Edge optimized multimodal cross fusion model with statistical validation for multi crop disease detection

Thomas Kinyanjui Njoroge [a,1,*], Kelvin Mugoye Sindu [b,2], Rachael Kibuku [b,3]

[a] Department of Computer Science and Informatics, School of Pure and Applied Science, Karatina University, Kenya
[b] Software Development & Information Systems (SD&IS) Department, School of Technology, KCA University, Kenya
[1] tnjoroge@karu.ac.ke; [2] kmugoye@kcau.ac.ke; [3] rkibuku@kcau.ac.ke
* corresponding author

ARTICLE INFO

Article history
Received April 1, 2025
Revised June 13, 2025
Accepted June 27, 2025
Available online August 31, 2025

Keywords
Crop disease detection
Multimodal fusion model
Transfer learning
Edge computing
MobileNetV2

ABSTRACT

Accurate and timely crop disease detection is crucial for mitigating agricultural losses and ensuring food security, particularly in resource-limited settings. Traditional diagnostic methods are inefficient and prone to errors, while existing deep learning models, such as ResNet50 and Inception V3, struggle with generalizability and computational efficiency. This study proposes a Dynamic Edge-Optimized Multimodal Fusion (DEMF) model, integrating EfficientNetV2 and MobileNetV2 to enhance feature learning and scalability. The model was trained on a 76-class dataset comprising PlantVillage and locally collected images of crop diseases, ensuring robustness across diverse conditions. Feature fusion via concatenation, combined with compound scaling and transfer learning, enabled the model to capture fine-grained patterns of disease. Extensive experiments, including ablation studies and comparative evaluations against DenseNet-121, DenseNet-50, AlexNet, and ResNet-152, validated the model's superiority. The proposed model achieved 99.2% accuracy, a Kappa of 0.9919, and an AUC of 0.9999, outperforming benchmarks. Statistical validation confirmed significant improvements ($p<0.05$) and stability. To enhance accessibility, an AI-powered mobile application was deployed on the Google Play Store, enabling real-time disease detection and actionable recommendations. To enhance accessibility, an AI-powered mobile application was deployed on the Google Play Store, enabling real-time disease detection and actionable recommendations. This research advances transfer learning, feature fusion, and statistical validation for robust, scalable crop disease detection in low-resource environments.

1. Introduction

Recent technological advancements have enhanced disease detection through machine learning (ML), computer vision, and deep learning (DL) innovations, transforming the domain and making disease identification precise and effective [1] demonstrated that these modern techniques use sophisticated algorithms to analyze plant leaf photographs, enabling more accurate detection of disease patterns. Given the challenges of distinguishing between similar plant diseases, integrating image processing with ML and computer vision is crucial for achieving reliable diagnoses [2], [3]. DL has made remarkable strides in image classification, achieving notable accuracy and speed in detecting and classifying crop leaf diseases. Despite its advancements, Abdu et al. [4] illustrated that DL methods face challenges, like the need for extensive training data. Transfer learning (TL) has been demonstrated to

be a powerful answer to these limitations, enhancing learning efficiency by leveraging knowledge from pre-trained models [5], [6].

Accurate and timely crop disease detection is critical for mitigating agricultural losses and ensuring food security. This challenge is especially acute in low-resource settings, where traditional diagnostic methods are often inefficient and error-prone. While machine learning (ML), computer vision, and deep learning (DL) have revolutionized disease identification by enabling precise analysis of plant leaf images, significant hurdles remain. Distinguishing between visually similar diseases as demonstrated by Kaleem et al. [1] is inherently difficult, and advanced DL models typically demand large volumes of training data while often struggling with computational efficiency and generalization to new conditions. Transfer learning (TL) has emerged as a powerful strategy to address data scarcity and efficiency constraints. By leveraging knowledge from pre-trained models, as implemented by Ishaq et al. [7] .TL enhances learning efficiency, reduces overfitting, lowers computational costs, and improves performance on smaller datasets, making it particularly suitable for practical, resource-constrained applications. However, collecting and curating diverse, high-quality image datasets remains a significant practical barrier as illustrated by Adilah and Kristiyanti [8] and other previous researchers [9]–[16].

A critical limitation of many current approaches is their static nature. Models trained on fixed datasets frequently lack adaptability, leading to degraded performance when encountering novel disease variants, changing environmental conditions, or different imaging scenarios as also elaborated by Amin et al. [17]. This underscores a pressing need for dynamic detection methods capable of evolving with new data and contexts. Techniques like data regularization and augmentation are essential for building robust, generalizable dynamic systems as implemented by Sun et al. [18]. Despite these technological advances, a significant research gap persists for real-world deployment in low-resource settings: Existing models often fail to effectively capture both subtle local features (low-range dependencies) and broader contextual patterns (long-range dependencies) in agricultural imagery. Furthermore, they frequently struggle to achieve the necessary balance between high diagnostic accuracy and the computational efficiency required for deployment on edge devices common in these environments. Rigorous statistical validation of model robustness and stability is also often lacking. This study addresses these critical gaps by proposing a novel model that leverages the globally recognized PlantVillage dataset augmented with locally sourced images to enhance robustness. The core contributions of this paper are.

2. Literature Review

Effective plant disease identification is vital for precision agriculture, impacting plant health and productivity. Accurate plant disease detection is critical for sustainable agriculture, where timely identification minimizes economic losses. While traditional methods are labor-intensive and subjective [19], [20], computer vision and DL have transformed disease recognition through automated image analysis [21]. CNNs has demonstrate particular promise, achieving >95% accuracy on benchmark datasets like PlantVillage [22], yet high-performance models like ResNet-50 incur prohibitive computational costs (>200MB) that hinder edge deployment [23] With increasing disease outbreaks, timely detection is crucial for diagnosis, control, and damage assessment. Early identification allows targeted treatments and prevents significant economic losses. Lightweight alternatives such as MobileNet variants [24], [25] address efficiency through depthwise separable convolutions but sacrifice precision in complex field conditions with overlapping symptoms [26]. Hybrid approaches show potential but face critical constraints: fusion models (e.g., MobileNetV2 integrations [27] and and XDNet [28] enhance feature representation but lack edge optimization; attention mechanisms (C-DenseNet [29], coordinate attention [30]) improve focus but neglect cross-crop generalization; and EfficientNet variants achieve high accuracy [31] but omit computational efficiency analysis. Meanwhile, edge/IoT solutions [32], Addu et al. [4] describe key metrics that include disease incidence (proportion of affected plants), severity (extent of damage), and consequence (impact on yield). Research by [33], [34] enable field accessibility but introduce latency and connectivity dependencies, while transfer learning adaptations [35] remain crop-specific. These limitations reveal persistent gaps in balancing accuracy with

efficiency, enabling cross-crop generalization, ensuring edge compatibility, and providing rigorous validation - challenges summarized in Table 1. References [36], [37] classifiers detect diseases by comparing extracted features with categorized datasets. DL models offer improved accuracy by processing large datasets and identifying subtle patterns [38], [39]. These advancements facilitate more effective disease segmentations on images, crucial for precise diagnosis and treatment. Barman *et al.* [19] demonstrated the potential of CNNs for classifying diseases in crops like tomatoes and potatoes using the PlantVillage dataset, achieving accuracies above 95%. However, bulkier architectures, as shown in the works of such as Padshetty and Ambika [20], ResNet-50, while accurate, suffer from high computational costs (>200 MB model size), making them impractical for edge deployment [8], [40]. However, Tragoudaras et al. [41] points out that these models still face accuracy-efficiency trade-offs in complex agricultural environments with overlapping disease symptoms [22]. Similarly, studies indicate that multimodal fusion involves integrating different data types or features from multiple sources [23], [25]. This is crucial for improving the reliability and precision of disease detection. This is further supported by [26]–[29], [42]–[47].

Table 1. Summary of Deep Learning Models for Crop Disease Detection

Ref.	Model	Deployment Type	Model Size	Dataset Used	Attained Performance	Limitations
[19]	CNN-based model	Cloud-only	Large (>100MB)	PlantVillage	>95% accuracy	High computational cost
[52]	EfficientNetV2L	Cloud	Very Large	PlantVillage	99.6% accuracy	Lacks computational efficiency analysis
[28]	MobileNetV3 with coordinate attention	Mobile	Small (<20MB)	Custom dataset	Improved efficiency	Trade-off between efficiency and accuracy not fully analyzed
[53]	EfficientNet-B7	Cloud	Very Large	Public agricultural dataset	Optimized training speed	High resource consumption
[54]	Pruned CNN model	Mobile	Small	Multi-crop dataset	Enhanced efficiency	Impact on accuracy not fully examined
[31]	IoT-cloud CNN model	Hybrid (IoT + Cloud)	Large	IoT-collected crop images	High classification accuracy	Latency issues due to cloud dependence
[20]	Edge AI model	Edge	Optimized	Field images	Optimized for edge devices	Memory constraints on and no validation
[55]	IoT-integrated deep learning	Hybrid (IoT + Cloud)	Variable	Smart agriculture dataset	Enhanced disease detection	Data inconsistencies in IoT sensors
[56]	AI-driven mobile application	Mobile	Small	Agricultural dataset	Improved accessibility	Internet dependency
[32]	Mobile AI model	Mobile	Small	Real-world crop images	High detection rates	High data costs and connectivity issues

XDNet's approach to fusing these architectures leverages DenseNet's feature extraction strengths [48]. The combination enables XDNet to handle diverse disease patterns with high precision [28] explored an improved version of SqueezeNet using the PlantVillage dataset. SqueezeNet's design focused on minimizing the variables while maintaining competitive accuracy, achieved through its fire modules that combine convolutions and pooling operations [49] incorporated a coordinate attention mechanism into MobileNet to boost the Model performance while reducing its parameter count. MobileNetV2, known for its efficient depthwise separable convolutions, was further improved with coordinate attention, which allowed the Model to focus on spatially significant regions. This combination helped MobileNetV2 to capture disease-related features in plant images better [50] proposed an improved VGG16 grape disease detection model incorporating transfer learning [17], [35], [50], [51].

Despite significant advancements in plant disease detection through CNNs and transfer learning, several limitations remain, as outlined in Table 1. Many existing models focus on single-crop classification, limiting their generalizability across diverse agricultural settings. Others lack the computational efficiency required for real-time deployment, fail to incorporate multimodal fusion techniques or omit rigorous statistical validation. Furthermore, while optimization strategies have improved CNN performance, there is limited exploration of dynamic edge-optimized architectures tailored for low-resource environments. This paper addresses these critical gaps by proposing an adaptive, multimodal fusion framework optimized for real-time disease detection in resource-constrained agricultural settings.

3. Method

This section details the design and implementation of the hybrid deep-learning framework. We first present the architectural components of the model, emphasizing its dual-branch multimodal design that synergizes CNNs for localized and global feature extraction. Next, we describe the feature extraction process. The proposed framework is evaluated through rigorous ablation studies, benchmark comparisons, and field trials, with implementation details provided for reproducibility.

3.1. Proposed Model Architecture

As shown in Fig. 1, the proposed architecture comprises two parallel branches: EfficientNetV2 and MobileNetV2, each designed to optimize input data through tailored preprocessing steps. The ReLU (Rectified Linear Unit) activation function is employed in the convolutional layers to introduce non-linearity, effectively addressing the vanishing gradient problem. The architecture incorporates compound scaling, uniformly scaling network depth, width, and resolution to balance accuracy and computational efficiency. This approach ensures consistent performance across all layers, maintaining a stable trade-off between feature extraction quality and computational demands. The outputs from both branches are fed into dedicated, fully connected layers. These outputs are concatenated to form a combined feature set, allowing the Model to integrate unique feature representations from EfficientNetV2 and MobileNetV2. The concatenation preserves the full feature space from each branch, resulting in a comprehensive representation for downstream classification tasks. This concatenated output is processed and fine-tuned through an additional set of fully connected layers and a SoftMax layer for classification, producing the final output for all classes.

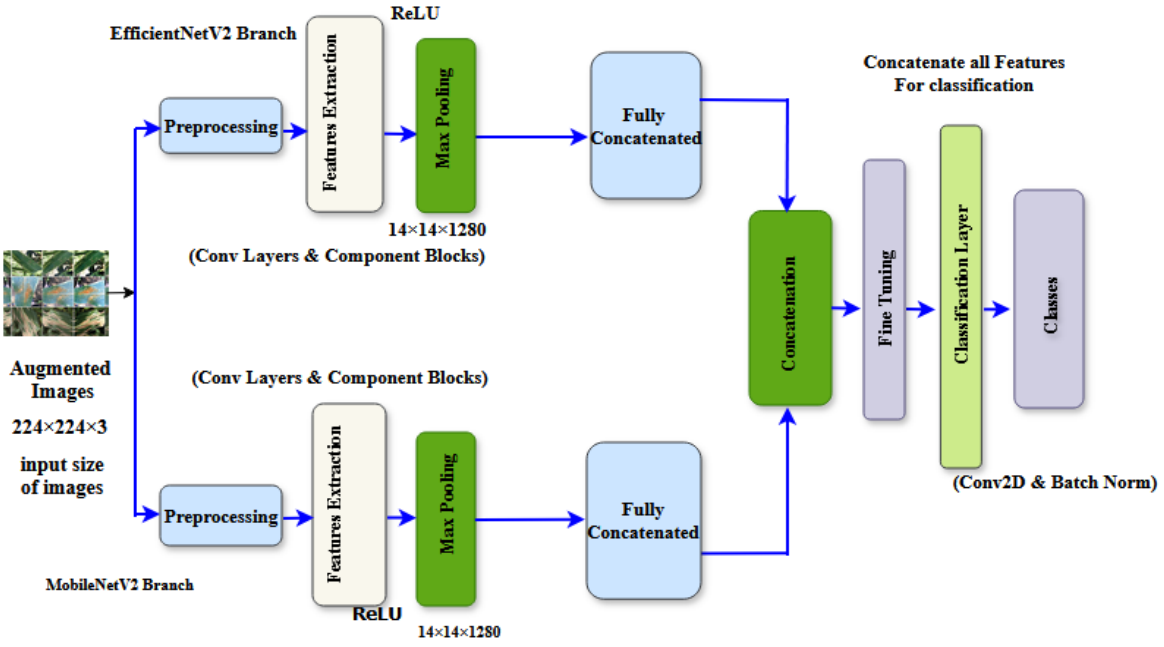


Fig. 1. Proposed Architecture

3.2. Dataset Preparation and Preprocessing

This section presents the dataset preparation and preprocessing techniques employed in this study. It outlines the steps followed, data sources, preprocessing steps, and strategies to enhance model performance and ensure robust feature extraction:

3.2.1. Step 1: Loading and Organizing Data

The dataset was loaded from a directory structure where images were organized in subfolders based on their class labels. Each subfolder represented a distinct disease category, ensuring efficient label assignment during model training.

- Step 1: Data Preparation and Augmentation

The image dataset was organized in class-specific folders, enabling automatic label assignment during loading. Each image was paired with a one-hot encoded label vector $y \in \{0,1\}^{\wedge} \wedge C$ where C is the number of classes. One-hot encoding is defined as

$$\text{one\_hot}(y) = \begin{cases} 1 & \text{if class } = y \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

To improve generalization, data augmentation was applied dynamically during training. This included random rotations, flips, brightness scaling (0.7–1.3), and zooming to 224×224 resolution as shown in Table 2.

Normalization of pixel values was also performed as shown:

$$\text{Normalized\_pixel } = \frac{\text{pixel\_value}}{255.0} \quad (2)$$

An 80-20 training-validation split was performed using stratified sampling to preserve class distribution as shown :

$$\text{Train\_size } = \left(\frac{\text{number of samples in class}}{\text{total samples}}\right) \times \text{ total samples } \times (1 - \text{ test\_size }) \quad (3)$$

Table 2. Data Augmentation

Transformation Type	Range/Details
Rotation	0°, 90°, 180°, or 270°
Flipping	Horizontal flip, Vertical flip
Brightness Adjustment	Between 0.7 (dark) and 1.3 (bright)
Zoom	Resizing and cropping to 224×224 pixels

The resulting sample augmented images in Fig. 2 demonstrate the transformations applied during preprocessing.

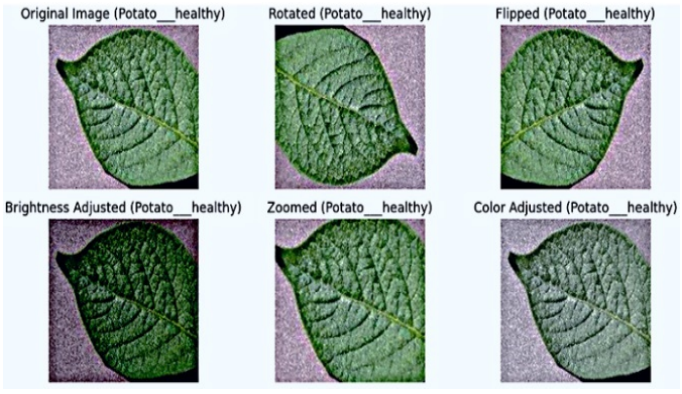


Fig. 2. Transformations applied during preprocessing

3.2.2. Step 2: Categorical Labeling and One-Hot Encoding

The model employed a dual-input architecture, integrating MobileNetV2 and EfficientNetV2B0, both pre-trained on ImageNet. Each image input I had a Shape 224×224×3. Feature extraction produced outputs as shown:

$$F_1 = \text{MobileNetV2}(I_1), F_2 = \text{EfficientNetV2B0}(I_2) \tag{4}$$

Global Average Pooling (GAP) was applied to compress spatial dimensions as follows:

$$\text{GAP}(F) = \frac{1}{h \times w} \sum_{i=1}^{h} \sum_{j=1}^{w} F(i, j, k) \tag{5}$$

The outputs were concatenated into a unified feature vector as shown:

$$x = concat\ x_1, x_2 \tag{6}$$

3.2.3. Step 3: Classification Layers and Regularization

The combined feature vector passed through a dense ReLU layer, followed by batch normalization and dropout (p = 0.3) as shown :

$$x = ReLU(W_x + b) \tag{7}$$

$$z_{\text{nom}} = \frac{z - \mu}{\sqrt{\sigma^2 + \epsilon}} \tag{8}$$

The final layer used softmax activation to output class probabilities were calculated as follows:

$$\text{softmax}(z_i) = \frac{e^{-1}}{\sum_{j=1}^{C} e_j} \tag{9}$$

3.2.4. Step 4:) Loss Function and Optimization

The model was trained using categorical cross-entropy with label smoothing, where smoothed labels were computed as follows:

$$y_{\sigma\mu oo\tau\eta} = y \times (1 - \alpha) + \frac{\alpha}{C} \tag{10}$$

The smoothed loss function was calculated as:

$$L = -\sum_{i=1}^{C} y_{\sigma\mu oo\tau\eta, i} \log(p_i) \tag{11}$$

Adam optimizer was used for weight updates and was calculated as:

$$w_{t+1} = w_t - \alpha \frac{\widehat{m}_t}{\sqrt{\widehat{v}_t + \epsilon}} \tag{12}$$

Training callbacks included the ModelCheckpoint where model was Saved model if val_loss_new < val_loss_best, **EarlyStopping** was halted the training if no improvement in val_loss and for p epochs and ReduceLROnPlateau was implemented by Adjusting the learning rate as follows:

$$\eta_t = \eta_{t-1} \cdot f\ if\ no\ improvement\ over\ 3\ epochs \tag{13}$$

3.2.5. Step 5: Model Evaluation

Evaluation used standard classification metrics: accuracy, precision, recall, and F1-score as shown:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{14}$$

$$Precision = \frac{TP}{TP+FP} \tag{15}$$

$$Recall = \frac{TP}{TP+FN} \tag{16}$$

$$F1 = 2 \times \frac{Precision\ x\ Recall}{Precision+Recall} \tag{17}$$

Confusion matrix and ROC-AUC were also computed as follows:

$$PR = \frac{TP}{TP+FN}, FPR = \frac{FP}{FP+TN} \tag{18}$$

$$AUC = \int_0^1 TPR(FPR)d(FPR) \tag{19}$$

Predicted and true labels were extracted and calculated as shown:

$$y_{pred_{clas}} = argmax(y_{pred}, axis = 1), y_{true} = argmax(y_{labels}, axis = 1) \tag{20}$$

Table 3 presents the mathematical notations used in the feature extraction process.

Table 3. Notations and Definitions

Notation	Definition
I	Input image represented as a 3D tensor of dimensions 224x 224× 3
H, W	Height and width of input images
f_{mobile} , $f_{\text{efficient}}$	Extracted feature maps from MobileNetV2 and EfficientNetV2
ReLU$(Wx + b)$	Rectified Linear Unit (ReLU) activation function applied to the dense layer
$GAP(x)$	Global Average Pooling applied to feature maps.
W, b	Weights and biases of the dense layer
μ_B, σ_B^2	Batch mean and variance for batch normalization.
m_t, v_t	First and second-moment estimates for Adam optimizer
$\hat{m}_t, \hat{v}_t$	Bias-corrected first and second-moment estimates
concat(·)	Concatenation of feature maps
V_d, b_d	Weights and biases for a dense layer
y_i	Original class label
$\bar{y}_i$	Smoothed label using label smoothing
p_i	Predicted probability for class i after applying softmax
W_s, b_s	Weights and biases for SoftMax classification layer
$P(y)$	Probability distribution over class labels
argmax$(\hat{y})$	Function selecting the class with the highest probability.

3.3. Dataset Description

This study created the DEMF dataset, as shown in Table 4, by integrating the Kaggle dataset (38 classes, 60,343 images) [57] with the FieldPlant dataset with 25,775 images from Central Kenya.

The FieldPlant dataset accounted for seasonal variations, emphasizing fungal and bacterial diseases during April, May, October, and November while prioritizing viral infections in June, July, and December. A standardized collection process ensured diverse lighting conditions and angles, enhancing generalization. Images were classified and annotated by an agricultural expert, and data augmentation techniques were applied to address class imbalances by generating additional samples. The final dataset comprised 22 crop types, 76 individual classes, and 99,551 images, divided into 79,601 training images and 19,950 validation images.

3.4. Dataset Experimental Parameters and Environment

Table 5 outlines the experimental setup, where data was stored and accessed via Google Drive. The dataset was organized for efficient preprocessing and stratified splitting into training and validation sets. Data augmentation and batching were performed to enhance model robustness, with a batch size of 16 ensuring manageable computational loads. Key parameters, such as categorical labels and class names,

were extracted and processed to ensure compatibility with the classification pipeline. The Model architecture combined outputs from MobileNetV2 and EfficientNetV2 through concatenation, followed by dense layers with ReLU activation, batch normalization, and dropout for regularization.

Table 4. Dataset Distribution

Crop Type	Total Images	Training Images	Validation Images
Apple	4,651	3,719	932
Banana	4,008	3,204	804
Beans	8,096	6,475	1,621
Blueberry	1,502	1,201	301
Cassava	4,894	3,914	980
Cherry	2,054	1,642	412
Corn	4,358	3,484	874
Grape	4,641	3,711	930
Maize	1,002	801	201
Maize-L	1,239	991	248
Maize	4,985	3,986	999
Orange	5,507	4,405	1,102
Peach	3,299	2,638	661
Pepper	2,480	1,983	497
Potatoes	3,006	2,403	603
Raspberry	1,002	801	201
Rice	5,010	4,005	1,005
Squash	1,835	1,468	367
Strawberry	2,111	1,688	423
Sugarcane	5,010	4,005	1,005
Sunflower	4,008	3,204	804
Tea	6,012	4,806	1,206
Tomatoes	18,841	15,067	3,774
Total	99,551	79,601	19,950

The Model was compiled with the Adam optimizer and a learning rate of 1×10^{-5}, using categorical cross-entropy loss and label smoothing to improve classification performance. The experiments were conducted on an NVIDIA RTX 3090 GPU, providing reliable and efficient performance. The setup included a virtualized Intel Xeon CPU with access to virtualized GPUs (NVIDIA T4, Tesla P100, K80) and an operating system based on Linux (Ubuntu). Python was the primary language, supported by frameworks such as TensorFlow, PyTorch, Keras, and OpenCV.

Table 5. Hyperparameter Configurations

Hyperparameter	Value
Image size	224 × 224
Image channels	3
Batch size	16
Number of MobileNetV2 layers	Feature extraction only
Number of EfficientNetV2B0 layers	Feature extraction only
Hidden dimension	256
Dropout rate	0.3
Number of epochs	17
Learning rate	1e-5
Optimizer	Adam
Loss function	Categorical Cross entropy (label smoothing = 0.1)
Callbacks	Early Stopping, ModelCheckpoint, ReduceLROnPlateau

4. Results and Discussion

The training and validation curves in Fig. 3 illustrate a steady improvement in accuracy and a consistent decline in loss over 17 epochs. This trend indicates effective learning, strong generalization, and the absence of overfitting. The total training time was 23 hours, 37 minutes, and 24 seconds.

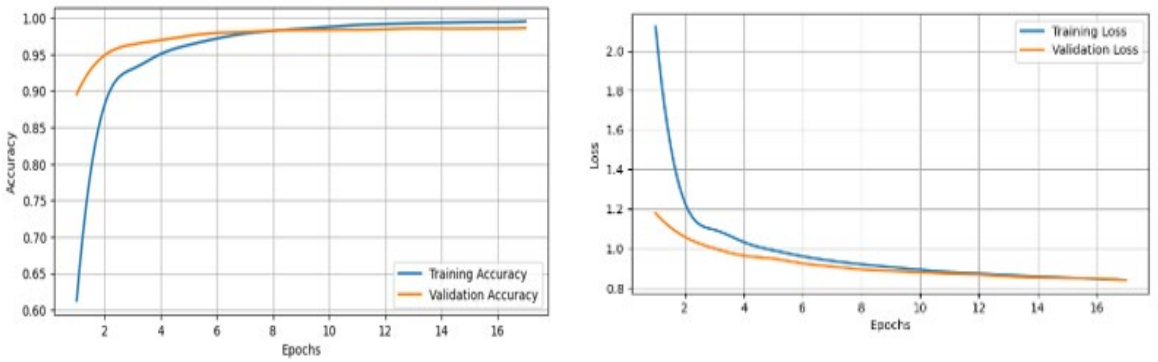


Fig. 3. Training And Validation Graph

The results highlight the model's robustness in real-world environments. Its superiority over standalone EfficientNetV2 and MobileNetV2 lies in the fusion architecture as shown in Fig. 4. EfficientNetV2 captures high-level semantics such as symptom progression and spatial relationships, while MobileNetV2 focuses on low-level textures like leaf lesions and vein patterns. Combined through the fusion layer, these complementary features produce attention-weighted feature maps that balance local and global information. This multi-scale representation enhances the model's capability to generalize across varied agricultural conditions, leading to stable and accurate diagnosis. Moreover, by explicitly integrating both fine-grained edge details and broader contextual cues, the architecture reduces misclassification risks and ensures consistent performance even under challenging field scenarios such as varying lighting, background clutter, and overlapping leaves.

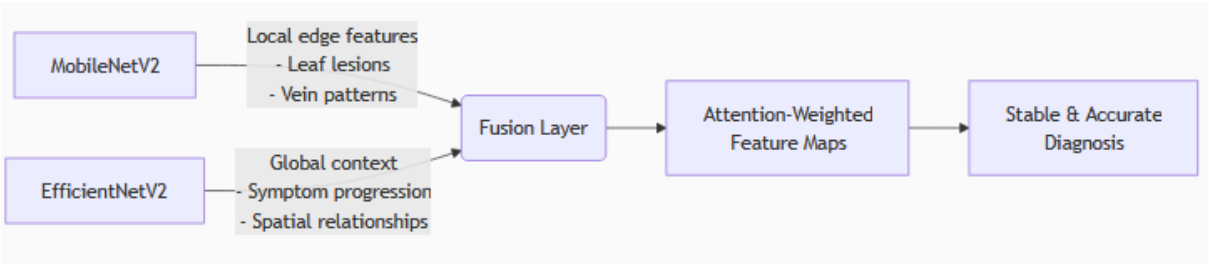


Fig. 4. Training And Validation Graph

The confusion matrices shown in Fig. 5(a)-(f) for each class (ranging from class 0 to class 70) illustrate the performance of the disease detection task and ROC curves in Fig. 6(a)-(b). These matrices display actual class labels on the X-axis and predicted labels on the Y-axis, providing insights into the model's classification accuracy for each class. The confusion matrices for classes 0–10 and 10–20 demonstrate high classification accuracy, particularly for apple, banana, blueberry, cassava, and bean-related diseases, with minimal misclassifications. The confusion matrices for classes 30–40 and 40–50 indicate strong classification accuracy across maize, orange, peach, potato, rice, soybean, and strawberry diseases, with most categories achieving near-perfect predictions. Minor misclassifications, particularly in Rice Brown Spot and Squash Powdery Mildew, suggest that subtle overlaps in visual features, such as similar lesion shapes or coloration patterns, may occasionally lead to errors. Nevertheless, these misclassifications are limited in frequency and do not significantly undermine overall model reliability. The confusion matrices for classes 50–70 further highlight the model's consistent performance across sugarcane, sunflower, tea, tomato, and bean diseases, demonstrating stable recognition with near-perfect accuracy in most categories. Importantly, the balanced results across diverse crop types emphasize the model's robustness in handling both monocot and dicot plant species. This suggests that the fusion-based framework

generalizes well beyond specific plant families and is capable of distinguishing disease symptoms across a wide range of agricultural contexts, making it highly suitable for practical deployment in precision farming and disease management systems.

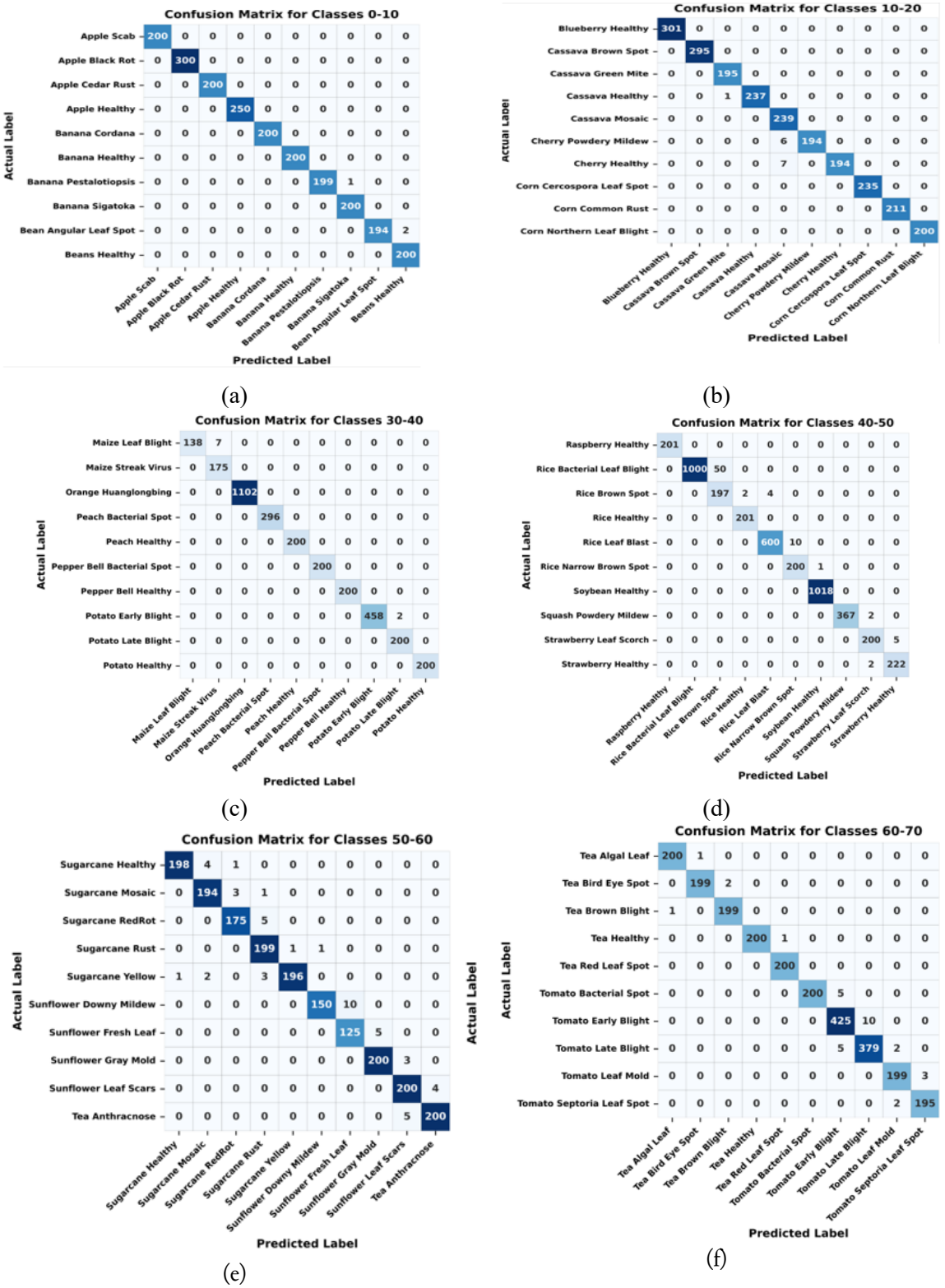
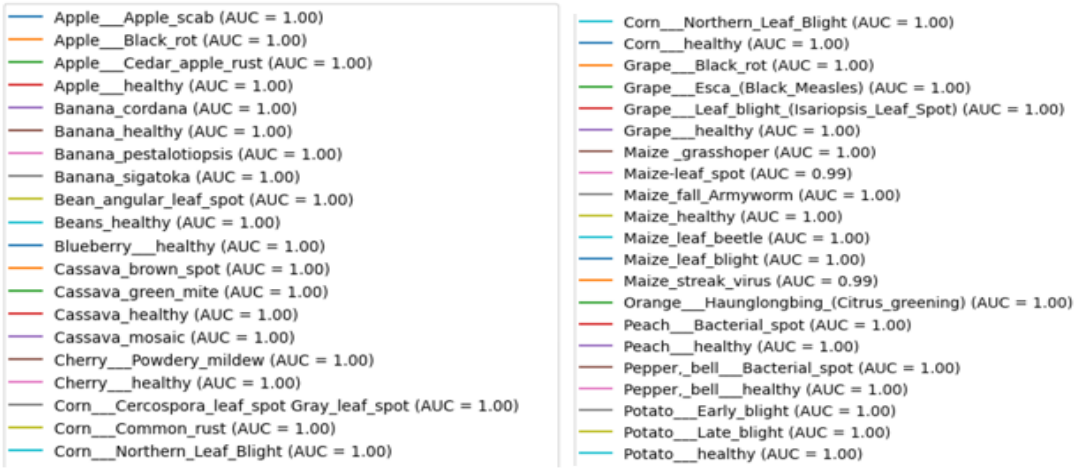
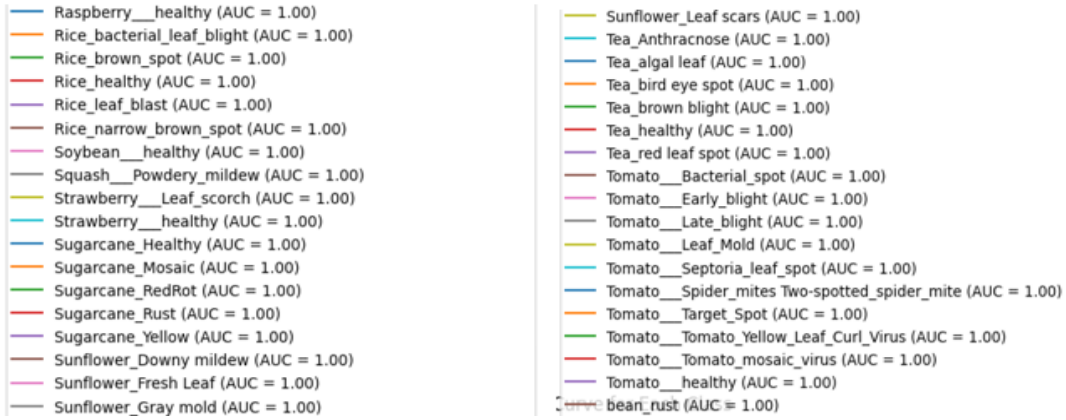


Fig. 5. Confusion Matric for Classes of 0-10 (a), Classes of 10-20 (b), Classes of 30-40, Classes of 40-50, Classes of 50-60 (e) and 60-70 (f)

The model's AUC of 0.9999 indicates near-perfect class separation, ensuring reliable detection even in imbalanced field conditions. A low confidence variance confirms stable predictions under challenges like variable lighting or background clutter. Fig. 6(a) shows classification results with AUC scores for multiple crop diseases and healthy classes. Almost all categories achieve an AUC of 1.00, indicating perfect classification performance, while only a few (e.g., Maize leaf spot and Maize leaf blight) score slightly lower at 0.99. Overall, the model demonstrates highly reliable and consistent accuracy across diverse crops and disease types. Fig. 6 (b) presents AUC scores for disease and healthy classifications across crops such as rice, soybean, squash, strawberry, sugarcane, sunflower, tea, and tomato. Nearly all classes achieve an AUC of 1.00, reflecting perfect classification, with no categories falling below this threshold. This demonstrates the model's exceptional ability to distinguish a wide variety of plant diseases and healthy leaves with consistent reliability.



(a)



(b)

Fig. 6. ROC- AUC Scores

4.1. Ablation Studies

Ablation studies were conducted to assess the impact of individual components, such as the Multi-scale Module, Fine-tuning, and regularization, on model performance. We analyzed their contribution to accuracy and efficiency by systematically removing or adding these elements. This helped optimize the architecture by identifying the most effective configurations for robust crop disease detection. Table 6 presents the ablation experiments conducted to evaluate the impact of different architectural modifications on the performance of the proposed model. Initially, we experimented with a baseline CNN model incorporating EfficientNetV2 and MobileNetV2 with no fine-tuning, which achieved an accuracy of 98.55%. A slight variation of this model resulted in a marginal decrease in accuracy to 98.45%. Next, we integrated a multi-scale module into the CNN architecture, which improved the accuracy to 98.57%. This enhancement demonstrates the effectiveness of multi-scale feature extraction

in capturing more complex patterns in crop disease images. Finally, we introduced data augmentation, fine-tuning, and regularization to refine feature selection further. This combination yielded a final accuracy of 98.63%, indicating that selective feature enhancement, fine-tuning, and regularization strategies contributed to better model generalization. Although these modifications slightly increased the model's complexity, the performance improvements suggest that the proposed approach effectively enhances disease classification accuracy.

Table 6. Impact of Different Architectural Modifications

Model Number	Model Configuration	Multi-scale Module	Gated Mechanism	Accuracy
1	Basic model: EfficientNetV2 and MobileNetV2	No	No	98.55%
2	CNN models: EfficientNetV2 and MobileNetV2 (Epochs Reduction)	No	No	98.45%
3	CNN models: EfficientNetV2 and MobileNetV2, with Multiscale module	Yes	No	98.57%
4	Proposed Model: EfficientNetV2 and MobileNetV2, with Concatenation, Regularization and Fine Tuning	Yes	Yes	98.63%

4.2. Performance of Proposed Model on Unseen Data

The images were randomly stored in one folder. On the unseen data, 249 images were processed, with 239 correctly classified, resulting in an accuracy of 95.98%. Only 10 images (4.02%) were misclassified, further supporting the Model overall strong performance. The Model demonstrated its ability to accurately classify plant diseases, even when the confidence scores were low for a few classes lacking dominant features. This suggested that the proposed Model, as shown in Fig. 7, generally distinguished between healthy and diseased plants.

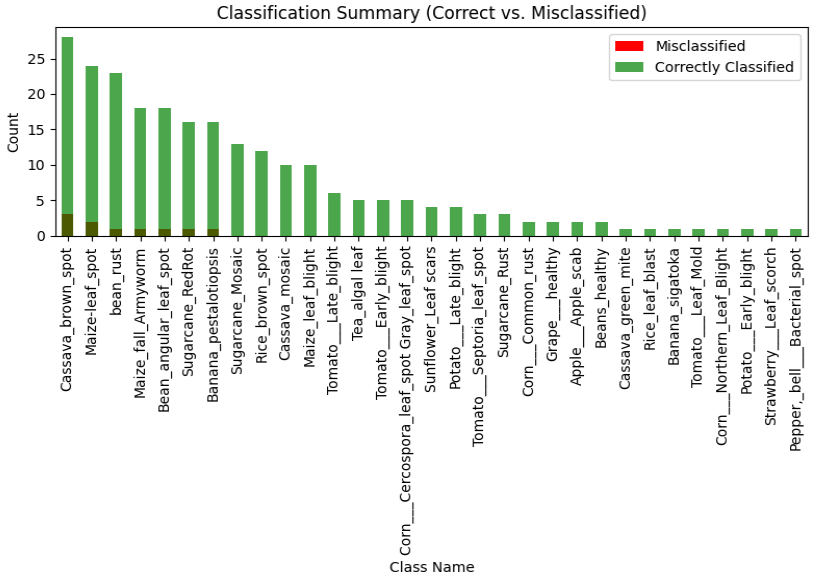


Fig. 7. Classification Summary on Unseen Images

The proposed Model capacity to deliver high-confidence predictions for diseases with strong visual cues, alongside moderate performance for others, highlights its practical utility for real-world deployment. The Model proved its ability to classify plant diseases effectively in real-world scenarios, as demonstrated by the sample results in Fig. 8. When applied to field data, these samples illustrate the Model performance, showcasing its ability to provide accurate predictions across various plant diseases with confidence scores.

Fig. 8. Random Classes Actual Vs. Predicted Classification

4.3. Statistical Testing

Statistical testing confirmed the superiority and stability of the proposed model across all performance metrics. As shown in Table 7, the model achieved the highest Kappa value (0.9919), indicating strong agreement between predictions and actual classifications with minimal misclassification. Its AUC (0.999998) demonstrated near-perfect class distinction, ensuring effective differentiation between healthy and diseased crops.

Table 7. Model Performance

Model	Accuracy	Precision	Recall	F1 Score	Kappa	AUC
Proposed Model	0.992	0.9934	0.9929	0.9923	0.9919	0.999998
Swin_TransformerSE	0.988	0.9901	0.9894	0.9888	0.9878	0.999888
VGG-16	0.972	0.9762	0.9718	0.9712	0.9716	0.999967
ShuffleNet	0.958	0.9676	0.9644	0.9613	0.9574	0.999844
DenseNet121	0.958	0.9676	0.9644	0.9613	0.9574	0.999844
AlexNet	0.948	0.9569	0.9484	0.9452	0.9472	0.999142
DenseNet50	0.896	0.9080	0.8993	0.8922	0.8945	0.998850

A confidence variance analysis (Table 8) further assessed the stability of predictions, where lower variance indicated more consistent predictions. The proposed model achieved the lowest confidence variance (0.000011), highlighting its robustness and reliability, whereas DenseNet50 (0.000035) and AlexNet (0.000027) exhibited higher variance, indicating less stability in classification confidence. The confidence variance visually reinforced these findings, showing that the proposed model maintained the most stable confidence scores, while DenseNet50 and AlexNet displayed greater fluctuations, suggesting lower prediction consistency.

Table 8. Confidence Variance Analysis

Model	Confidence Variance
DenseNet50	0.000035
AlexNet	0.000027
DenseNet121	0.000023
ShuffleNet	0.000023
VGG-16	0.000015
Swin_TransformerSE	0.000013
Proposed Model	0.000011

A Kruskal-Wallis test and pairwise comparisons were conducted to assess statistical differences in confidence variance across models (Table 9). The pairwise comparisons identified significant differences in variance, helping determine which models exhibited more consistent predictions. The Kruskal-Wallis test ($H = 597.40$, $p = 8.4755e-126$) confirmed a highly significant overall difference in confidence scores, making it suitable for analyzing deep learning models with varying confidence distributions. Most

comparisons were significant at 0.05, indicating that models had statistically distinct variances. The proposed model, Swin_TransformerSE, and VGG-16 exhibited significantly lower variance than DenseNet50 and AlexNet, reinforcing their prediction stability. These findings further highlight the robustness and reliability of the proposed model for crop disease detection, as it consistently maintained the lowest confidence variance.

Table 9. Pairwise Comparisons

Model A	Model B	Adj. p-value	Significant (0.05)
DenseNet50	Swin_TranformerSE	2.8595e-89	Yes
Proposed Model	DenseNet50	3.7988e-62	Yes
DenseNet50	VGG-16	2.7891e-59	Yes
AlexNet	Swin_TranformerSE	5.8236e-42	Yes
DenseNet121	Swin_TranformerSE	2.9339e-28	Yes
ShuffleNet	Swin_TranformerSE	2.9339e-28	Yes
AlexNet	DEMF	3.8810e-24	Yes
AlexNet	VGG-16	2.3074e-22	Yes
DenseNet50	ShuffleNet	1.2925e-17	Yes
DenseNet121	DenseNet50	1.2925e-17	Yes
Proposed Model	ShuffleNet	4.6563e-14	Yes
Proposed Model	DenseNet121	4.6563e-14	Yes
ShuffleNet	VGG-16	1.0419e-12	Yes
DenseNet121	VGG-16	1.0419e-12	Yes
AlexNet	DenseNet50	3.4884e-09	Yes
Swin_TranformerSE	VGG-16	3.5459e-03	Yes
Proposed Model	Swin_TranformerSE	1.6019e-02	Yes
AlexNet	DenseNet121	2.6124e-01	No
AlexNet	ShuffleNet	2.6124e-01	No
Proposed Model	VGG-16	1.0000e+00	No
DenseNet121	ShuffleNet	1.0000e+00	No

4.4. Comparison with other Hybrid approaches

The proposed model demonstrated a significant improvement in validation accuracy, rising to 98.63% and outperforming other hybrid approaches, as shown in Table 10. This enhancement is primarily attributed to integrating concatenation, regularization, and fine-tuning, collectively enhancing feature extraction and contextual learning. In contrast, the models presented exhibit comparatively lower classification accuracy due to limitations in their architectural design, likely due to the absence of adaptive feature recalibration mechanisms that optimize feature importance dynamically.

Table 10. Confidence variance analysis

Studies	Classification Accuracy
[34]	98.00%%
[58]	97.50%%
[59]	90.00%
[19]	90.99%
[60]	85.02%
The proposed model	98.63%

4.5. Comparison with Other Models on a Similar Dataset

The proposed hybrid model consistently outperformed existing crop disease detection models, achieving high accuracy rates that surpass traditional CNN-based. As summarized in Table 11, we tested several models, including MobileNetV2, EfficientNetB0, EfficientNetV2, DenseNet121, DenseNet50, ResNet152, AlexNet, and Custom CNN, all on the same dataset for comparison.

Table 11. Comparative Performance of All Models

Model	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss
Proposed Model	99.61%	98.63%	0.0331	0.8458
MobileNetV2	98.92%	98.21%	0.0507	0.8663
EfficientNetB0	97.65%	91.02%	0.0811	1.2424
EfficientNetV2	99.08%	97.95%	0.0702	1.0291
DenseNet121	98.80%	97.75%	0.0675	1.0733
DenseNet50	98.75%	96.11%	0.0706	1.0975
ResNet152	98.74%	96.45%	0.0852	1.2092
AlexNet	97.88%	93.50%	0.1189	1.5391
Custom CNN	92.10%	61.84%	0.2750	2.5675

Table 12 shows that the final trained model sizes varied significantly, with AlexNet being the largest at 551,564 KB due to its complex architecture. ResNet followed with 100,500 KB, while the DEMF Model was slightly smaller at 104,625 KB, which was later optimized to 30 MB for edge deployment. MobileNetV2 was the smallest at 30,908 KB, designed for lightweight operations. EfficientNetB0 and EfficientNetV2 were compact, with sizes around 74,000 KB, offering a balance of performance and efficiency. DenseNet121 was around 97,697 KB, and DenseNet50 was more compact at 42,015 KB, focusing on feature reuse. The Custom CNN was the smallest Model at 8,272 KB, ideal for resource-constrained environments.

Table 12. Model Size Comparisons

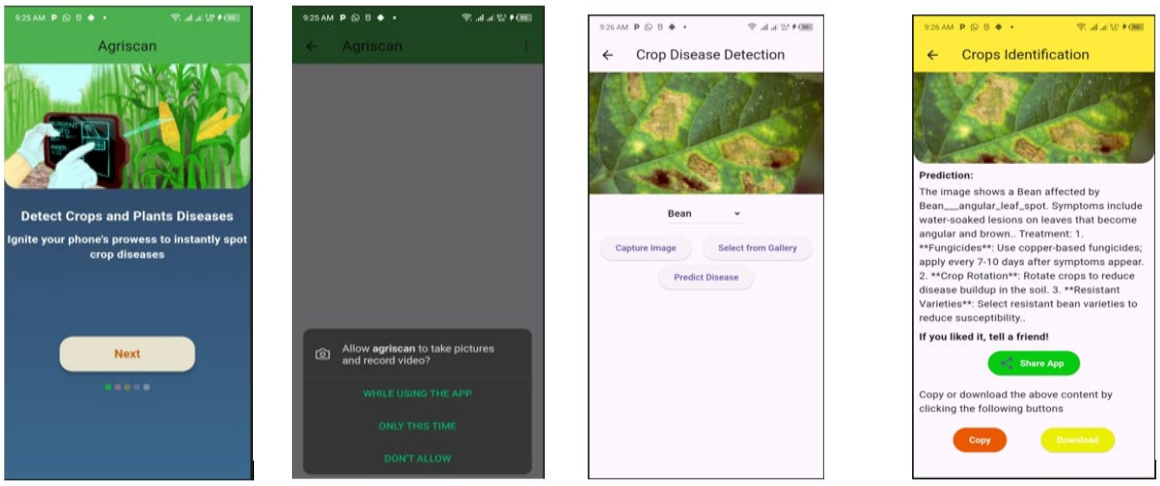
Model	Size (KB)	Size (MB)
Proposed Model	104,625	102.6
MobileNetV2	30,908	30.2
EfficientNetB0	74,256	72.5
EfficientNetV2	73,983	72.3
DenseNet121	97,697	95.5
DenseNet50	42,015	41.1
ResNet	100,500	98.1
AlexNet	551,564	539.5
Custom CNN	8,272	8.1

4.6. Limitations and Deployment Considerations

Despite the model's strong performance, several limitations must be considered for real-world deployment. First, the dataset—though comprehensive with 76 disease classes—may not adequately capture rare or emerging strains, especially under extreme conditions like drought or pest stress, potentially limiting generalization. Second, while the fused architecture provides high accuracy, its 45MB size poses challenges for deployment on ultra-low-cost edge devices (e.g., <$50). Power consumption (~350mW for 2MP inference) is borderline for solar-powered sensors, requiring further optimization. Finally, measures taken to prevent overfitting, such as high dropout (rate=0.5) and batch normalization, may inadvertently suppress rare disease signals or complicate continual learning updates. These issues highlight the need for adaptive fine-tuning, lightweight deployment strategies, and ongoing data expansion to ensure robust and scalable performance in diverse agricultural environments

4.7. A Custom-Built Mobile Application

The app was designed with an intuitive user interface and streamlined workflow to ensure ease of use for farmers seeking real-time crop disease detection. Upon opening the app, users can register in three ways: creating an account with an email and password, signing in through Google, or choosing an anonymous option designed for farmers without email access. After logging in, the app automatically requests users' permission to take pictures, record videos, and record the user's location, as shown in Fig. 9, which was crucial for incorporating user location for further customizations, such as weather data.



(a) Login and Permission Request Screens (b) Diseases Detection Module

Fig. 9. Custom-Built Mobile Appliaction

Once logged in, users are directed to the home page, where they can tap on the crop detection module. The crop diseases module allows users to select the crop they want to test for disease. The app then provides options to capture a real-time image of the crop using the mobile device's camera or to upload an existing image from the gallery. After selecting or capturing an image, users can tap the "Diagnose" button to initiate analysis. The app's model then processes the image, detecting and classifying visible diseases within seconds. The results are instantly displayed, including the disease name, symptoms, and suggested treatment and preventive measures, all tailored to the specific crop, as shown in Fig. 9(b).

5. Conclusion

The proposed Model represents a significant advancement in real-time crop disease detection for resource-constrained settings. By synergizing the architectural strengths of EfficientNetV2, notable for compound scaling and adaptive training efficiency, and MobileNetV2, renowned for depth-wise separable convolutions, the model achieved a training accuracy of 99.52% and validation accuracy of 98.63% within 17 epochs. This rapid convergence underscores the effectiveness of its hybrid design. The model's compact size (30 MB post-quantization) ensures seamless deployment on edge devices, addressing critical latency and storage constraints in low-resource agricultural regions. The key technical innovations include a cross-fusion mechanism, where the concatenation-based feature fusion strategy enabled robust integration of multiscale spatial patterns from both networks, enhancing discriminative power for fine-grained disease identification. Additionally, lightweight optimization techniques, such as ReLU activation, regularization, and quantization, minimized computational overhead while preserving accuracy, making the model viable for real-time inference on devices like Raspberry Pi or low-end drones. Statistical validation across multiple crop categories demonstrated generalizability, with precision-recall metrics and confusion matrices confirming consistent performance, positioning the model as a versatile tool for diverse agricultural systems. Despite its success, the study acknowledges several challenges, including the complexity vs. global context trade-off, where attention mechanisms improved feature refinement but added computational cost, conflicting with edge deployment goals. The current implementation prioritizes efficiency over long-range dependency modeling, potentially limiting performance in cluttered field environments. Dataset constraints also present a challenge, as while the model generalized well to the tested dataset, its scalability to regional disease variants or extreme environmental conditions, such as heavy occlusion and varying lighting, remains unvalidated. Furthermore, hardware-specific bottlenecks persist since, despite the model's 30 MB size being edge-friendly, ultra-low-power microcontrollers, such as the Arduino Nano, may struggle with memory allocation during parallel inference tasks.To address these limitations and expand the model's impact,

future work should focus on integrating lightweight attention alternatives, such as Linformer (linear self-attention) or Performer (kernel-based approximation), to reduce computational costs while retaining global context awareness. Additionally, improving cross-regional adaptability by leveraging transfer learning with region-specific datasets and few-shot learning for rare diseases would enhance practical applicability. Multimodal data integration, combining image data with environmental sensor readings, including humidity and temperature, as well as temporal growth patterns, could improve diagnostic robustness under dynamic field conditions. Further optimization via binary neural networks or hardware-aware neural architecture search (NAS) would extend usability to IoT devices, making the model more accessible. This research establishes a foundational framework for edge-based crop disease detection, balancing accuracy, efficiency, and adaptability. The work paves the way for AI-driven solutions to empower smallholder farmers directly, reduce agricultural losses, and enhance food security in resource-limited regions.

Declarations

Author contribution. All authors contributed to and approved the content of this paper
Funding statement. This research did not receive any specific grant from public, commercial, or not-for-profit funding agencies
Conflict of interest. The authors declare no conflict of interest.
Additional information. No additional information is available for this paper.

References

[1] M. K. Kaleem, N. Purohit, K. Azezew, and S. Asemie D A Assistant, "A Modern Approach for Detection of Leaf Diseases Using Image Processing and ML Based SVM Classifier," *Turkish J. Comput. Math. Educ.*, vol. 12, no. 13, pp. 3340–3347, 2021, [Online]. Available at: https://www.proquest.com/openview/6330532b32b6f07544035f7dd2be60df/1?pq-origsite=gscholar&cbl=2045096.

[2] H. Kim and D. Kim, "Deep-Learning-Based Strawberry Leaf Pest Classification for Sustainable Smart Farms," *Sustainability*, vol. 15, no. 10, p. 7931, May 2023, doi: 10.3390/su15107931.

[3] R. C. Bernardes *et al.*, "Deep-Learning Approach for Fusarium Head Blight Detection in Wheat Seeds Using Low-Cost Imaging Technology," *Agriculture*, vol. 12, no. 11, p. 1801, Oct. 2022, doi: 10.3390/agriculture12111801.

[4] A. M. Abdu, M. M. M. Mokji, and U. U. U. Sheikh, "Machine learning for plant disease detection: an investigative comparison between support vector machine and deep learning," *IAES Int. J. Artif. Intell.*, vol. 9, no. 4, p. 670, Dec. 2020, doi: 10.11591/ijai.v9.i4.pp670-683.

[5] G. G. and A. P. J., "Identification of plant leaf diseases using a nine-layer deep convolutional neural network," *Comput. Electr. Eng.*, vol. 76, pp. 323–338, Jun. 2019, doi: 10.1016/j.compeleceng.2019.04.011.

[6] M. J. Mia, S. K. Maria, S. S. Taki, and A. A. Biswas, "Cucumber disease recognition using machine learning and transfer learning," *Bull. Electr. Eng. Informatics*, vol. 10, no. 6, pp. 3432–3443, Dec. 2021, doi: 10.11591/eei.v10i6.3096.

[7] R. A. F. Ishaq *et al.*, "A Synergistic Framework for Coupling Crop Growth, Radiative Transfer, and Machine Learning to Estimate Wheat Crop Traits in Pakistan," *Remote Sens.*, vol. 16, no. 23, p. 4386, Nov. 2024, doi: 10.3390/rs16234386.

[8] M. Tika Adilah and D. A. Kristiyanti, "Implementation of Transfer Learning Mobilenetv2 Architecture for Identification of Potato Leaf Disease," *J. Theor. Appl. Inf. Technol.*, vol. 101, no. 16, pp. 6273–6285, 2023, [Online]. Available at: https://www.jatit.org/volumes/Vol101No16/3Vol101No16.pdf.

[9] Z. Wu, F. Jiang, and R. Cao, "Research on recognition method of leaf diseases of woody fruit plants based on transfer learning," *Sci. Rep.*, vol. 12, no. 1, p. 15385, Sep. 2022, doi: 10.1038/s41598-022-18337-y.

[10] L. Mohammed and Y. Yusoff, "Detection And Classification Of Plant Leaf Diseases Using Digtal Image Processing Methods: A Review," *ASEAN Eng. J.*, vol. 13, no. 1, pp. 1–9, Feb. 2023, doi: 10.11113/aej.v13.17460.

[11] Z. Zhao, L. Alzubaidi, J. Zhang, Y. Duan, and Y. Gu, "A comparison review of transfer learning and self-supervised learning: Definitions, applications, advantages and limitations," *Expert Syst. Appl.*, vol. 242, p. 122807, May 2024, doi: 10.1016/j.eswa.2023.122807.

[12] Hai Thanh Nguyen, Huong Hoang Luong, Long Bao Huynh, Bao Quoc Hoang Le, Nhan Hieu Doan, and Duc Thien Dao Le, "An Improved MobileNet for Disease Detection on Tomato Leaves," *Adv. Technol. Innov.*, vol. 8, no. 3, pp. 192–209, Jul. 2023, doi: 10.46604/aiti.2023.11568.

[13] M. Kabir, F. Unal, T. C. Akinci, A. A. Martinez-Morales, and S. Ekici, "Revealing GLCM Metric Variations across a Plant Disease Dataset: A Comprehensive Examination and Future Prospects for Enhanced Deep Learning Applications," *Electronics*, vol. 13, no. 12, p. 2299, Jun. 2024, doi: 10.3390/electronics13122299.

[14] S. Barman *et al.*, "Optimized Crop Disease Identification in Bangladesh: A Deep Learning and SVM Hybrid Model for Rice, Potato, and Corn," *J. Imaging*, vol. 10, no. 8, p. 183, Jul. 2024, doi: 10.3390/jimaging10080183.

[15] P. K. Mensah *et al.*, "CCMT: Dataset for crop pest and disease detection," *Data Br.*, vol. 49, p. 109306, Aug. 2023, doi: 10.1016/j.dib.2023.109306.

[16] A. Blekos *et al.*, "A Grape Dataset for Instance Segmentation and Maturity Estimation," *Agronomy*, vol. 13, no. 8, p. 1995, Jul. 2023, doi: 10.3390/agronomy13081995.

[17] H. Amin, A. Darwish, A. E. Hassanien, and M. Soliman, "End-to-End Deep Learning Model for Corn Leaf Disease Classification," *IEEE Access*, vol. 10, pp. 31103–31115, 2022, doi: 10.1109/ACCESS.2022.3159678.

[18] C. Sun, X. Zhou, M. Zhang, and A. Qin, "SE-VisionTransformer: Hybrid Network for Diagnosing Sugarcane Leaf Diseases Based on Attention Mechanism," *Sensors*, vol. 23, no. 20, p. 8529, Oct. 2023, doi: 10.3390/s23208529.

[19] U. Barman *et al.*, "ViT-SmartAgri: Vision Transformer and Smartphone-Based Plant Disease Detection for Smart Agriculture," *Agronomy*, vol. 14, no. 2, p. 327, Feb. 2024, doi: 10.3390/agronomy14020327.

[20] S. Padshetty and Ambika, "Leaky ReLU-ResNet for Plant Leaf Disease Detection: A Deep Learning Approach," in *RAiSE-2023*, Dec. 2023, vol. 59, no. 1, p. 39, doi: 10.3390/engproc2023059039.

[21] T. Wang, H. Xia, J. Xie, J. Li, and J. Liu, "A Multi-Scale Feature Focus and Dynamic Sampling-Based Model for Hemerocallis fulva Leaf Disease Detection," *Agriculture*, vol. 15, no. 3, p. 262, Jan. 2025, doi: 10.3390/agriculture15030262.

[22] Y. Gulzar, "Fruit Image Classification Model Based on MobileNetV2 with Deep Transfer Learning Technique," *Sustainability*, vol. 15, no. 3, p. 1906, Jan. 2023, doi: 10.3390/su15031906.

[23] W. Li, L. Zhu, and J. Liu, "PL-DINO: An Improved Transformer-Based Method for Plant Leaf Disease Detection," *Agriculture*, vol. 14, no. 5, p. 691, Apr. 2024, doi: 10.3390/agriculture14050691.

[24] J. Fu, Y. Zhao, and G. Wu, "Potato Leaf Disease Segmentation Method Based on Improved UNet," *Appl. Sci.*, vol. 13, no. 20, p. 11179, Oct. 2023, doi: 10.3390/app132011179.

[25] M. Xie and N. Ye, "Multi-Scale and Multi-Factor ViT Attention Model for Classification and Detection of Pest and Disease in Agriculture," *Appl. Sci.*, vol. 14, no. 13, p. 5797, Jul. 2024, doi: 10.3390/app14135797.

[26] N. R. Almarines, S. Hashimoto, J. M. Pulhin, C. L. Tiburan, A. T. Magpantay, and O. Saito, "Influence of Image Compositing and Multisource Data Fusion on Multitemporal Land Cover Mapping of Two Philippine Watersheds," *Remote Sens.*, vol. 16, no. 12, p. 2167, Jun. 2024, doi: 10.3390/rs16122167.

[27] S. Yi, S. Hong, Y. Qin, H. Wang, and N. Liu, "Virtual Machine Placement in Edge Computing Based on Multi-Objective Reinforcement Learning," *Electronics*, vol. 14, no. 3, p. 633, Feb. 2025, doi: 10.3390/electronics14030633.

[28] C. Bi, S. Xu, N. Hu, S. Zhang, Z. Zhu, and H. Yu, "Identification Method of Corn Leaf Disease Based on Improved Mobilenetv3 Model," *Agronomy*, vol. 13, no. 2, p. 300, Jan. 2023, doi: 10.3390/agronomy13020300.

[29] X. Chao, G. Sun, H. Zhao, M. Li, and D. He, "Identification of Apple Tree Leaf Diseases Based on Deep Learning Models," *Symmetry (Basel).*, vol. 12, no. 7, p. 1065, Jun. 2020, doi: 10.3390/sym12071065.

[30] I. V. D. Srihith, "A Short Review on Deep Learning in Agriculture," *J. Adv. Res. Artif. Intell. It's Appl.*, vol. 1, no. 3, pp. 23–39, 2024. [Online]. Available at: https://zenodo.org/records/11612440.

[31] M. Gu, K.-C. Li, Z. Li, Q. Han, and W. Fan, "Recognition of Crop Diseases Based on Depthwise Separable Convolution in Edge Computing," *Sensors*, vol. 20, no. 15, p. 4091, Jul. 2020, doi: 10.3390/s20154091.

[32] P. Rajendran, S. Maloo, R. Mitra, A. Chanchal, and R. Aburukba, "Comparison of Cloud-Computing Providers for Deployment of Object-Detection Deep Learning Models," *Appl. Sci.*, vol. 13, no. 23, p. 12577, Nov. 2023, doi: 10.3390/app132312577.

[33] R. Yulvina *et al.*, "Hybrid Vision Transformer and Convolutional Neural Network for Multi-Class and Multi-Label Classification of Tuberculosis Anomalies on Chest X-Ray," *Computers*, vol. 13, no. 12, p. 343, Dec. 2024, doi: 10.3390/computers13120343.

[34] S. Parez, N. Dilshad, N. S. Alghamdi, T. M. Alanazi, and J. W. Lee, "Visual Intelligence in Precision Agriculture: Exploring Plant Disease Detection via Efficient Vision Transformers," *Sensors*, vol. 23, no. 15, p. 6949, Aug. 2023, doi: 10.3390/s23156949.

[35] F. Rajeena P. P., A. S. U., M. A. Moustafa, and M. A. S. Ali, "Detecting Plant Disease in Corn Leaf Using EfficientNet Architecture—An Analytical Approach," *Electronics*, vol. 12, no. 8, p. 1938, Apr. 2023, doi: 10.3390/electronics12081938.

[36] G. Arjun, O. B. A, O. Y. O, A. G. K, B. F. Muhammed, and A. A. M, "Computer Vision-Based Plant Disease Identification System: A Review," *AAN J. Sci. Eng. Technol.*, vol. 1, pp. 59–78, 2022[Online]. Available at : https://d1wqtxts1xzle7.cloudfront.net/95812316/6_Computer_Vision_for_Plant.

[37] Y.-Y. Zheng, J.-L. Kong, X.-B. Jin, X.-Y. Wang, T.-L. Su, and M. Zuo, "CropDeep: The Crop Vision Dataset for Deep-Learning-Based Classification and Detection in Precision Agriculture," *Sensors*, vol. 19, no. 5, p. 1058, Mar. 2019, doi: 10.3390/s19051058.

[38] B. Alabsi, M. Anbar, and S. Rihan, "CNN-CNN: Dual Convolutional Neural Network Approach for Feature Selection and Attack Detection on Internet of Things Networks," *Sensors*, vol. 23, no. 14, p. 6507, Jul. 2023, doi: 10.3390/s23146507.

[39] X. Fang, T. Zhen, and Z. Li, "Lightweight Multiscale CNN Model for Wheat Disease Detection," *Appl. Sci.*, vol. 13, no. 9, p. 5801, May 2023, doi: 10.3390/app13095801.

[40] L. Jia *et al.*, "MobileNet-CA-YOLO: An Improved YOLOv7 Based on the MobileNetV3 and Attention Mechanism for Rice Pests and Diseases Detection," *Agriculture*, vol. 13, no. 7, p. 1285, Jun. 2023, doi: 10.3390/agriculture13071285.

[41] A. Tragoudaras *et al.*, "Design Space Exploration of a Sparse MobileNetV2 Using High-Level Synthesis and Sparse Matrix Techniques on FPGAs," *Sensors*, vol. 22, no. 12, p. 4318, Jun. 2022, doi: 10.3390/s22124318.

[42] S. Zhuang, Y. Hou, and D. Wang, "Towards Efficient Object Detection in Large-Scale UAV Aerial Imagery via Multi-Task Classification," *Drones*, vol. 9, no. 1, p. 29, Jan. 2025, doi: 10.3390/drones9010029.

[43] L. Zhang, G. Ding, C. Li, and D. Li, "DCF-Yolov8: An Improved Algorithm for Aggregating Low-Level Features to Detect Agricultural Pests and Diseases," *Agronomy*, vol. 13, no. 8, p. 2012, Jul. 2023, doi: 10.3390/agronomy13082012.

[44] Z. Mi, X. Zhang, J. Su, D. Han, and B. Su, "Wheat Stripe Rust Grading by Deep Learning With Attention Mechanism and Images From Mobile Devices," *Front. Plant Sci.*, vol. 11, p. 558126, Sep. 2020, doi: 10.3389/fpls.2020.558126.

[45] H. Ge *et al.*, "Cross Attention-Based Multi-Scale Convolutional Fusion Network for Hyperspectral and LiDAR Joint Classification," *Remote Sens.*, vol. 16, no. 21, p. 4073, Oct. 2024, doi: 10.3390/rs16214073.

[46] 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, no. 1, pp. 1–11, Jan. 2016, doi: 10.1155/2016/3289801.

[47] W. Lu, X. Wang, L. Sun, and Y. Zheng, "Spectral–Spatial Feature Extraction for Hyperspectral Image Classification Using Enhanced Transformer with Large-Kernel Attention," *Remote Sens.*, vol. 16, no. 1, p. 67, Dec. 2023, doi: 10.3390/rs16010067.

[48] M. Bakr, S. Abdel-Gaber, M. Nasr, and M. Hazman, "DenseNet Based Model for Plant Diseases Diagnosis," *Eur. J. Electr. Eng. Comput. Sci.*, vol. 6, no. 5, pp. 1–9, Sep. 2022, doi: 10.24018/ejece.2022.6.5.458.

[49] K. Dong, C. Zhou, Y. Ruan, and Y. Li, "MobileNetV2 Model for Image Classification," in *2020 2nd International Conference on Information Technology and Computer Application (ITCA)*, Dec. 2020, pp. 476–480, doi: 10.1109/ITCA52113.2020.00106.

[50] S. Mousavi and G. Farahani, "A Novel Enhanced VGG16 Model to Tackle Grapevine Leaves Diseases With Automatic Method," *IEEE Access*, vol. 10, pp. 111564–111578, 2022, doi: 10.1109/ACCESS.2022.3215639.

[51] Q. Dai *et al.*, "Citrus Disease Image Generation and Classification Based on Improved FastGAN and EfficientNet-B5," *Agronomy*, vol. 13, no. 4, p. 988, Mar. 2023, doi: 10.3390/agronomy13040988.

[52] H. Ulutaş and V. Aslantaş, "Design of Efficient Methods for the Detection of Tomato Leaf Disease Utilizing Proposed Ensemble CNN Model," *Electronics*, vol. 12, no. 4, p. 827, Feb. 2023, doi: 10.3390/electronics12040827.

[53] M. E. H. Chowdhury *et al.*, "Automatic and Reliable Leaf Disease Detection Using Deep Learning Techniques," *AgriEngineering*, vol. 3, no. 2, pp. 294–312, May 2021, doi: 10.3390/agriengineering3020020.

[54] T. Paranayapa, P. Ranasinghe, D. Ranmal, D. Meedeniya, and C. Perera, "A Comparative Study of Preprocessing and Model Compression Techniques in Deep Learning for Forest Sound Classification," *Sensors*, vol. 24, no. 4, p. 1149, Feb. 2024, doi: 10.3390/s24041149.

[55] G. R. Babu, M. Gokuldhev, and P. S. Brahmanandam, "Integrating IoT for Soil Monitoring and Hybrid Machine Learning in Predicting Tomato Crop Disease in a Typical South India Station," *Sensors*, vol. 24, no. 19, p. 6177, Sep. 2024, doi: 10.3390/s24196177.

[56] A. Siddiqua, M. A. Kabir, T. Ferdous, I. B. Ali, and L. A. Weston, "Evaluating Plant Disease Detection Mobile Applications: Quality and Limitations," *Agronomy*, vol. 12, no. 8, p. 1869, Aug. 2022, doi: 10.3390/agronomy12081869.

[57] M. H. Saleem, J. Potgieter, and K. M. Arif, "Plant Disease Classification: A Comparative Evaluation of Convolutional Neural Networks and Deep Learning Optimizers," *Plants*, vol. 9, no. 10, p. 1319, Oct. 2020, doi: 10.3390/plants9101319.

[58] D. Zhu, J. Tan, C. Wu, K. Yung, and A. W. H. Ip, "Crop Disease Identification by Fusing Multiscale Convolution and Vision Transformer," *Sensors*, vol. 23, no. 13, p. 6015, Jun. 2023, doi: 10.3390/s23136015.

[59] S. A. Shah, I. Taj, S. M. Usman, S. N. Hassan Shah, A. S. Imran, and S. Khalid, "A hybrid approach of vision transformers and CNNs for detection of ulcerative colitis," *Sci. Rep.*, vol. 14, no. 1, p. 24771, Oct. 2024, doi: 10.1038/s41598-024-75901-4.

[60] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jegou, "Training data-efficient image transformers & distillation through attention," in *Proceedings of the 38th International Conference on Machine Learning*, Jul. 2021, pp. 10347–10357. [Online]. Available at: https://proceedings.mlr.press/v139/touvron21a.html.