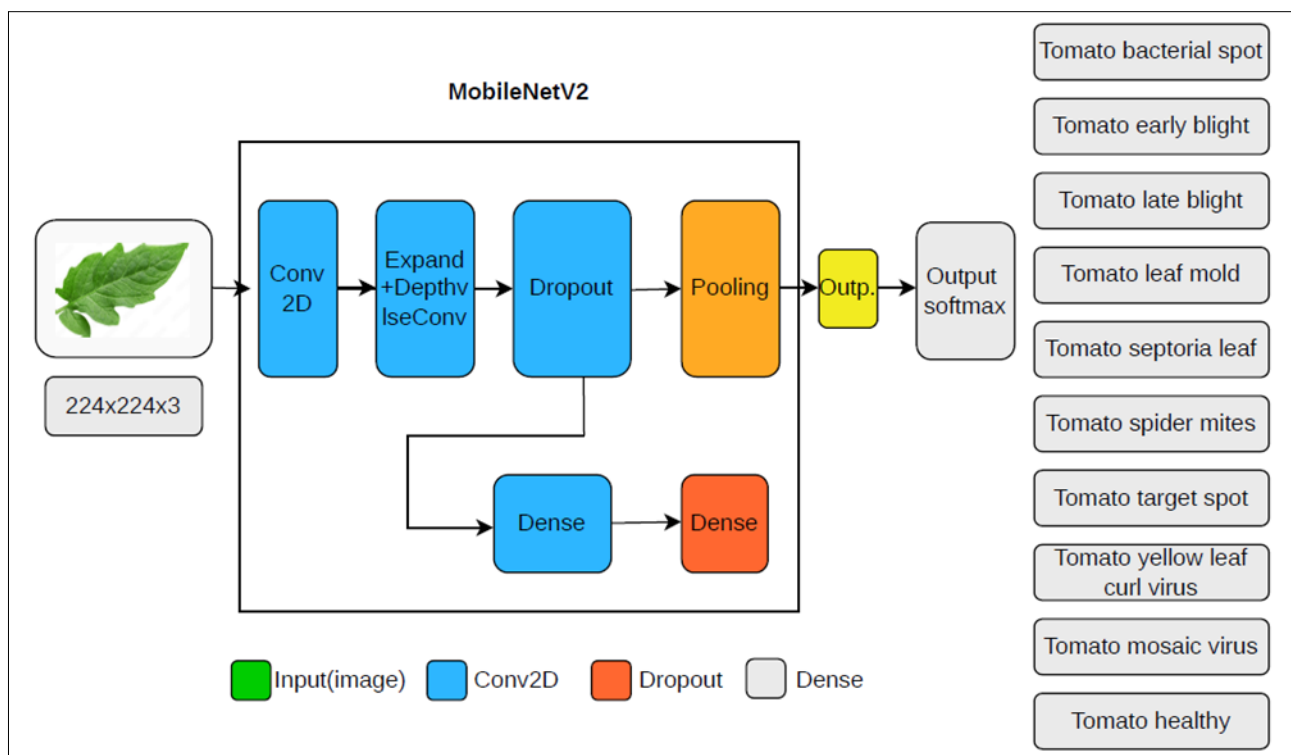
Dropout regularisation: Dropout prevents overfitting by randomly dropping neuron activations with probability p . For an activation a_i , dropout applies:

$$a_i^{\sim} = \frac{m_i \cdot a_i}{1-p}, \quad m_i \sim \text{Bernoulli}(1-p) \quad [6]$$

This $E[a_i^{\sim}] = a_i$, ensures maintaining expected activation levels while promoting generalisation.

Table 1. Comparison between the papers

Sl. No	Paper title	Model	Contribution	Limitation
1	InsightNet: A deep learning framework for enhanced plant disease detection (11).	InsightNet (DL)	Improved detection of plant diseases, especially tomato leaf diseases.	Limited generalisation to other crops; may require large training data.
2	A robust ensemble model for plant disease detection using pre-trained deep learning models (12).	Ensemble of pre-trained CNNs	Robust disease detection through model integration.	Higher computational cost; slower inference time.
3	Optimised custom CNN for real-time tomato leaf disease detection (13)	Custom CNN	Optimised CNN for real-time tomato leaf disease detection	May have lower accuracy on complex datasets; model tuning needed
4	A lightweight and efficient model for plant disease detection (14)	Lightweight CNN	Efficient and fast detection of plant diseases	May compromise accuracy for lightweight design
5	A lightweight dual-attention network for tomato leaf disease classification (15)	Dual-attention CNN	Enhanced classification using attention mechanism	Increased model complexity; longer training time
6	Transform and deep learning algorithms for the early detection of tomato diseases (16)	Transformer + CNN	Early detection of tomato diseases	Requires extensive computational resources; complex architecture
7	Plant leaf disease detection and classification using deep learning models (17)	Various CNN models	Overview of DL models for leaf disease detection and classification	Lacks experimental comparison; mainly theoretical
8	High-resolution leaf disease classification using deep learning (18)	High-resolution CNN	Accurate classification using high-resolution images	High memory and computation requirements
9	BotanicX-AI: Identification of tomato leaf diseases using deep learning (19)	BotanicX-AI (DL model)	Deep learning-based identification of tomato leaf diseases	Limited testing on diverse environmental conditions
10	Plant disease detection and classification techniques (20)	Various AI & DL models	Survey of plant disease detection techniques	Mostly descriptive; lacks performance benchmarking
11	Deep learning-based automatic diagnosis of rice leaf diseases (21)	CNN/DL models	Methodologies applicable to tomato leaf diseases	May not generalise well to other crop types
12	Tomato leaf disease detection and management using deep learning (22)	CNN/DL-based detection	Combines detection and management strategies	Integration with management tools may be challenging

**Fig. 1.** Architecture of MobileNetV2.

Global average pooling (GAP): GAP replaces traditional flattening by averaging each feature map spatially:

$$g_c = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W Z_{(i,j,c)} \quad [7]$$

The resulting vector $g \in R^c$ represents global descriptors for each feature channel, improving model interpretability and reducing overfitting.

Fully connected (dense) layers: Dense layers combine extracted features for classification:

$$h = f(W^{(1)}g + b^{(1)}) \quad [8]$$

where $f(\cdot)$ is a ReLU activation function. Subsequent dense layers further refine representations:

$$h' = f(W^{(2)}h + b^{(2)}) \quad [9]$$

These layers learn discriminative patterns linking leaf textures to specific disease categories.

Output softmax layer: The output layer maps logits z_k (one per class) to probabilities using the softmax function:

$$P(y = k|z) = \frac{e^{z_k}}{\sum_{j=1}^{10} e^{z_j}} \quad [10]$$

The model predicts the class with the highest probability:

$$\hat{y} = \operatorname{argmax}_k P(y = k) \quad [11]$$

This yields classification among ten possible outcomes corresponding to the tomato leaf diseases.

The architectural flow of MobileNetV2 is detailed in Table 2.

B. DenseNet121

The presented Fig. 2 illustrates a modified DenseNet121 architecture developed for multi-class classification of tomato leaf diseases. The model is designed to identify 10 categories: Tomato bacterial spot, early blight, late blight, leaf mold, Septoria leaf spot, spider mites, target spot, yellow leaf curl virus, mosaic virus and healthy leaves.

Input layer: The network accepts RGB tomato leaf images resized to $224 \times 224 \times 3$ pixels. Prior to feeding the images into the network, pixel values are normalised to accelerate convergence and stabilise training. The input tensor is mathematically represented as:

$$X_0 \in R^{224 \times 224 \times 3} \quad [12]$$

Initial convolution and pooling: The first stage of DenseNet121 applies a 7×7 convolution with stride 2, followed by batch normalisation and ReLU activation:

$$X_1 = \operatorname{ReLU}(\operatorname{BN}(W_1 * X_0 + b_1)) \quad [13]$$

This is followed by a 3×3 max-pooling layer with stride 2, which reduces the spatial resolution while preserving salient features.

Dense blocks: DenseNet121 comprises 4 dense blocks with layer configurations {6, 12, 24, 16} and a growth rate $k=32$. Each dense block follows a dense connectivity pattern, where each layer receives feature maps from all preceding layers:

$$x_i = H_i([x_0, x_1, \dots, x_{i-1}]) \quad [14]$$

Table 2. Summary for the MobileNetV2 architecture flow

Stage	Operation	Purpose
Input	(224 \times 224 \times 3)	Image preprocessing
Conv2D	Spatial feature extraction	Detect texture/edges
Depth wise separable Conv	Efficient multi-channel filtering	Reduce computation
Dropout	Regularisation	Prevent overfitting
Global average pooling	Dimensionality reduction	Spatial summarisation
Dense (2 layers)	Feature combination	High-level representation
Softmax	Probability output	10-class classification

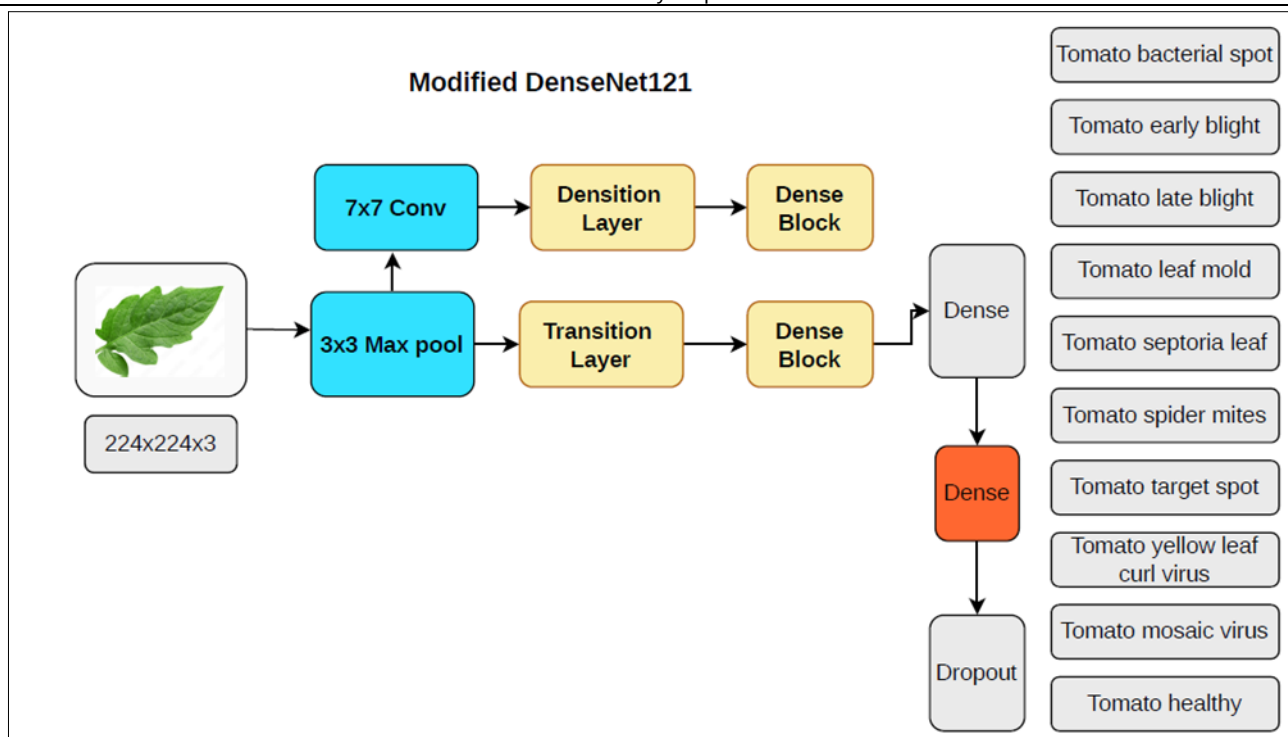


Fig. 2. Architecture of DenseNet121.

Here, $H(\cdot)$ denotes a composite operation of batch normalisation, ReLU activation, 1×1 convolution and 3×3 convolution. This dense connectivity promotes feature reuse, improves gradient flow and enhances parameter efficiency.

Transition layers: Transition layers are inserted between dense blocks to control model complexity and feature dimensionality. Each transition layer performs:

1×1 Convolution to reduce channel dimensionality

2×2 Average pooling to reduce spatial dimension

$$X_{t+1} = \text{AvgPool}(\text{Conv}_{1 \times 1}(X_t)) \quad [15]$$

This design helps maintain compact and discriminative feature representations.

Global average pooling: After the final dense block, GAP is applied to reduce the spatial dimensions ($7 \times 7 \rightarrow 1 \times 1$) while retaining semantic information:

$$z_c = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W X_{ijc} \quad [16]$$

This operation converts each feature map into a single representative value, reducing overfitting and model complexity.

Custom classification head: The original DenseNet121 classifier is replaced with a task-specific classification head tailored for tomato disease detection. It consists of:

A fully connected dense layer with 1024 units and ReLU activation:

$$h = \text{ReLU}(W_h z + b_h) \quad [17]$$

A dropout layer with a rate of 0.2 to mitigate overfitting.

A final dense layer with 10 units and Softmax activation to generate class probabilities:

$$\hat{y}_i = \frac{e^{z_i}}{\sum_{j=1}^{10} e^{z_j}} \quad [18]$$

Objective function: Training employs categorical cross-entropy loss:

$$L = - \sum_{i=1}^{10} y_i \log(\hat{y}_i) \quad [19]$$

Where y_i and $\hat{y}_i$ denote the ground truth and predicted class probabilities respectively.

C. MobileNetV2–DenseNet121 Ensemble Architecture

This research presents a robust deep learning ensemble framework that leverages the complementary strengths of MobileNetV2 and DenseNet121 for accurate tomato leaf disease detection, as illustrated in Fig. 3. The proposed architecture integrates lightweight and densely connected feature extraction through a feature-level concatenation strategy, enabling the model to capture both high-level semantic information and fine-grained visual patterns from tomato leaf images. The overall workflow of the proposed system, including data preprocessing, feature extraction, feature fusion and classification, is described in the following sections.

Dataset description: The proposed system is evaluated using the Plant Village dataset, a publicly available benchmark dataset containing images of healthy and diseased plant leaves. In this study, the tomato subset of the dataset is considered, comprising multiple disease categories such as early blight, late blight, leaf mold, septoria leaf spot, bacterial spot, as well as

healthy leaf samples. All images are provided in RGB format and are resized to a uniform resolution of $224 \times 224 \times 3$ to ensure compatibility with the pre-trained deep neural network backbones used in this work.

Image preprocessing: Preprocessing is a crucial step that enhances model efficiency and ensures consistent input representation. Some of the preprocessing images are shown in Fig. 4. The RGB images are normalised by scaling the pixel intensity values to the range $[0, 1]$ using the formula: $X_{\text{norm}} = X/255$, where X represents the original pixel value. Data augmentation techniques such as random rotation, horizontal flip, zoom and shear transformations are applied to increase dataset diversity and minimise overfitting. These augmentations simulate natural variations in leaf orientation, lighting and scale, thus improving the generalisation of the model.

Data splitting strategy: The dataset is divided into three parts: 80 % for training, 10 % for validation and 10 % for testing. The training set is used for model learning; the validation set monitors generalisation and prevents overfitting and the test set evaluates final model performance. This 80–10–10 split ensures both adequate training samples and reliable model assessment.

Feature extraction networks: Two distinct CNN architectures are employed in parallel: MobileNetV2 and DenseNet121.

MobileNetV2 is a lightweight network optimised for mobile and embedded vision applications. It employs depth wise separable convolutions and inverted residual blocks to reduce computational cost while maintaining representational power.

DenseNet121, on the other hand, consists of dense blocks where each layer receives input from all preceding layers, promoting feature reuse and strong gradient flow. This structure enables efficient learning of deep hierarchical features.

Both networks are initialised with pre-trained ImageNet weights to leverage transfer learning, followed by fine-tuning on the tomato leaf dataset. The final classification layers of both networks are removed and GAP is applied to extract compact feature representations:

$$F_1 = \text{GAP}(\text{MobileNetV2_output}),$$

$$F_2 = \text{GAP}(\text{DenseNet121_output}).$$

Feature concatenation ensemble mechanism: To exploit the complementary nature of features from both networks, a feature concatenation ensemble is performed. The feature vectors F_1 and F_2 are concatenated along the feature dimension: $F = [F_1; F_2]$, where “[;]” denotes concatenation. This combined vector F captures both lightweight spatial representations and deeply connected texture features, leading to enhanced discriminative capacity for disease classification.

Classification layer design: The concatenated feature vector F is passed through a classification head composed of:

Batch normalisation (BN): Stabilises training and accelerates convergence.

Dropout (rate = 0.5): Reduces overfitting by randomly deactivating neurons during training.

Dense layer: Fully connected layer that maps the concatenated features to the number of disease classes.

Softmax activation: Produces class probability scores.

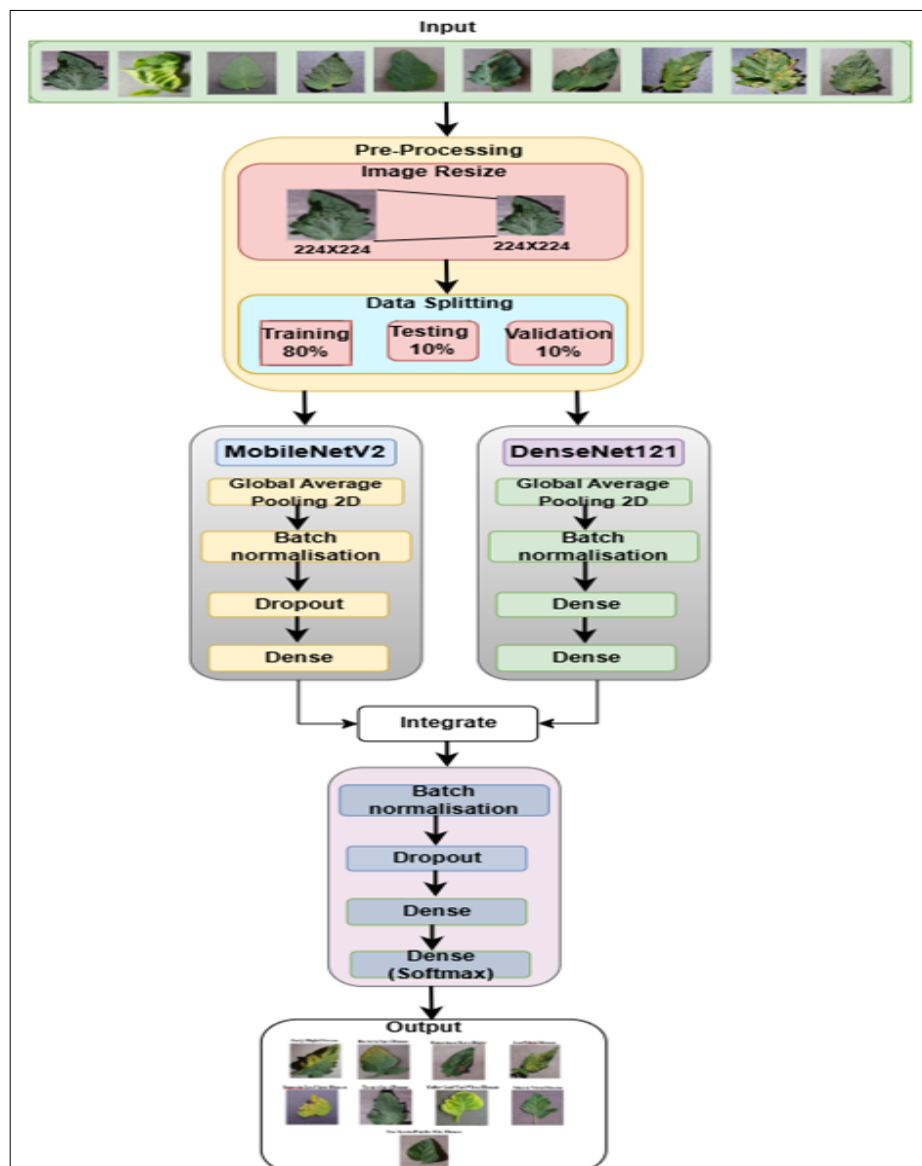


Fig. 3. MobileNetV2–DenseNet121 Ensemble Architecture.

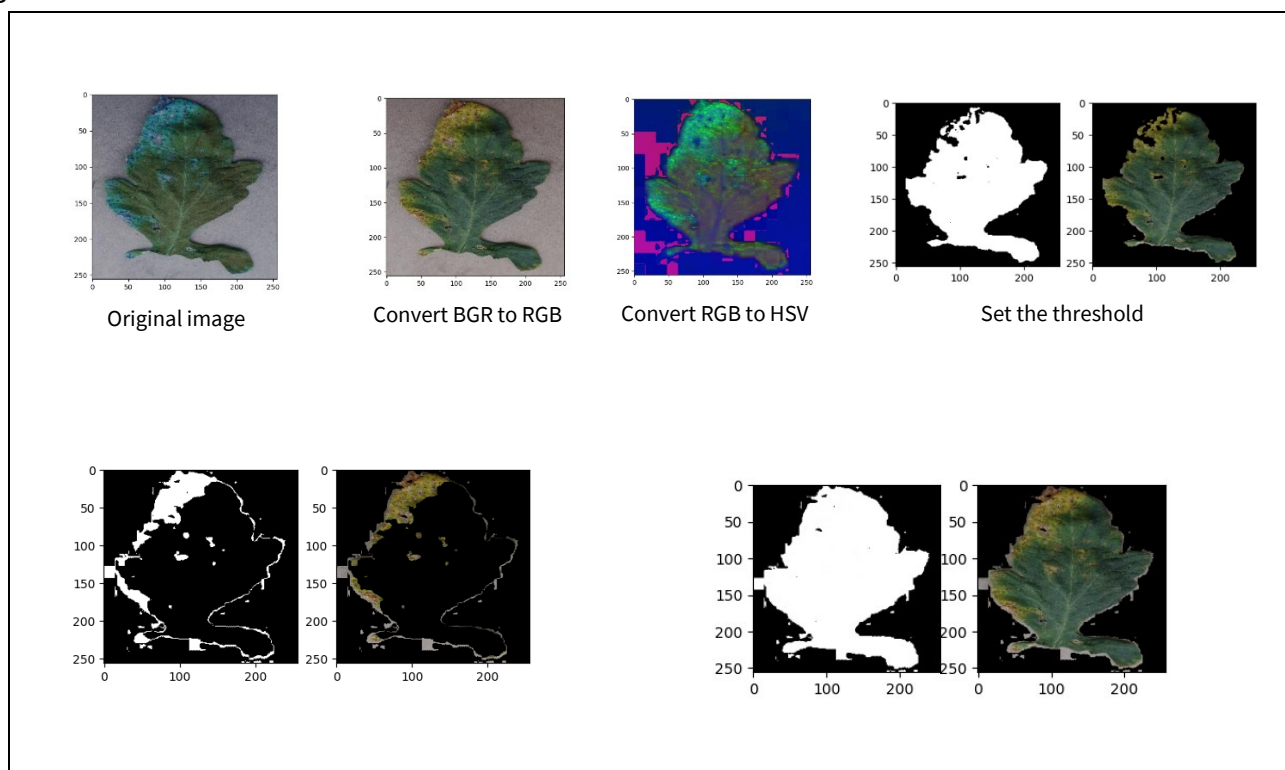


Fig. 4. Preprocessing of tomato leaf disease.

The final class probabilities are computed as:

$$\text{softmax}(z_i) = e^{z_i} / \sum e^{z_i},$$

where z_i is the logit value corresponding to class i .

Mathematical formulation: Let X denote the input image and $f_1(X)$, $f_2(X)$ be the feature extraction functions of MobileNetV2 and DenseNet121 respectively. The ensemble feature vector can be expressed as:

$$F = \text{concat}(f_1(X), f_2(X)) \quad [19]$$

The dense layer computes output logits as:

$$Z = W \cdot F + b \quad [20]$$

where W and b denote the trainable weight matrix and bias vector.

The softmax activation then yields the predicted probability distribution:

$$P(y_i | X) = e^{Z_i} / \sum e^{Z_i} \quad [21]$$

The network is optimised using categorical cross-entropy loss, defined as:

$$L = -\sum y_i \log(P(y_i | X)) \quad [22]$$

where y_i represents the true label (one-hot encoded vector) and $P(y_i | X)$ is the predicted probability for class i .

Output and Prediction Stage: The output layer produces the class label corresponding to the highest probability among disease categories. The final classification result is expressed as:

$$\hat{y} = \text{argmax}(P(y_i | X)) \quad [23]$$

This result indicates the predicted tomato leaf disease class or “Healthy” category.

Performance evaluation: Model performance is assessed using key metrics such as accuracy, precision, recall and F1-score, which provide insights into the classifier’s reliability. These are defined as:

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN)$$

$$\text{Precision} = TP / (TP + FP)$$

$$\text{Recall} = TP / (TP + FN)$$

$$\text{F1-score} = 2 \times (\text{precision} \times \text{recall}) / (\text{precision} + \text{recall})$$

is shown in the Fig. 5. The dataset was balanced across classes and model performance was assessed using accuracy, precision and F1-score.

Training and validation performance: To analyse learning behaviour and generalisation, training and validation accuracies were monitored over 200 epochs. The summarised results are presented in Table 3.

The accuracy curves (Fig. 6–9) indicate that the ensemble model achieves smoother and more stable convergence compared to the individual models. Although both MobileNetV2 and DenseNet121 reached high training accuracy, their validation performance showed mild fluctuations, highlighting the benefit of feature-level fusion in improving generalisation.

Quantitative performance evaluation: A comparative analysis of accuracy, precision and F1-score is reported in Table 4.

The ensemble model achieved the highest performance across all metrics, with an accuracy of 99.45 %. This improvement is attributed to the complementary nature of the two backbones, where MobileNetV2 captures efficient low-level features and DenseNet121 provides rich hierarchical representations. Fig. 10 presents the comparative performance of all models.

Confusion matrix analysis: The confusion matrix of the ensemble model across the 10 tomato leaf classes is illustrated in Fig. 11. The strong concentration of values along the diagonal indicates high classification accuracy for all categories. Each class achieved over 99 % accuracy, with only a few misclassifications observed between visually similar diseases such as early blight, late blight and leaf mold. These minor errors can be attributed to overlapping visual characteristics in leaf texture and lesion patterns.

The few observed misclassifications primarily between diseases with overlapping visual patterns can be attributed to the inherent similarities in leaf texture and lesion colour. Nevertheless, the ensemble model substantially reduced class overlap compared to individual base learners, confirming that feature concatenation between MobileNetV2 and DenseNet121 enhances discriminative representation. Overall, the ensemble model achieved an accuracy of 99.45 %, a precision of 98.8 % and an F1-score of 99.1 %, signifying superior generalisation and robustness for real-world tomato disease detection applications.

Results

The proposed models were evaluated on the tomato leaf disease dataset, which includes ten classes: Tomato bacterial spot, early blight, late blight, leaf mold, septoria leaf spot, target spot, yellow leaf curl virus, mosaic virus, spider mites and healthy leaves, which

Discussion

The experimental findings highlight the practical advantages of combining lightweight and densely connected architectures for plant disease detection. While MobileNetV2 offers computational efficiency suitable for resource-constrained environments,

Table 3. Training and validation performance

Model	Training accuracy (%)	Validation accuracy (%)	Remarks
MobileNetV2	99.0	90.8	The model converges quickly but shows a gap,
DenseNet121	99.1	94.3	Improved feature extraction and generalisation over
Ensemble (MobileNetV2 + DenseNet121)	99.8	96.9	Achieves best overall performance with reduced overfitting and enhanced robustness.

Table 4. Quantitative performance for all three models

Model	Accuracy (%)	Precision (%)	F1-score (%)
MobileNetV2	93.2	91.8	92.4
DenseNet121	95.1	94.7	94.9
Ensemble (MobileNetV2 + DenseNet121)	99.45	98.8	99.1

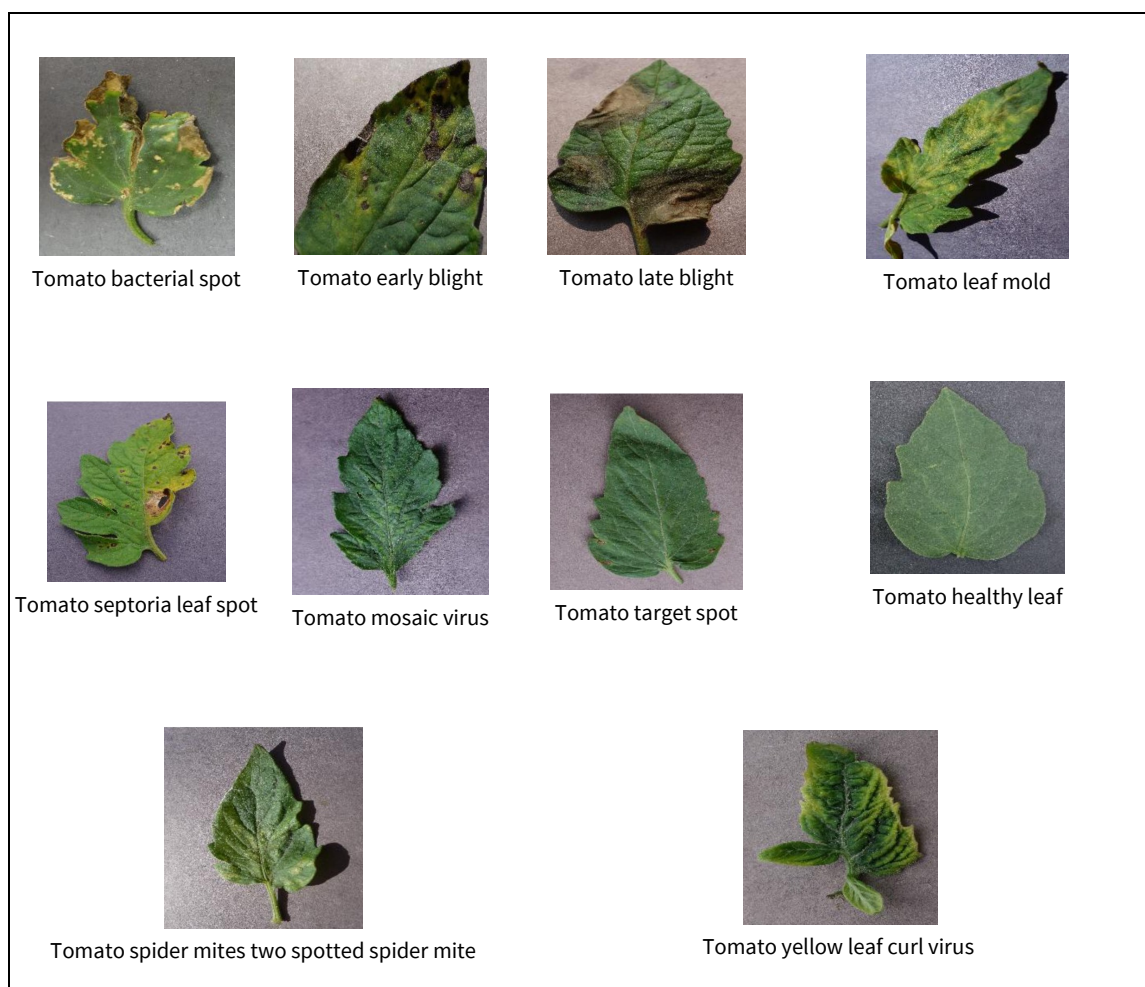


Fig. 5. Different types of tomato leaf diseases.

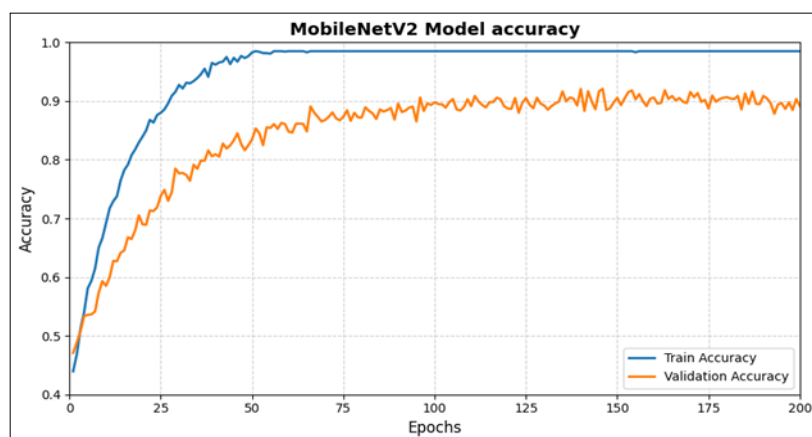


Fig. 6. Performance analysis of MobileNetV2 training and validation accuracy.

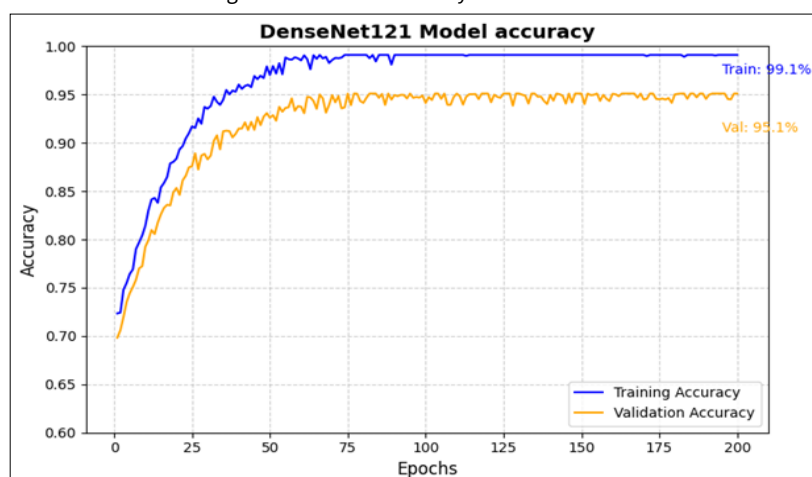


Fig. 7. Performance analysis of DenseNet121 training and validation accuracy.

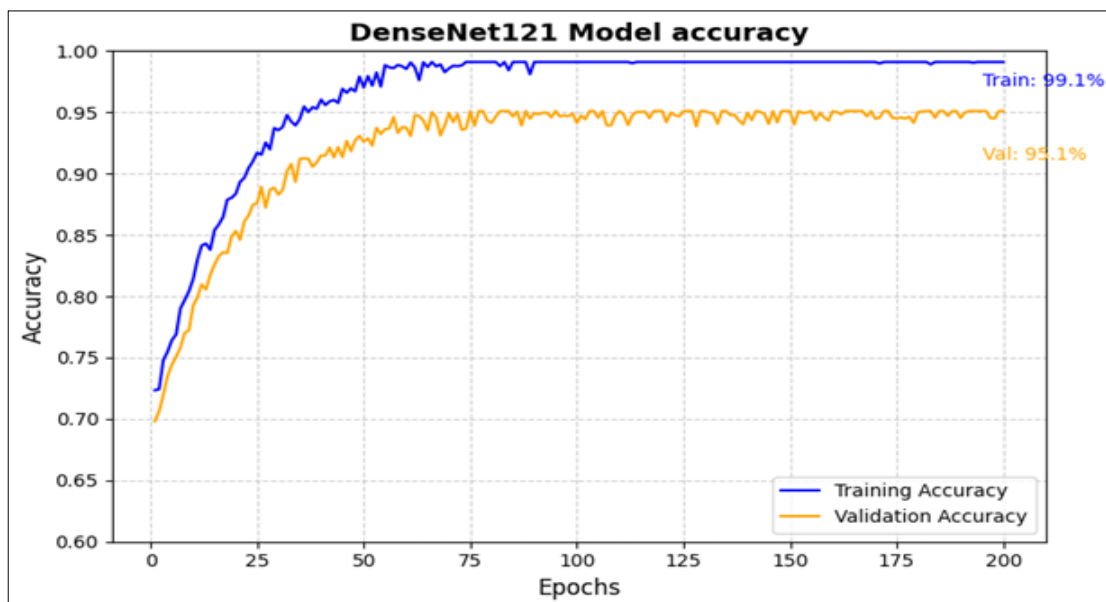


Fig. 8. Performance analysis of ensemble MobileNetV2 and DenseNet121 training and validation accuracy.

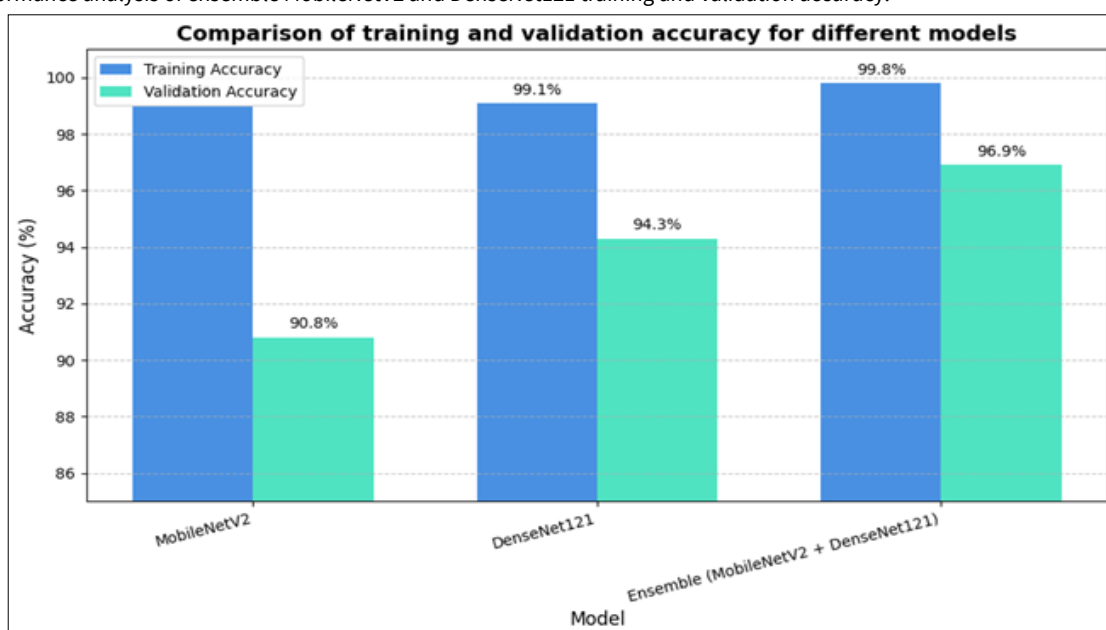


Fig. 9. Comparison of training and validation accuracy for different models.

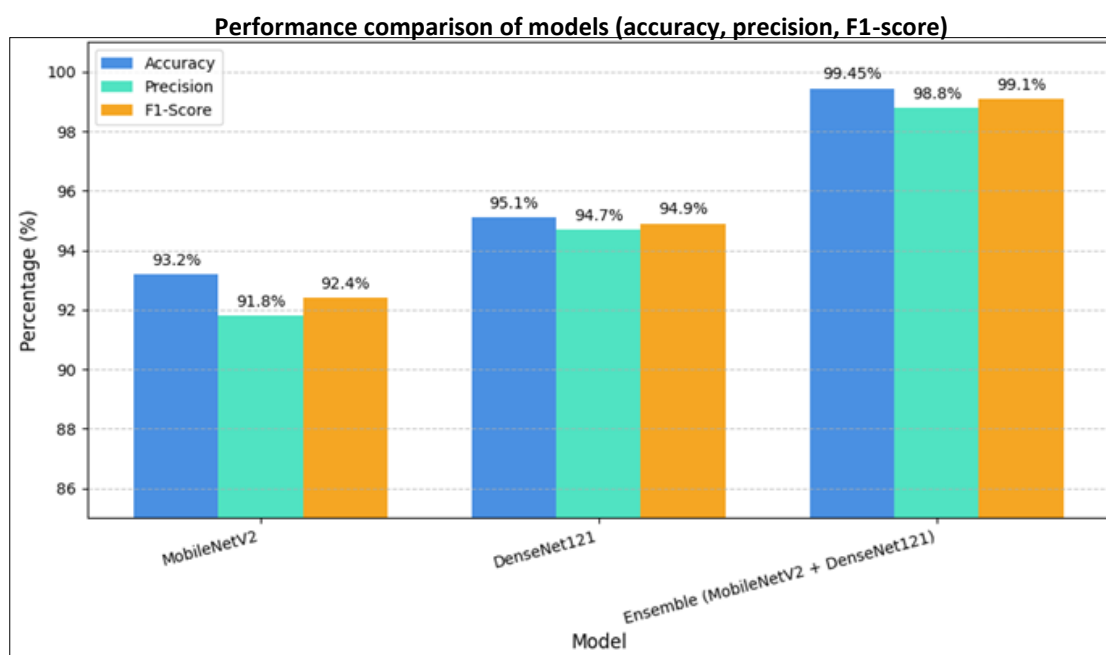


Fig. 10. Performance comparisons for the models.

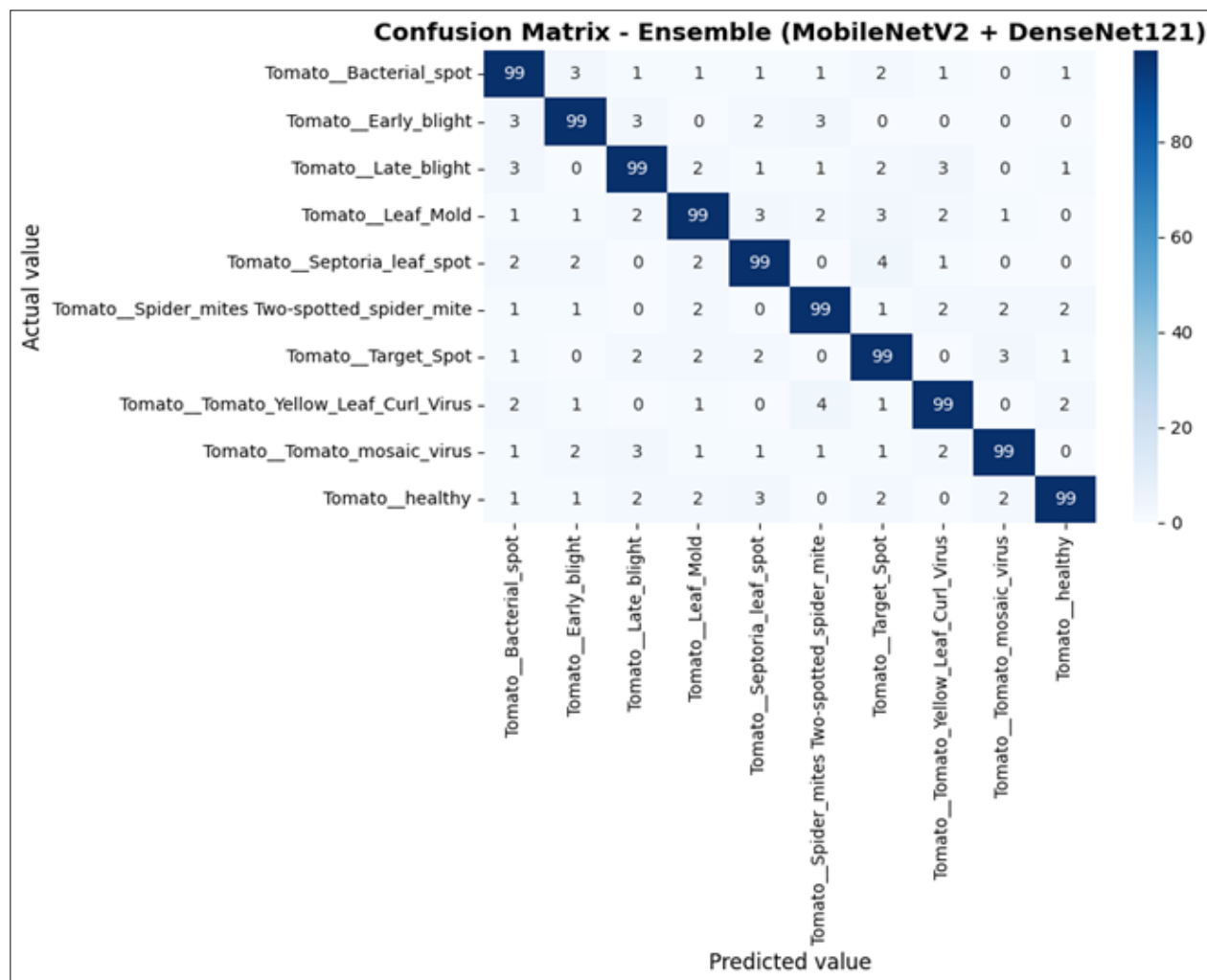


Fig. 11. Confusion matrix for the ensemble (MobileNetV2+DenseNet121).

DenseNet121 provides richer hierarchical feature representations that enhance discriminative capability. The ensemble framework effectively integrates these complementary properties through feature concatenation, leading to improved generalisation and reduced overfitting compared to standalone models. This demonstrates that model diversity at the feature level plays a critical role in capturing both fine-grained visual cues and high-level semantic patterns present in tomato leaf disease images.

From an application perspective, the proposed approach shows strong potential for deployment in real-world agricultural settings, such as mobile-based disease diagnosis or edge-assisted monitoring systems. The high accuracy and robustness across multiple disease categories suggest that the model can support timely decision-making for farmers and agricultural experts. However, the current framework is primarily validated on controlled benchmark datasets and further evaluation under varying illumination conditions, backgrounds and real-field image noise would strengthen its practical reliability.

Conclusion

This study presented an efficient and robust deep learning framework for automated tomato leaf disease detection using MobileNetV2, DenseNet121 and their ensemble integration. By leveraging the complementary strengths of MobileNetV2's lightweight depth wise separable convolutions and DenseNet121's dense feature connectivity, the proposed

ensemble model achieved superior classification performance and improved generalisation. Experimental results showed that while the individual models attained strong accuracy (93.2 % for MobileNetV2 and 95.1 % for DenseNet121), the ensemble significantly outperformed both, achieving 99.45 % accuracy, 98.8 % precision and a 99.1 % F1-score, along with enhanced training validation stability and reduced overfitting. These findings confirm that ensemble-based deep learning approaches can substantially improve the diagnostic precision of plant disease classification systems by integrating diverse feature representations, making the proposed framework well suited for precision agriculture applications such as early disease detection and yield protection.

Future work will extend this framework by incorporating transformer-based vision models, exploring multimodal learning with spectral or thermal data and validating the approach on real-field datasets and edge-device deployments to further enhance robustness and practical applicability.

Authors' contributions

All authors contributed equally.

Compliance with ethical standards

Conflict of interest: Authors do not have any conflict of interests to declare.

Ethical issues: None

References

- Jakubec M, Lieskovská E, Bučko B, Záborská K. Comparison of CNN-based models for pothole detection in real-world adverse conditions: Overview and evaluation. *Applied Sciences*. 2023;13(9):5810. <https://doi.org/10.3390/app13095810>
- Akay M, Du Y, Sershen CL, Wu M, Chen TY, Assassi S, et al. Deep learning classification of systemic sclerosis skin using the MobileNetV2 model. *IEEE Open Journal of Engineering in Medicine and Biology*. 2021;2:104–10. <https://doi.org/10.1109/OJEMB.2021.3066097>
- Albelwi SA. Deep architecture based on DenseNet-121 model for weather image recognition. *International Journal of Advanced Computer Science and Applications*. 2022;13(10). <https://dx.doi.org/10.14569/IJACSA.2022.0131065>
- Sharma J, Al-Huqail AA, Almogren A, Doshi H, Jayaprakash B, Bharathi B, et al. Deep learning based ensemble model for accurate tomato leaf disease classification by leveraging ResNet50 and MobileNetV2 architectures. *Scientific Reports*. 2025;15(1):13904. <https://doi.org/10.1038/s41598-025-98015-x>
- Liu J, Wang X. Early recognition of tomato gray leaf spot disease based on MobileNetV2-YOLOv3 model. *Plant Methods*. 2020;16(1):83. <https://doi.org/10.1186/s13007-020-00624-2>
- Mazumder MKA, Mridha M, Alfarhood S, Safran M, Abdullah-Al-Jubair M, Che D. A robust and light-weight transfer learning-based architecture for accurate detection of leaf diseases across multiple plants using less amount of images. *Frontiers in Plant Science*. 2024;14:1321877. <https://doi.org/10.3389/fpls.2023.1321877>
- Ronihah SS, Selvam S, editors. Enhancing crop yield: A deep learning approach to detect tomato leaf diseases. In: *Proceedings of the 2025 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE)*; 2025. IEEE.
- Sutaji D, Yıldız O. LEMOXINET: Lite ensemble MobileNetV2 and Xception models to predict plant disease. *Ecological Informatics*. 2022;70:101698. <https://doi.org/10.1016/j.ecoinf.2022.101698>
- Mahabub S, Jahan I, Hasan MN, Islam MS, Akter L, Musfiqur M, et al. Efficient detection of tomato leaf diseases using optimized compact convolutional transformers (CCT) model. *Magna Sci Adv Res Rev*. 2024;12:39–53. <https://doi.org/10.30574/msarr.2024.12.2.0183>
- Ariawan KR, Ekayana AAG, Indrawan IPY, Winatha KR, Setiawan INAF. Performance comparison of DenseNet-121 and MobileNetV2 for cacao fruit disease image classification. *Indonesian Journal of Data and Science*. 2025;6(1):29–37. <https://doi.org/10.56705/ijodas.v6i1.233>
- Tamim MU, Hamim SA, Malik S, Mridha M, Mahmood S. InsightNet: A deep learning framework for enhanced plant disease detection and explainable insights. *Plant Direct*. 2025;9(5):e70076. <https://doi.org/10.1002/pld3.70076>
- Zubair F, Saleh M, Akbari Y, Al Maadeed S. A robust ensemble model for plant disease detection using deep learning architectures. *AgriEngineering*. 2025;7(5):159. <https://doi.org/10.3390/agriengineering7050159>
- Oni MK, Prama TT. Optimized custom CNN for real-time tomato leaf disease detection. *arXiv preprint*. 2025;arXiv:250218521. <https://doi.org/10.48550/arXiv.2502.18521>
- Duhan S, Gulia P, Gill NS, Narwal E. RTR_Lite_MobileNetV2: A lightweight and efficient model for plant disease detection and classification. *Current Plant Biology*. 2025;42:100459. <https://doi.org/10.1016/j.cpb.2025.100459>
- Zhang E, Zhang N, Li F, Lv C. A lightweight dual-attention network for tomato leaf disease identification. *Frontiers in Plant Science*. 2024;15:1420584. <https://doi.org/10.3389/fpls.2024.1420584>
- Alzahrani MS, Alsaade FW. Transform and deep learning algorithms for the early detection and recognition of tomato leaf disease. *Agronomy*. 2023;13(5):1184. <https://doi.org/10.3390/agronomy13051184>
- Pushpa B, Ashok A, AV SH, editors. Plant disease detection and classification using deep learning model. In: *Proceedings of the 3rd International Conference on Inventive Research in Computing Applications (ICIRCA)*; 2021. IEEE. <https://doi.org/10.1109/ICIRCA51532.2021.9544729>
- Zhang X, Han L, Dong Y, Shi Y, Huang W, Han L, et al. A deep learning-based approach for automated yellow rust disease detection from high-resolution hyperspectral UAV images. *Remote Sensing*. 2019;11(13):1554. <https://doi.org/10.3390/rs11131554>
- Bhandari M, Shahi TB, Neupane A, Walsh KB. Botanicx-ai: Identification of tomato leaf diseases using an explanation-driven deep-learning model. *Journal of Imaging*. 2023;9(2):53. <https://doi.org/10.3390/jimaging9020053>
- Demilie WB. Plant disease detection and classification techniques: A comparative study of the performances. *Journal of Big Data*. 2024;11(1):5. <https://doi.org/10.1186/s40537-023-00863-9>
- Pai P, Amutha S, Patil S, Shobha T, Basthikodi M, Shafeeq BA, et al. Deep learning-based automatic diagnosis of rice leaf diseases using ensemble CNN models. *Scientific Reports*. 2025;15(1):27690. <https://doi.org/10.1038/s41598-025-13079-z>
- Ashok S, Kishore G, Rajesh V, Suchitra S, Sophia SG, Pavithra B, editors. Tomato leaf disease detection using deep learning techniques. In: *Proceedings of the 5th International Conference on Communication and Electronics Systems (ICCES)*; 2020. IEEE. <https://doi.org/10.1109/ICCES48766.2020.9137986>

Additional information

Peer review: Publisher thanks Sectional Editor and the other anonymous reviewers for their contribution to the peer review of this work.

Reprints & permissions information is available at https://horizonpublishing.com/journals/index.php/PST/open_access_policy

Publisher's Note: Horizon e-Publishing Group remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Indexing: Plant Science Today, published by Horizon e-Publishing Group, is covered by Scopus, Web of Science, BIOSIS Previews, Clarivate Analytics, NAAS, UGC Care, etc
See https://horizonpublishing.com/journals/index.php/PST/indexing_abstracting

Copyright: © The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution and reproduction in any medium, provided the original author and source are credited (<https://creativecommons.org/licenses/by/4.0/>)

Publisher information: Plant Science Today is published by HORIZON e-Publishing Group with support from Empirion Publishers Private Limited, Thiruvananthapuram, India.