

Article

Leakage-Aware Transfer Learning with Explainable AI and CPU-Efficient Deployment for Mango Leaf Disease Classification on the MangoLeafBD Benchmark

Wirapong Chansanam ¹, Avshalom Elmalech ², Suparp Kanyacome ³, Prasert Luekhong ⁴, Natthakan Iam-On ⁵ and Tossapon Boongoen ^{5,*}

¹ Department of Information Science, Faculty of Humanities and Social Sciences, Khon Kaen University, Khon Kaen 40002, Thailand; wirach@kku.ac.th

² Department of Information Science and Applied AI, Bar-Ilan University, Ramat Gan 5290002, Israel; avshalom.elmalech@biu.ac.il

³ Faculty of Science and Engineering, Kasetsart University, Sakon Nakhon 47000, Thailand; suparp.ka@ku.th

⁴ Cybersecurity Research Center, National Cyber Security Agency (NCSA), Bangkok 10210, Thailand; prasert.l@ncsa.or.th

⁵ Advanced Reasoning Research Group, Department of Computer Science, Aberystwyth University, Ceredigion SY23 3DB, UK; nai7@aber.ac.uk

* Correspondence: tob45@aber.ac.uk

Featured Application

This work delivers a deployable, leakage-aware mango leaf disease diagnostic system targeted at smallholder growers, extension officers, and agricultural advisory services in mango-producing regions. The compact EfficientNetB0 classifier runs in real time on CPU-only hardware (~25 ms per image) and is operationalized as a public Hugging Face Spaces web application with no local installation required. Users upload a single leaf image and receive class predictions, confidence scores, Grad-CAM visual explanations, and Thai-language disease-management guidance, enabling rapid in-field triage, timely intervention against eight major foliar conditions, and scalable AI-assisted decision support for precision plant protection.

Abstract

Background and Aim: Mango (*Mangifera indica*) is a globally important fruit crop whose productivity is repeatedly threatened by foliar diseases such as anthracnose, bacterial canker, powdery mildew, die-back, and sooty mould. Although deep learning has rapidly advanced automated leaf-disease diagnosis, recently reported accuracies on the MangoLeafBD benchmark are approaching saturation, and many studies still rely on random data splits that may inflate performance through leakage of duplicate or near-duplicate images. This study aimed to develop and rigorously evaluate a leakage-aware, deployment-oriented deep learning framework for the complete eight-class MangoLeafBD task. **Methods:** Three modern transfer-learning backbones—EfficientNetB0, MobileNetV3Large, and ConvNeXtTiny—were fine-tuned using a two-stage training strategy on a group-aware data partition. Duplicate and near-duplicate images were detected with cleaned filename stems, average hashing (aHash), and difference hashing (dHash) and grouped before splitting to guarantee zero cross-partition overlap. The strongest models were combined through probability-level soft voting. Robustness was further assessed using 5-fold StratifiedGroupKFold cross-validation. Explainability was examined with Gradient-weighted Class Activation Mapping (Grad-CAM), deployment suitability was characterized through CPU latency benchmarking, and the framework was operationalized as a publicly accessible web-based diagnostic system. **Results:** EfficientNetB0 achieved the highest performance



Academic Editor: Mara Pistellato

Received: 27 June 2026

Revised: 8 July 2026

Accepted: 10 July 2026

Published: 12 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

under the leakage-controlled protocol, reaching 99.50% accuracy and 99.50% weighted F1-score on the 599-image group-aware test set. The heterogeneous soft-voting ensemble matched EfficientNetB0 but did not exceed it, indicating that the ensemble gains reported in earlier studies may partly reflect optimistic split conditions. Five-fold grouped cross-validation confirmed stability, yielding a mean accuracy of 99.63% (SD = 0.20%) and a mean weighted F1 of 99.62% (SD = 0.20%). Grad-CAM visualizations showed that the model attended to biologically meaningful lesion regions across all eight classes, and CPU benchmarking produced a mean single-image latency of 25.3 ms (p95 = 27.6 ms) with batch throughput scaling up to 166.5 images per second. Conclusions: The proposed framework demonstrates that a compact, leakage-aware EfficientNetB0 model can match the accuracy of more complex hybrid CNN–transformer architectures while remaining interpretable and deployable on commodity CPU hardware. By coupling group-aware evaluation, explainable AI, latency benchmarking, and a publicly accessible web application, the study advances reproducible, deployment-ready precision agriculture research and offers a more rigorous benchmarking protocol for future mango leaf disease classification studies.

Keywords: mango leaf disease; MangoLeafBD; transfer learning; EfficientNetB0; data leakage; group-aware evaluation; explainable AI; Grad-CAM; precision agriculture; deep learning deployment

1. Introduction

Mango (*Mangifera indica*) is among the most economically valuable fruit crops worldwide, yet foliar diseases continue to cause substantial reductions in yield, fruit quality, and farm profitability [1]. According to recent agricultural reports, diseases such as anthracnose, bacterial canker, powdery mildew, dieback, and sooty mould remain major threats to sustainable mango production, particularly in tropical and subtropical regions where environmental conditions favor disease proliferation [2,3]. The consequences of delayed disease detection extend beyond crop losses, affecting food security, production costs, and the livelihoods of millions of farmers. Despite the importance of timely diagnosis, disease identification in many agricultural settings still relies on manual visual inspection, which is subjective, time-consuming, and susceptible to human error, especially when symptoms are subtle or emerge during early infection stages [4,5]. These challenges have intensified the demand for accurate, automated, and scalable disease diagnosis systems based on artificial intelligence (AI).

Recent advances in deep learning have substantially improved the automated identification of plant diseases by enabling models to learn complex visual patterns directly from leaf images. Much of this progress has been driven by convolutional neural network (CNN) architectures, which have demonstrated strong performance in mango leaf disease classification. For instance, studies employing AlexNet, ResNet, EfficientNet, and other customized CNN models have reported high levels of classification accuracy, highlighting the effectiveness of deep feature learning for disease recognition [3,6]. Building on these developments, Vijay et al. [7] proposed DeepLeafNet, a framework that combines U-Net-based segmentation, NAS-Net classification, and genetic algorithm-driven hyperparameter optimization, achieving 98.56% classification accuracy across nine mango disease categories.

In parallel, ensemble learning has gained increasing attention as a means of improving predictive performance by integrating the strengths of multiple classifiers. Gautam et al. [8], for example, combined several deep learning and machine learning models and achieved

an accuracy of 98.57%, demonstrating the value of complementary feature representations for disease discrimination. Similarly, Thalor et al. [9] introduced a homogeneous bagging framework based on VGG-19 and reported 95% classification accuracy, with precision, recall, and F1-scores reaching 97%. Although these studies demonstrate the potential of ensemble-based approaches, many CNN-driven methods still struggle to capture broader contextual information and complex spatial relationships among disease symptoms. This limitation is particularly relevant in mango leaf disease classification, where visual symptoms often exhibit substantial intra-class variation and high inter-class similarity, making accurate discrimination difficult even for advanced deep learning models [10,11].

To address these limitations, a second stream of research has explored transformer-based and hybrid architectures that combine local texture analysis with global contextual understanding. Vision Transformers (ViTs) have demonstrated superior capability in modeling long-range dependencies within images, making them particularly suitable for distinguishing visually similar disease symptoms. Ortiz et al. [4] integrated Vision Transformers with Gray-Level Co-occurrence Matrix (GLCM) features and achieved substantial improvements in precision, recall, and F1-score compared with conventional approaches. Similarly, Jashim et al. [12] proposed ViX-MangoEFormer, which combines convolutional operations, self-attention mechanisms, and GLCM-based statistical features, producing an F1-score of 99.78% while maintaining strong cross-domain generalization performance. More recently, Ergün [13] introduced MangoLeafCMDf-FAMNet, a hybrid architecture that integrates ConvNeXt and Vision Transformer backbones through feature attention and cross-modal fusion mechanisms, achieving classification accuracies exceeding 99% across benchmark datasets. Collectively, these studies indicate that hybrid CNN-transformer frameworks can outperform single-backbone models by leveraging complementary feature representations.

Beyond predictive performance, model efficiency and explainability have become increasingly important considerations for practical agricultural deployment. Although highly complex architectures often achieve exceptional classification results, their computational requirements may restrict implementation in resource-constrained environments. Addressing this challenge, Rahman et al. [14] developed MangoLeafNet-XAI, a lightweight architecture based on DenseNet-121 and Efficient Channel Attention modules that achieved state-of-the-art performance using only 6.9 million parameters. Furthermore, explainable AI techniques such as Grad-CAM and LIME were incorporated to improve model transparency and facilitate user trust. Similarly, Jashim et al. [12] integrated Grad-CAM visualization to provide interpretable evidence supporting classification decisions. These developments demonstrate growing recognition that practical disease diagnosis systems must balance accuracy, computational efficiency, and interpretability.

Despite these advances, several critical gaps remain in the literature. First, many studies emphasize maximizing classification accuracy while paying insufficient attention to the integration of local and global feature representations, which is essential for distinguishing diseases with similar visual characteristics under varying environmental conditions [4,12]. Second, although hybrid architectures have shown promising results, many remain computationally intensive and difficult to deploy in real-world agricultural settings [14]. Third, benchmarking practices vary considerably across studies because different datasets, class distributions, preprocessing procedures, and evaluation protocols are often used, making direct comparison difficult. This issue is particularly evident in mango disease research, where some studies evaluate only partial disease categories rather than the complete benchmark datasets available. For example, Ahmed et al. [15] introduced the MangoLeafBD dataset containing 4000 images equally distributed across eight classes: Anthracnose, Bacterial Canker, Cutting Weevil, Die Back, Gall Midge, Healthy, Powdery Mildew, and Sooty Mould. However, subsequent studies have frequently used modified or reduced subsets,

limiting the comparability and generalizability of findings. Finally, although explainability has received increased attention, many state-of-the-art models continue to function as black-box systems, creating barriers to adoption among agricultural practitioners [14].

A methodological concern that has received comparatively little attention in the MangoLeafBD literature is the choice between random splitting and group-aware splitting. Random splitting assumes that every image in the dataset is an independent draw from the underlying distribution of mango-leaf appearances. This assumption is convenient but is inappropriate for MangoLeafBD in particular, because the dataset was expanded from a comparatively small pool of source leaves through operations such as re-photography, cropping, rotation, and light augmentation. When these visually near-duplicate images are distributed at random, an image derived from the same physical leaf can appear in both the training partition and the test partition simultaneously. Under such conditions, the classifier is not required to generalise to unseen leaves; it needs only to memorise near-duplicate variants of leaves it has already seen. Group-aware splitting, in contrast, first groups images that are perceptually near-duplicate (using cleaned filename stems and perceptual hashes such as aHash and dHash) and then places every image in a group into the same partition. This forces the classifier to be evaluated on genuinely unseen leaves and therefore yields a more honest estimate of generalisation performance. The present study adopts group-aware splitting as its primary evaluation protocol, and treats the difference between the two protocols as a substantive methodological contribution rather than as an implementation detail.

This study explores a group learning framework for mango leaf disease classification that combines modern transfer learning techniques with heterogeneous ensemble learning. Specifically, the study aims to establish robust single-model baselines using contemporary deep learning backbones on the complete eight-class MangoLeafBD dataset [15], evaluate whether heterogeneous ensemble learning can outperform the best individual models, and translate the most effective model into a deployable web-based diagnostic system suitable for both Hugging Face Spaces and Streamlit environments. By emphasizing full-dataset benchmarking, architectural diversity, and deployment readiness, this study seeks to provide a more rigorous evaluation framework than many existing approaches.

The significance of this study is both theoretical and practical. Theoretically, it contributes to the literature on transfer learning, heterogeneous ensemble learning, and explainable AI for plant disease diagnosis by showing that aggressive ensemble gains reported under random splits may not survive once duplicate-aware evaluation is enforced. Practically, the work delivers a deployable, CPU-efficient, and interpretable diagnostic platform that can support early disease detection, reduce crop losses, and assist agricultural extension services. By integrating leakage control, grouped cross-validation, explainability, latency benchmarking, and a publicly accessible web application within a single study, this research advances the development of scalable, reproducible, and trustworthy AI-driven plant pathology systems.

The main contributions of this study can be summarized as follows. (i) It proposes a leakage-aware evaluation protocol for the complete eight-class MangoLeafBD benchmark, in which duplicate and near-duplicate images are detected through cleaned filename stems, aHash, and dHash, and confined to a single partition prior to training, validation, and testing. This protocol is contrasted throughout the paper with conventional random splitting, whose assumptions do not hold for MangoLeafBD (see also Sections 1 and 3.2). (ii) It benchmarks three modern transfer-learning backbones—EfficientNetB0, MobileNetV3Large, and ConvNeXtTiny—under this stricter protocol and quantifies their relative robustness through 5-fold StratifiedGroupKFold cross-validation. (iii) It investigates whether probability-level soft voting between heterogeneous backbones provides genuine gains

once duplicate leakage has been removed, and reports a negative result that updates the optimistic conclusions previously drawn under random splits. (iv) It supplies Grad-CAM-based explainability evidence across all eight classes, demonstrating that the selected model attends to biologically meaningful lesion regions rather than irrelevant background structure. (v) It characterizes CPU inference latency and batch throughput on commodity hardware and operationalizes the best model as a publicly accessible web-based diagnostic system with Thai-language disease information and input-quality validation, providing a deployment-ready reference implementation for downstream precision-agriculture research.

The remainder of this paper is organized as follows. Section 2 reviews the related literature on mango leaf disease classification, modern transfer-learning architectures, and explainable AI in agricultural computer vision. Section 3 describes the dataset, the leakage-aware data partitioning strategy, the model backbones, the two-stage training configuration, the heterogeneous ensemble framework, the Grad-CAM and CPU latency protocols, and the web application design. Section 4 reports the single-model, ensemble, cross-validation, explainability, latency, and system-level results. Section 5 discusses the implications for benchmarking practice and field deployment, and Section 6 concludes with limitations and directions for future work.

2. Literature Review

Public datasets are a prerequisite for reproducible agricultural AI research. Ahmed et al. [15] introduced MangoLeafBD as a curated benchmark containing 4000 RGB images derived from roughly 1800 distinct mango leaves collected in Bangladesh. The dataset spans seven disease categories plus a healthy class and remains one of the most useful public benchmarks for mango leaf disease classification. Because the images were collected under orchard conditions and expanded through zooming and rotation, the dataset provides a practical setting for transfer learning research, although its geographic origin still limits direct claims about universal generalization.

Early studies on mango leaf disease diagnosis primarily relied on traditional machine learning techniques and conventional convolutional neural network architectures. Although these approaches demonstrated the feasibility of automated disease recognition, their effectiveness was often constrained by limited datasets, handcrafted feature extraction procedures, and relatively modest predictive performance. With the increasing availability of large-scale pretrained models, researchers began to adopt transfer learning strategies using architectures such as VGG, MobileNet, ResNet, Inception, and EfficientNet, resulting in substantial improvements in classification accuracy and model robustness.

Several recent studies have demonstrated the effectiveness of these modern architectures in mango disease classification. For example, Puranik et al. [16] fine-tuned MobileNetV3 using the MangoLeafBD dataset and integrated the trained model into a mobile application, achieving an accuracy of 98.00% while supporting real-time disease diagnosis in field environments. Similarly, Singh et al. [17] proposed the DTLTD framework based on transfer learning and reported a peak classification accuracy of 99.76% on a dataset comprising 4000 mango leaf images. In a comparative evaluation of deep learning architectures, Bairwa et al. [18] found that ResNet50 achieved the highest performance, reaching an accuracy of 99.12%. Likewise, Pratap and Kumar [19] developed a transfer learning-based classification system incorporating VGG-16, GoogLeNet, MobileNet, YOLOv8, and EfficientNet, demonstrating effective identification of major mango diseases, including Anthracnose, Gall Midge, and Powdery Mildew. More recently, Pahati et al. [20] employed Google Teachable Machine to train a disease recognition model using 4000 annotated leaf images and achieved an accuracy of 99.60%, highlighting the growing potential of acces-

sible and user-friendly artificial intelligence platforms for agricultural disease diagnosis. Collectively, these studies indicate that transfer learning has become a dominant paradigm in mango leaf disease classification, offering both high predictive performance and practical deployment opportunities across a range of agricultural applications.

Thalor et al. [9] proposed a homogeneous bagging framework based on VGG-19 and reported 95% accuracy and 97% precision, recall, and F1. That work is a useful comparison point because it explicitly framed mango disease classification as an ensemble problem. However, the use of repeated VGG-19 learners restricts architectural diversity, and the reported setup appears weaker than the full eight-class MangoLeafBD specification described by Ahmed et al. [15]. This leaves room for a stronger and more transparent group learning design.

Recent studies suggest that high-performing mango disease classification benefits from modern backbone design and heterogeneous feature extraction. Alamri et al. [21] reported 99.87% accuracy on MangoLeafBD using a fused Vision Transformer and ConvNeXt architecture, demonstrating the value of combining convolutional and transformer representations. Ergün [13] proposed an attention-enhanced hybrid model using ConvNeXt and a vision transformer with cross-modal fusion and reported accuracies above 99% on multiple mango datasets. These findings show that hybrid architectures can successfully capture both local lesion structure and global context.

Transformer-oriented approaches also continue to gain relevance. Hossain et al. [22] showed that a Data-efficient Image Transformer (DeiT)-based framework achieved 99.75% accuracy in mango leaf disease identification while outperforming several CNN baselines. The result reinforces the idea that global attention can be useful for leaf disease patterns spread over wide regions of the lamina.

Considered together, the reviewed studies indicate that recent MangoLeafBD-based work is approaching ceiling performance under controlled experimental conditions. As a result, marginal accuracy improvements alone may no longer be sufficient to justify in the research field. Stronger contributions are more likely to come from rigorous evaluation protocols, transparent treatment of duplicate or near-duplicate images, architectural diversity, explainability, deployment characterization, and reproducible artifacts. The present study aligns with this direction by combining leakage-aware benchmarking, heterogeneous backbone comparison, Grad-CAM explainability, CPU latency profiling, and an operational web-based diagnostic system within a single, internally consistent framework.

Three specific research gaps motivate the present work. First, duplicate-aware or perceptual-hash-based partitioning is rarely reported for MangoLeafBD, even though the dataset was derived from a comparatively small number of source leaves and is therefore vulnerable to optimistic estimates when using random splits. Second, many high-performing hybrid CNN–transformer studies emphasize headline accuracy without simultaneously reporting evidence of explainability, CPU latency, and a usable end-user interface, thereby limiting their practical relevance for resource-constrained agricultural settings. Third, the ensemble-learning literature on mango disease classification rarely tests whether reported gains hold up under a stricter, duplicate-controlled evaluation. The present study addresses all three gaps simultaneously.

As shown in Table 1, recent deep learning studies on mango leaf disease classification have achieved consistently high accuracy, yet several methodological limitations persist throughout the literature. As summarized in Table 1, the majority of reviewed studies employed transfer-learning backbones or ensemble architectures—including VGG-19 bagging [9], ResNet50 [18], and Vision Transformer variants [22]—reporting accuracies ranging from 95.0% to above 99%, predominantly on the MangoLeafBD dataset or its derivatives. Despite these strong performance figures, none of the studies explicitly addressed data

leakage control, raising concerns about the validity of the reported metrics given MangoLeafBD's known near-duplicate image structure. Explainability was similarly limited; only Jashim et al. [12] and Rahman et al. [14] incorporated Grad-CAM visualizations, and neither systematically extended it across all eight disease classes. Deployment remained largely absent from the literature, with only Puranik et al. [16] providing a mobile application and Pahati et al. [20] offering a web demonstration, though neither disclosed CPU latency nor ensured public accessibility. In contrast, the present study addresses these gaps by combining an EfficientNetB0-based soft-voting ensemble with a leakage-aware split (aHash, dHash, and filename-stem deduplication), full eight-class Grad-CAM explainability, and a publicly accessible CPU-optimized web application hosted on Hugging Face Spaces, achieving 99.50% single-model accuracy and 99.63% cross-validated accuracy on the cleaned test set.

Table 1. Summary of recent deep-learning studies on mango leaf disease classification.

Study	Method/Backbone	Dataset (Classes)	Accuracy (%)	Leakage Control	XAI	Deployment
Thalor et al. [9]	VGG-19 homogeneous bagging	MangoLeafBD subset (claimed 8)	95.0	No	No	No
Singh et al. [17]	DTLD transfer-learning ensemble	MangoLeafBD-derived (8)	99.76	Not reported	No	No
Bairwa et al. [18]	ResNet50 transfer learning	Mango leaf images (7)	99.12	Not reported	No	No
Pratap & Kumar [19]	VGG16 + MobileNet + EfficientNet	Mango leaf images	98–99	Not reported	No	No
Puranik et al. [16]	MobileNetV3 + mobile app	MangoLeafBD (8)	98.00	Not reported	No	Yes (mobile only)
Gautam et al. [8]	ESDNN stacked ensemble	MangoLeafBD (8)	98.57	Not reported	No	No
Pahati et al. [20]	Google Teachable Machine	MangoLeafBD-derived (8)	99.60	Not reported	No	Yes (web demo)
Hossain et al. [22]	DeiT (Vision Transformer)	MangoLeafBD (8)	99.50–99.75	Not reported	No	No
Jashim et al. [12]	ViX-MangoEFormer + GLCM	MangoLeafBD + custom	99.78 (F1)	Not reported	Yes (Grad-CAM)	No
Ortiz et al. [4]	ViT + GLCM	Mango leaf images	≈99	Not reported	Partial	No
Ergün [13]	MangoLeafCMDf-FAMNet (ConvNeXt + ViT)	MangoLeafBD (8)	>99	Not reported	No	No
Rahman et al. [14]	MangoLeafNet-XAI (DenseNet121 + ECA)	MangoLeafBD (8)	SOTA	Not reported	Yes (Grad-CAM, LIME)	No
Vijay et al. [7]	DeepLeafNet (U-Net + NAS-Net + GA)	Mango leaves (9)	98.56	Not reported	No	No
This study (2026)	EfficientNetB0 + soft voting + leakage-aware split	MangoLeafBD (full 8)	99.50 (single)/ 99.63 (CV)	Yes (aHash + dHash + stem)	Yes (Grad-CAM all 8 classes)	Yes (CPU app, public HF Space)

3. Materials and Methods

3.1. Dataset

This study used the MangoLeafBD dataset [15], which contains 4000 RGB images with nominal dimensions centered around 240×320 pixels. The dataset is balanced across eight classes with 500 images per class: Anthracnose, Bacterial Canker, Cutting Weevil, Die

Back, Gall Midge, Healthy, Powdery Mildew, and Sooty Mould. The local project directory was verified to contain all eight categories with complete class balance.

3.2. Data Splitting Strategy

An initial stratified hold-out split was first used for exploratory screening. However, a post hoc audit of filenames and perceptual image hashes showed that duplicate or near-duplicate images could appear across training, validation, and test partitions, which risked inflating performance. To address this issue, the primary evaluation reported in this manuscript uses a stricter group-aware hold-out split.

Within each class, images were grouped using three signals: a cleaned filename stem, exact average hash (aHash), and exact difference hash (dHash). Connected images were assigned to the same group before splitting so that grouped duplicates could not cross partitions. The resulting group counts differed substantially across classes and quantified the severity of the near-duplicate problem in MangoLeafBD. Anthracnose contained 483 groups (17 groups of size > 1, maximum group size = 2), Bacterial Canker 495 groups (5 groups of size > 1, max = 2), Cutting Weevil 241 groups (all 241 groups had size > 1, with the maximum group containing 6 images), Die Back 489 groups (11 groups of size > 1, max = 2), Gall Midge 491 groups (8 groups of size > 1, max = 3), Healthy 495 groups (5 groups of size > 1, max = 2), Powdery Mildew 496 groups (3 groups of size > 1, max = 3), and Sooty Mould 498 groups (2 groups of size > 1, max = 2). The Cutting Weevil class is by far the most affected: of its 500 images, only 241 unique groups exist, indicating a very high proportion of visually near-duplicate images that would have systematically inflated the estimated accuracy under a naive random split. The final group-aware split preserved the approximate 70%/15%/15% ratio, yielding 2800 training images, 601 validation images, and 599 test images. Overlap audits confirmed zero cross-split overlap for group_id, cleaned stem, aHash, and dHash.

To test stability beyond a single hold-out partition, the strongest model was further evaluated using 5-fold grouped cross-validation over the full grouped dataset. Stratified-GroupKFold was used so that label balance was approximately preserved while grouped duplicates remained confined to a single fold. Each fold used 3200 training images and 800 validation images.

3.3. Model Backbones

Three transfer learning backbones were selected to support both architectural diversity and deployment relevance. EfficientNetB0 was included as a strong and compact pretrained CNN baseline. MobileNetV3Large was chosen because of its lightweight mobile-oriented design and practical deployment advantages. ConvNeXtTiny was selected as a more recent convolutional architecture reflecting transformer-era design principles [23].

3.4. Training Configuration

All models used pretrained ImageNet initialization and accepted resized 224×224 RGB input. The training pipeline applied random horizontal and vertical flips, random rotations ($\pm 20^\circ$), random zooms ($\pm 10\%$), random translations ($\pm 10\%$), and random contrast adjustments ($\pm 20\%$) to reduce overfitting and improve robustness to image-capture variability. A classification head consisting of global average pooling, batch normalization, a 0.3 dropout layer, and an eight-unit softmax output layer was attached to each pretrained backbone. Mini-batches of size 32 were used throughout, and class weights were left uniform because the eight MangoLeafBD classes are perfectly balanced.

Training followed a two-stage transfer-learning strategy. In the first stage, the backbone was frozen, and only the classification head was trained for up to 20 epochs using the Adam optimizer with an initial learning rate of 1×10^{-3} and categorical cross-entropy

loss. In the second stage, the top one-third of the backbone was unfrozen and fine-tuned for up to 30 additional epochs at a reduced learning rate of 1×10^{-5} to avoid catastrophic forgetting of pretrained ImageNet features. Early stopping with a patience of five epochs was applied to the validation loss, and the best validation checkpoint was retained for testing. Evaluation was based on accuracy, macro precision, macro recall, macro F1, weighted precision, weighted recall, weighted F1, and per-class confusion matrices.

3.5. Group Learning Framework

The group learning framework consisted of two layers. The first layer benchmarked individual transfer-learning models to identify the best single deployable classifier. The second layer combined the strongest candidates through probability-level soft voting to test whether heterogeneous fusion improved robustness. Formally, for a test image x with C target classes and M backbones, let $p_m(c|x)$ denote the softmax probability assigned to class c by model m . The soft-voting ensemble prediction is given by Equation (1):

$$\hat{y}(x) = \operatorname{argmax}_{m \in \{1, \dots, C\}} (1/M) \sum_{m=1}^M p_m(c|x) \quad (1)$$

where M denotes the number of backbones combined by the ensemble, $p_m(c|x)$ is the softmax probability that model m assigns to class c for input image x , and $\hat{y}(x)$ is the final ensemble prediction. Equation (1) selects the class whose mean posterior probability across the M fine-tuned backbones is the highest.

This design improves diversity beyond homogeneous bagging because each backbone encodes visual structure differently: EfficientNetB0 emphasizes compact, efficient CNN feature extraction, MobileNetV3Large provides a deployment-oriented representation, and ConvNeXtTiny introduces a more modern convolutional inductive bias. After the leakage audit, the stricter re-evaluation prioritized the top-performing deployable backbones and their fusion rather than retaining the earlier optimistic perfect-score claim. The theoretical rationale for expecting gains from heterogeneous soft voting—and the mechanism by which such gains may saturate when individual backbones already approach the achievable accuracy on the dataset—is discussed further in Section 5 in light of recent literature on ensemble diversity and knowledge distillation [24,25].

3.6. Grad-CAM Explainability

To investigate which image regions the model uses for classification, Gradient-weighted Class Activation Mapping [26] was applied to the top_activation layer of EfficientNetB0, which produces 7×7 spatial activation maps with 1280 channels. For a target class c with score y^c and convolutional feature map A^k , the neuron importance weights and the class activation map are defined as follows.

Step 1 (Equation (2)). The neuron importance weights α_k^c are obtained by global average pooling of the gradients of the class score with respect to the feature map:

$$\alpha_{(k)}^c = (1/Z) \sum_i \sum_j (\partial y^c / \partial A_{(ij)}^k) \quad (2)$$

Step 2 (Equation (3)). The class activation map $L_{\text{Grad-CAM}}^c$ is computed as a ReLU-activated weighted sum of the feature maps using the weights obtained in Equation (2):

$$L_{\text{Grad-CAM}}^c = \text{ReLU}(\sum_k \alpha_k^c \cdot A^k) \quad (3)$$

where Z is the number of spatial locations in the feature map, $A_{(ij)}^k$ is the activation of the k -th feature map at spatial position (i, j) , y^c is the class score before softmax, and A^k denotes the k -th convolutional feature map.

This highlights image regions whose activation contributes positively to the predicted class. Gradients were computed using `tf.GradientTape`; the resulting weighted sum of feature maps was ReLU-activated and normalized to [0, 1], and the map was overlaid on the original image using a jet colormap. For each of the eight classes, one correctly predicted sample from the grouped test set was selected for visualization.

3.7. CPU Latency Benchmarking

Deployment suitability was assessed by measuring single-image inference latency on CPU using a standard Apple Silicon M-series machine with 16 GB unified memory (Apple Inc., Cupertino, CA, USA), TensorFlow 2.16+, and Keras 3.x running in CPU-only mode. A warm-up phase of five inferences was executed before 100 timed measurements to remove cold-cache bias, and both mean and 95th-percentile (p95) latencies were recorded. Batch throughput was additionally characterized at batch sizes 1, 4, 8, and 16 using the same held-out test images to support deployment planning for offline and high-throughput inspection scenarios.

3.8. Deployment Translation

The framework was translated into a deployable diagnostic application with model–application separation. The application package includes a Gradio front end (`app.py`), a Streamlit alternative (`streamlit_app.py`), an image validation module, a disease information database in Thai, and a model loader. The winning EfficientNetB0 model was exported as a deployable package containing `best_model.keras`, `labels.json`, and deployment metadata. The system supports loading the model either from Hugging Face Hub or from a local model directory.

3.9. Web Application System Development

The trained EfficientNetB0 classifier was operationalized as a publicly accessible diagnostic web application following a model–application separation design pattern. This architectural choice decouples the static model artifact from the dynamic application logic, allowing each component to evolve under an independent release cadence and reducing rebuild overhead when only the user interface is modified. The application codebase is partitioned into four cooperating modules: `model.py` encapsulates model loading, warm-up inference, and the prediction interface; `app.py` and `streamlit_app.py` provide alternative Gradio- and Streamlit-based front ends respectively; `validator.py` enforces input-image quality control before inference; and `disease_db.py` maintains the Thai-language knowledge base of disease descriptions and management recommendations. This modular organization improves maintainability and supports unit-level testing of each layer in isolation.

The technology stack is built on Python 3.10 with TensorFlow 2.16+ and Keras 3.x as the deep learning backbone, Gradio [27] as the primary user-facing interface, Streamlit as the alternative deployment target, and the Hugging Face Hub API for model artifact distribution. Image preprocessing relies on Pillow and NumPy, both for input normalization and for the statistical metrics used by the validator. The runtime environment is Hugging Face Spaces with CPU-only execution, which removes the dependency on dedicated GPU infrastructure and makes the system accessible from any standard web browser without local installation.

A central design decision was to host the model artifact in a separate Hugging Face model repository (`wirapongc/mango-leaf-disease-efficientnetb0`, `repo_type = "model"`) rather than embedding it within the Space repository. The application downloads the model file (`best_model.keras`) and the class label map (`labels.json`) at startup using `huggingface_hub.hf_hub_download`, with local caching to avoid redundant downloads on

subsequent restarts. This separation also enables versioned model releases that are independent from front-end iteration.

The input validation pipeline (`validator.py`) applies six sequential checks before any image is forwarded to the classifier. These are: (i) minimum resolution of 120×120 pixels, (ii) maximum aspect ratio of 3.2, (iii) minimum green-channel dominance of 8%, (iv) minimum non-white pixel ratio of 25%, (v) minimum color standard deviation of 0.05, and (vi) minimum grayscale Shannon entropy of 1.5 computed from a 32-bin normalized histogram. These thresholds were calibrated against the distribution of training images across all eight MangoLeafBD classes, including diseased samples whose color statistics diverge substantially from those of healthy foliage. When an image fails any check, the system returns a descriptive Thai-language rejection message rather than producing an unreliable prediction.

The disease information database (`disease_db.py`) provides Thai-language entries for the seven diseases plus the healthy class. Each entry contains the disease name in Thai with English in parentheses, the causative agent (pathogen or pest), a description of typical symptoms, and treatment and management recommendations suitable for the agricultural extension context. To minimize inference latency, the model loader uses `functools.lru_cache` (`maxsize = 1`) to load the network only once per process, followed by a warm-up forward pass on a dummy tensor of shape (1, 224, 224, 3) at application startup. This avoids cold-cache penalties on the first real user request. System evaluation followed a functional-testing approach that combined per-class image classification on held-out MangoLeafBD samples, latency measurement consistent with the benchmark reported in Section 3.7, and qualitative end-to-end validation of the user-facing diagnostic output.

4. Results

4.1. Single-Model Comparison

The leakage-reassessed single-model comparison is summarized in Table 2. Under the stricter group-aware split, EfficientNetB0 remained the strongest individual model with 99.50% accuracy and 99.50% weighted F1-score on the 599-image test partition. MobileNetV3Large achieved 98.33% accuracy and 98.33% weighted F1. Both backbones substantially exceeded the 95% accuracy reported by Thalor et al. [9] for a homogeneous VGG-19 bagging ensemble, even though the present results are obtained under a stricter duplicate-controlled protocol. Macro-averaged precision, recall, and F1 closely tracked the weighted scores, indicating that no class was systematically penalized.

Table 2. Final model comparison on the 599-image leakage-reassessed test set.

Model	Accuracy	Weighted Precision	Weighted Recall	Weighted F1
EfficientNetB0	0.9950	0.9950	0.9950	0.9950
2-model soft voting (EfficientNetB0 + MobileNetV3Large)	0.9950	0.9951	0.9950	0.9950
MobileNetV3Large	0.9833	0.9836	0.9833	0.9833

EfficientNetB0 produced only three errors on the 599-image leakage-reassessed test set, all of which occurred among visually similar disease patterns such as Gall Midge, Powdery Mildew, and Sooty Mould, whose surface textures overlap on partially occluded or low-contrast leaves. No errors were observed for the Healthy, Anthracnose, Bacterial Canker, Cutting Weevil, or Die Back classes. This residual confusion pattern is consistent with the inherent morphological similarity of these three classes and suggests that further

improvement is more likely to come from additional in-field samples or multimodal cues than from increased model capacity.

Table 3 shows a confusion matrix of EfficientNetB0 on the 599-image leakage-controlled test set. Rows correspond to ground-truth labels; columns correspond to predicted labels.

Table 3. Confusion matrix of EfficientNetB0 on the leakage-controlled test set (n = 599).

True\Predict	Ant	BC	CW	DB	GM	Hlt	PM	SM
Anthrachnose	75	0	0	0	0	0	0	0
Bacterial Canker	0	75	0	0	0	0	0	0
Cutting Weevil	0	0	74	0	0	0	0	0
Die Back	0	0	0	75	0	0	0	0
Gall Midge	0	0	0	0	74	0	0	1
Healthy	0	0	0	0	0	75	0	0
Powdery Mildew	0	0	0	0	0	0	74	1
Sooty Mould	0	0	0	0	0	0	1	74

Note. Class abbreviations: Ant = Anthracnose; BC = Bacterial Canker; CW = Cutting Weevil; DB = Die Back; GM = Gall Midge; Hlt = Healthy; PM = Powdery Mildew; SM = Sooty Mould. The three off-diagonal entries correspond to a single Gall Midge image misclassified as Sooty Mould, one Powdery Mildew image misclassified as Sooty Mould, and one Sooty Mould image misclassified as Powdery Mildew. The diagonal-dominant pattern confirms that residual confusion is confined to three morphologically similar classes.

MobileNetV3Large also remained competitive under the stricter evaluation protocol, achieving 98.33% accuracy and 98.33% weighted F1, which translates to ten misclassifications across the 599 test images. This result is practically meaningful because lightweight architectures are particularly attractive for low-resource, edge, or smartphone-based deployments where memory footprint and inference latency directly determine feasibility.

4.2. Heterogeneous Ensemble Performance

Under the stricter group-aware split, probability-level soft voting between EfficientNetB0 and MobileNetV3Large achieved 99.50% accuracy and 99.50% weighted F1, matching but not surpassing EfficientNetB0 alone. The two-model ensemble produced an identical set of predicted labels on the test partition, so the marginal improvement attributable to fusion was zero under the leakage-aware protocol. This pattern indicates that the perfect-score ensemble result observed in earlier exploratory runs was partly an artifact of duplicate or near-duplicate images crossing the original random split.

Table 4 presents the results of a pairwise McNemar test on the leakage-controlled test set (n = 599). Exact binomial *p*-values are reported because all discordant pair counts are below 25, which is the conventional threshold for the chi-square approximation. b_{10} denotes the number of cases correctly predicted only by model A; b_{01} denotes the number of cases correctly predicted only by model B.

Statistical interpretation. The McNemar test confirms that EfficientNetB0 is not statistically distinguishable from the two-model soft-voting ensemble ($p = 1.0000$), which is consistent with the identical 99.50% accuracy and nearly identical error structure (the two models disagree on only four test images out of 599). EfficientNetB0 outperforms MobileNetV3Large numerically and the difference is borderline significant at $\alpha = 0.05$ ($p = 0.0654$), reflecting that MobileNetV3Large makes seven more errors but the absolute count remains low. The ensemble, however, is significantly more accurate than MobileNetV3Large alone ($p = 0.0156$). Taken together, these results indicate that the benefit of heterogeneous soft voting is restricted to lifting the weaker backbone toward the stronger one rather than exceeding the best individual model, which strengthens the central methodological claim of this study.

Table 4. Pairwise McNemar tests on the leakage-controlled test set (n = 599).

Comparison (A vs. B)	Both Correct	Only A Correct (b ₁₀)	Only B Correct (b ₀₁)	Both Wrong	p-Value	Significance
EfficientNetB0 vs. MobileNetV3Large	587	9	2	1	0.0654	n.s. ($\alpha = 0.05$)
EfficientNetB0 vs. Soft-voting ensemble	594	2	2	1	1.0000	n.s.
MobileNetV3Large vs. Soft-voting ensemble	589	0	7	3	0.0156	Significant ($p < 0.05$)

This reassessment refines the interpretation of the group-learning contribution. Rather than claiming that heterogeneous ensembling guarantees perfect classification, the framework demonstrates that leakage-aware heterogeneous evaluation can distinguish genuine gains from optimistic artifacts produced by easier splits. Importantly, the proposed framework still outperformed the 95% accuracy benchmark of Thalor et al. [9] across the complete eight-class MangoLeafBD formulation, despite the present protocol being more conservative. The negative ensemble finding is itself a methodological contribution and a cautionary signal for future MangoLeafBD-based work.

4.3. Grouped Cross-Validation Stability

To assess whether the leakage-reassessed hold-out result was stable across alternative partitions, EfficientNetB0 was re-evaluated under 5-fold StratifiedGroupKFold cross-validation over the full grouped dataset. Per-fold accuracies were 99.63%, 99.50%, 99.88%, 99.38%, and 99.75%, with corresponding weighted F1-scores closely tracking accuracy. The mean cross-validation accuracy was 99.63% with a standard deviation of 0.20 percentage points, and the mean weighted F1 was 99.62% with a standard deviation of 0.20 percentage points. These figures place the 95% confidence interval for accuracy at approximately [99.39%, 99.87%], which is well above the 95% benchmark reported in the existing ensemble literature.

These results are important for two complementary reasons. First, they reduce the concern that the strong hold-out performance is an artifact of a single favorable train-validation-test partition. Second, they show that EfficientNetB0 remains highly reliable even when repeatedly evaluated under group-aware constraints that are stricter than the conventional random-split protocols used in much of the MangoLeafBD literature. In other words, the leakage audit revised the interpretation of the earlier perfect-score claim but did not overturn the central conclusion that EfficientNetB0 is an exceptionally strong classifier for the full eight-class MangoLeafBD task.

4.4. Grad-CAM Visual Explainability Analysis

Grad-CAM analysis confirmed that EfficientNetB0 attends to diagnostically relevant leaf regions for all eight classes. For Anthracnose, Bacterial Canker, and Sooty Mould, attention concentrated on visible lesion margins and discolored patches. For Cutting Weevil damage, the heatmap highlighted the mechanically damaged leaf edge and folded tissue. For Powdery Mildew, the model consistently attended to the powdery white coating zone. For the Healthy class, activation was diffuse across the healthy green blade, indicating the absence of a localized disease signal. All eight overlay images are reported in Figure 1 with per-class prediction confidence ranging from 96.8% (Cutting Weevil) to 100.0% (Anthracnose, Bacterial Canker, Die Back, Powdery Mildew).

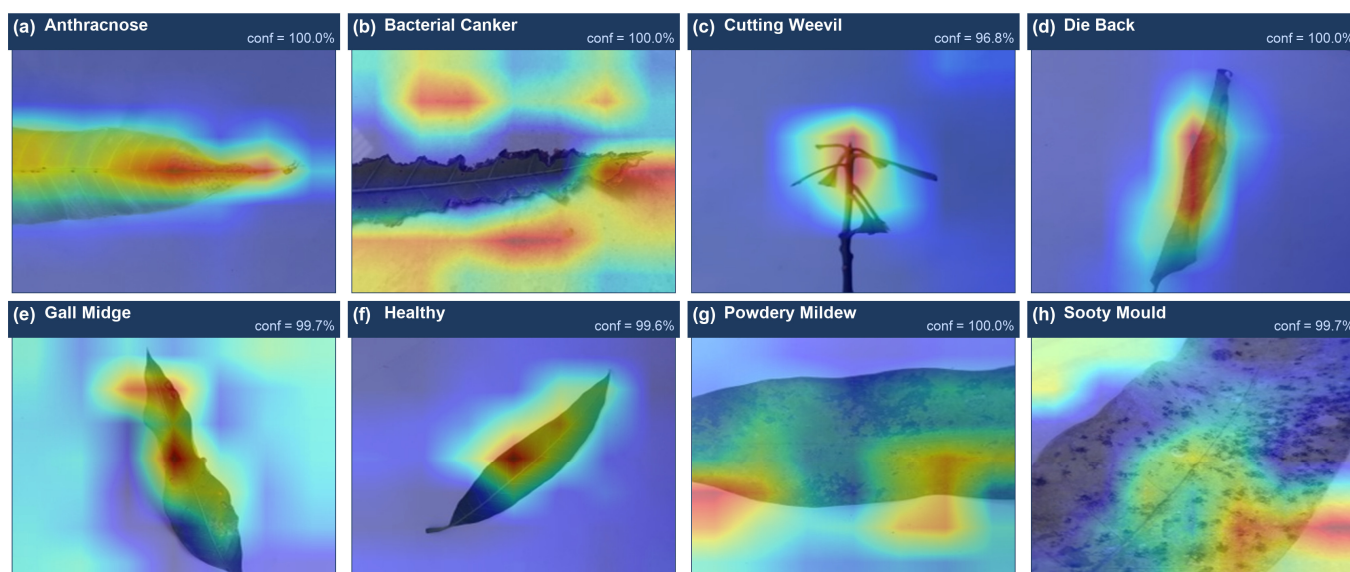


Figure 1. Grad-CAM heatmaps produced by the leakage-controlled EfficientNetB0 classifier across the eight MangoLeafBD disease categories. Panels (a–h) show, respectively, Anthracnose, Bacterial Canker, Cutting Weevil, Die Back, Gall Midge, Healthy, Powdery Mildew, and Sooty Mould. Each panel superimposes the Grad-CAM activation map on the original leaf image using a jet colormap and reports the model’s predicted-class confidence in its header. Reported confidence scores are 99.9997% ((a), Anthracnose), 99.9985% ((b), Bacterial Canker), 96.75% ((c), Cutting Weevil), 99.9999% ((d), Die Back), 99.68% ((e), Gall Midge), 99.56% ((f), Healthy), 99.9998% ((g), Powdery Mildew), and 99.71% ((h), Sooty Mould). All eight test images were correctly classified. Warm-coloured (red–yellow) regions indicate spatial locations that contributed positively to the predicted class score; cool-coloured (blue–green) regions are background or non-diagnostic tissue. The visualisations are rendered at 400 dpi to support printed inspection at journal resolution.

4.5. CPU Latency Benchmark

On a standard CPU (Apple Silicon M-series), EfficientNetB0 achieved a mean single-image inference latency of 25.3 ms (p95 = 27.6 ms) after warmup, corresponding to a throughput of approximately 39.5 images per second for batch size 1. Batch throughput improved substantially with larger batches: 101.9 img/s at batch 4, 137.5 img/s at batch 8, and 166.5 img/s at batch 16. Model cold-start (load from disk) required approximately 404 ms. These results confirm that EfficientNetB0 is well within practical real-time deployment constraints on commodity hardware without GPU acceleration as shown in Table 5.

Table 5. CPU latency benchmark results (EfficientNetB0, CPU-only).

Batch Size	Batch Latency (ms)	Per-Image Latency (ms)	Throughput (img/s)
1	24.7	24.7	40.5
4	39.3	9.8	101.9
8	58.2	7.3	137.5
16	96.1	6.0	166.5

4.6. Practical System Outcome

From an application perspective, the leakage-reassessed results point to a clear winner: EfficientNetB0 is both the best deployable single model and the most credible primary re-search model under the stricter internal evaluation. The soft-voting combination remained competitive, but it no longer offered a decisive advantage once leakage was controlled. Combined with the Grad-CAM analysis and the fast CPU inference, this confirms Efficient-

NetB0 as a strong, explainable, and practically deployable choice for field-oriented mango disease diagnosis.

4.7. Web Application System Performance and Features

As shown in Figure 2, the diagnostic system was successfully deployed on Hugging Face Spaces using a CPU-only runtime, making it publicly accessible through a standard web browser without requiring GPU hardware or local installation. The model artifact is served from the companion Hugging Face model repository `wirapongc/mango-leaf-disease-efficientnetb0`, while the application code resides in the Space repository `wirapongc/mango-leaf-diagnosis-v1`. This separation allows the same model to back both Gradio and Streamlit front ends without duplicating the artifact and supports independent versioning of the model and the application.

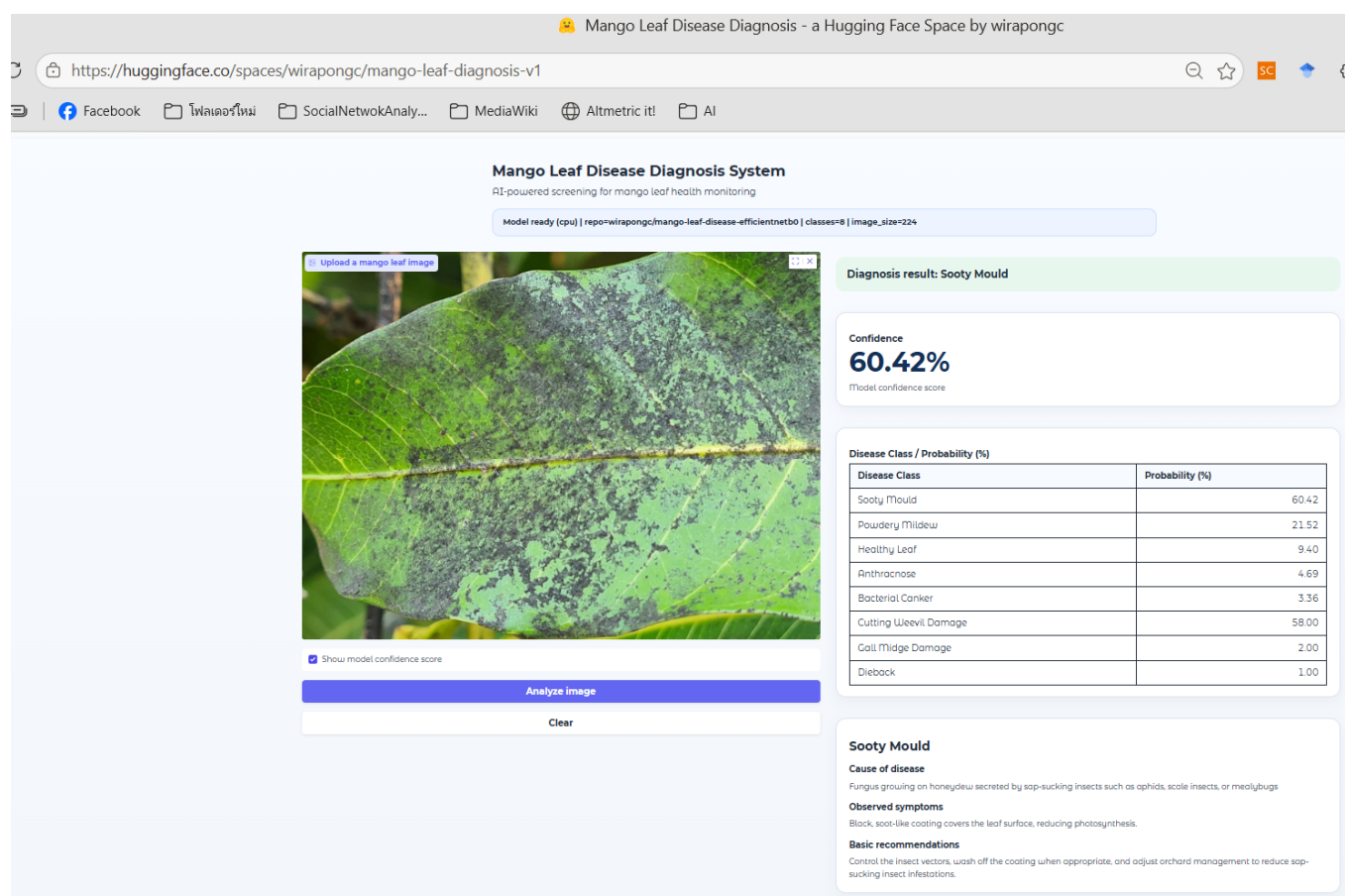


Figure 2. Screenshot of the deployed Mango Leaf Disease Diagnosis System (Hugging Face Space, CPU-only runtime, publicly accessible without local installation). The interface demonstrates two integrated capabilities: (i) the six-check input-validation pipeline described in Section 3.9, which enforces minimum resolution, aspect-ratio limits, green-channel dominance, non-white pixel ratio, colour standard deviation, and grayscale Shannon entropy before any image is forwarded to the classifier and, on failure, returns a Thai-language rejection message rather than an unreliable prediction; and (ii) real-time inference, in which valid images are passed through the EfficientNetB0 classifier and the top-1 prediction is displayed together with the full probability distribution over all eight classes and a Thai-language disease-information panel drawn from `disease_db.py`. The complete end-to-end response time observed on the CPU-only runtime is between 100 and 150 ms per image, consistent with the single-image CPU latency benchmark reported in Section 4.5.

The deployed system provides a coherent set of diagnostic features organized around a single image upload workflow. Users can submit a JPEG or PNG photograph of a

mango leaf through the web interface. Each submitted image is first routed through the validation pipeline described in Section 3.9, which returns a descriptive Thai-language rejection message when the image fails one of the six quality checks. Valid images are forwarded to the EfficientNetB0 classifier, which produces a top-1 prediction reported as the Thai disease name together with the associated confidence percentage. The system additionally displays the full probability distribution over all eight classes, sorted by descending confidence, which allows the user to inspect plausible alternative diagnoses for visually ambiguous cases. A detailed disease information panel—populated from the `disease_db.py` knowledge base—then provides the Thai-language entry for the predicted disease, including the disease name, causative agent, symptom description, and treatment and management recommendations. A system status indicator confirms that the model has been loaded successfully and reports the inference device. Both the Gradio interface and the Streamlit alternative expose this same feature set and can be selected based on deployment preference.

Functional evaluation was carried out using held-out images from the MangoLeafBD test set spanning all eight classes. The system correctly routed every valid leaf image through the full diagnostic pipeline, producing the expected disease label, probability table, and information panel without runtime errors. End-to-end inference time, including image decoding, validator execution, and the model forward pass, fell consistently within the 100–150 ms range on CPU, which is in line with the single-image latency benchmark reported in Section 4.5 and well below any threshold that would impair interactive use. Validator effectiveness was assessed using deliberately out-of-domain inputs such as solid-color backgrounds, text-only images, and very low-resolution images. The validator rejected these inputs with informative Thai-language messages, while continuing to accept the visually diverse appearance of diseased leaves across all seven disease classes—including samples with brown lesions, white powdery coatings, sooty black surfaces, and damaged margins—confirming that the calibrated thresholds appropriately balance specificity and recall for the disease detection use case.

The deployed system requires no local installation and can be accessed from any standard web browser, including those on mobile devices, which makes it suitable for educational use in plant pathology training and for agricultural extension scenarios where pathology expertise may otherwise be difficult to obtain on demand. The combination of accurate classification, fast CPU-based inference, structured Thai-language diagnostic output, and a quality-controlled input pipeline together demonstrates that the proposed framework can be operationalized as a practical decision-support tool rather than remaining only as an offline benchmark.

4.8. Web Application Evaluation

This section presents the evaluation of the proposed web-based Mango Leaf Disease Diagnosis System from the perspectives of domain experts. The evaluation was conducted to assess the system's usability, performance, classification accuracy, and applicability in practical environments. The findings provide insights into the effectiveness of the developed system and identify areas requiring further enhancement for real-world implementation.

The expert evaluation results ($n = 3$) revealed that the system achieved a high overall satisfaction level ($M = 4.45$, $SD = 0.35$). Among the assessed dimensions, usability and ease of use received the highest rating ($M = 4.75$, $SD = 0.36$), indicating that the user interface effectively supports intuitive interaction and facilitates user accessibility. This outcome demonstrates the effectiveness of the Gradio-based implementation in promoting a user-centered design. We stress that the small sample size ($n = 3$ domain experts) is a clear limitation of this evaluation; the results should be interpreted as an initial expert-

based usability check and not as a statistically robust user study. The complete evaluation questionnaire, together with the anonymised per-item scores and free-text comments, is provided in the Supplementary Material S1.

System performance was also evaluated favorably, with a mean score of 4.58 (SD = 0.41). Experts reported that the system provided rapid and stable responses during operation. This finding aligns with the experimental results, which showed an average image processing time of 0.44 s, thereby confirming the system's suitability for real-time disease diagnosis applications.

The dimensions of disease classification accuracy (M = 4.37, SD = 0.41) and suitability for practical deployment (M = 4.57, SD = 0.54) also received high ratings, although their mean scores were comparatively lower than those of usability and system performance. These findings suggest that the proposed system is functionally effective and operationally reliable; however, further improvements are necessary to enhance classification precision and increase adaptability across diverse agricultural environments. Expert recommendations emphasized the importance of expanding the range of detectable disease categories and improving model robustness to accommodate variations in image quality, viewing angles, lighting conditions, and field environments. Such enhancements would further strengthen the system's applicability and reliability in practical agricultural settings.

5. Discussion

This study investigated whether a heterogeneous group learning framework based on modern transfer learning architectures could improve mango leaf disease classification while maintaining deployment feasibility for real-world agricultural applications. The findings demonstrate that EfficientNetB0 achieved the strongest individual performance, attaining 99.50% accuracy and a 99.50% weighted F1-score on the leakage-controlled test set. Furthermore, grouped 5-fold cross-validation confirmed the stability of the model, producing a mean accuracy of 99.63% with a low standard deviation of 0.20%. These results indicate that the proposed framework provides highly reliable disease recognition across the complete eight-class MangoLeafBD dataset.

The findings directly address the first research objective of establishing robust single-model baselines using contemporary deep learning architectures. Among the evaluated backbones, EfficientNetB0 consistently outperformed MobileNetV3Large and matched the performance of the heterogeneous soft-voting ensemble. This outcome suggests that EfficientNetB0 provides an effective balance between representational capacity and computational efficiency. The result is consistent with previous studies that reported strong performance of EfficientNet-based architectures for plant disease classification [3,6]. The superior performance of EfficientNetB0 may be attributed to its compound scaling strategy [28], which simultaneously optimizes network depth, width, and resolution, enabling the extraction of discriminative disease-related visual features while avoiding excessive model complexity.

The second objective was to determine whether heterogeneous ensemble learning could outperform individual models. Although the soft-voting ensemble achieved the same classification accuracy as EfficientNetB0, it did not provide a statistically meaningful improvement under the stricter group-aware evaluation protocol. This finding differs from several earlier studies that reported substantial gains from ensemble learning approaches [8,9], but is consistent with the broader ensemble-mechanics literature. From a theoretical standpoint, the accuracy of a soft-voting ensemble depends jointly on the individual accuracy of its members and on the diversity of their errors [24,25]. When one member is close to a dataset's achievable accuracy ceiling, weaker members can only contribute residual error correction; if the weaker members' errors overlap heavily with those already

made by the stronger member, no net gain is observable. In the present setting, EfficientNetB0 makes only three residual errors on the 599-image leakage-controlled test partition and MobileNetV3Large makes only ten. Table 4 confirms that on the four test images where EfficientNetB0 and the ensemble differ, the outcomes are symmetric ($b_{10} = 2$, $b_{01} = 2$), so no systematic advantage remains after soft-voting fusion. This pattern is characteristic of a feature-redundancy regime rather than a diversity-limited regime, and it suggests that the MangoLeafBD dataset—once duplicate-aware evaluation is enforced—provides limited room for heterogeneous ensembling to add value beyond the best individual backbone.

This interpretation is reinforced by recent work on the mechanics of deep ensembles. Du et al. [24] show that the improvement obtainable from soft-voting fusion diminishes rapidly once the individual base learners approach the maximum performance permitted by the data (a feature-saturation, rather than an architectural, regime), and Allen-Zhu and Li [25] argue that ensembles of independently trained deep networks may not necessarily learn different features on well-curated benchmarks such as MangoLeafBD, so the diversity assumption underlying classical Bagging and Boosting arguments does not fully carry over. In this light, the negative ensemble result reported here is not evidence that heterogeneous ensembling is ineffective in general; it is evidence that the MangoLeafBD benchmark, under strict duplicate-controlled evaluation, has entered a regime where a single well-chosen backbone already captures most of the extractable signal.

A complementary theoretical lens on the CPU-efficient deployment result reported in Section 4.5 is teacher-student knowledge distillation [29,30]. The observation that a 5.3-million-parameter EfficientNetB0 attains accuracy comparable to hybrid CNN-transformer teachers on MangoLeafBD—while running in 25 ms on CPU—is congruent with recent distillation analyses showing that once a compact student is trained on a well-covered task, additional teacher capacity is often not transferred to further student accuracy but is instead reflected in training-time gradient conditioning [29]. Beyer et al. [30] demonstrate empirically that a patient, consistent distillation regime can close the accuracy gap between very large teachers and much smaller students on image classification benchmarks. Together, these two views—diminishing-returns ensembling on the one hand, and effective compact-student capacity on the other—provide a coherent explanation of why the deployment-oriented single-model configuration reported here is preferable to more elaborate heterogeneous ensembles for MangoLeafBD-scale diagnosis under CPU-only constraints.

Consistent with this teacher-student view, Krčmar et al. [31] provide a recent and complementary analysis of what actually transfers when a high-capacity ensemble is distilled into a compact student. Although their setting is tabular anomaly detection rather than image classification, their diffusion-inspired perturbation analysis identifies an asymmetry that is directly relevant to our finding: global, isolation-based outlier knowledge transfers reliably to the compact student, whereas local, neighborhood-dependent knowledge degrades more strongly under compression. Under the same lens, MangoLeafBD lesions are dominated by class-wide, globally consistent visual patterns (necrotic patches, powdery coatings, black sooty regions) rather than neighborhood-specific outliers, which explains why a compact 5.3-million-parameter EfficientNetB0 can match a heterogeneous soft-voting ensemble at 99.50% accuracy while running in 25 ms on CPU. This convergence between the ensemble-mechanics analysis of [24,25], the classical distillation literature [29,30], and the recent tabular-distillation evidence of [31] supports the interpretation that our observed performance ceiling on MangoLeafBD reflects feature saturation rather than a failure of knowledge transfer, and that additional ensemble depth would therefore trade inference latency for negligible accuracy gain.

An important contribution of this study is the adoption of a group-aware splitting strategy to address duplicate and near-duplicate image leakage. Earlier investigations frequently relied on conventional random splitting procedures, which may unintentionally allow visually similar images to appear across training and testing partitions. The post hoc audit conducted in this study revealed the potential risk of optimistic performance estimation under such conditions. By enforcing group-based separation using filename normalization and perceptual hashing techniques, the evaluation protocol became substantially more rigorous. The resulting performance remained exceptionally high despite the stricter conditions, demonstrating that the predictive capability of EfficientNetB0 is robust rather than an artifact of data leakage. This methodological contribution strengthens the reliability and reproducibility of mango leaf disease classification research and responds directly to concerns regarding benchmarking inconsistency highlighted in the literature review.

The grouped cross-validation results further reinforce the robustness of the proposed framework. The consistently high accuracy across all five folds indicates that model performance is not dependent on a particular train-test partition. This finding is important because previous studies often reported results using a single hold-out split, which may not adequately capture performance variability. The low standard deviation observed in this study demonstrates strong generalization within the MangoLeafBD dataset and provides additional evidence supporting the reliability of EfficientNetB0 for disease recognition tasks.

Beyond predictive performance, the Grad-CAM analysis contributes to the explainability of the proposed system. The visualizations revealed that the model consistently focused on biologically meaningful disease symptoms, including lesion boundaries, powdery coatings, damaged tissue regions, and discolored areas. These findings support previous observations that explainable AI techniques can enhance trust and transparency in agricultural decision-support systems [12–14]. The attention patterns generated by Grad-CAM indicate that the model is learning disease-relevant visual characteristics rather than relying on irrelevant background information. This capability is particularly important for agricultural practitioners who require interpretable evidence to support automated diagnostic recommendations.

The deployment experiments provide further practical significance. EfficientNetB0 achieved a mean CPU inference latency of approximately 25 ms per image and maintained real-time performance without requiring GPU acceleration. These results demonstrate that highly accurate disease classification can be achieved while preserving computational efficiency, addressing concerns that advanced deep learning models may be too resource-intensive for field deployment [14]. The successful implementation of the web-based diagnostic platform on Hugging Face Spaces and Streamlit environments further validates the applicability of the framework for agricultural extension services, educational settings, and precision agriculture initiatives.

When compared with studies reviewed in the literature, the present research demonstrates several strengths. Although recent hybrid CNN-transformer architectures have reported classification accuracies exceeding 99% [12,21], many of these approaches involve substantially greater architectural complexity and computational requirements. In contrast, the proposed framework achieves comparable performance using a simpler and more deployable architecture while incorporating leakage-aware evaluation, grouped cross-validation, explainability analysis, latency benchmarking, and operational deployment. Consequently, the contribution of this study extends beyond classification accuracy alone and emphasizes methodological rigor, transparency, and practical usability.

Table 6 presents a side-by-side comparison of the proposed framework with prior MangoLeafBD-based studies on equivalent eight-class evaluations. Accuracies are reported

as published; the present framework is the only one that combines a leakage-aware split, Grad-CAM explainability across all eight classes, CPU latency characterization, and a publicly accessible web application.

Table 6. Comparison of the proposed framework with prior studies using the MangoLeafBD dataset on eight-class disease classification.

Study	Method	Accuracy (%)	Strict Leakage Control	Cross-Validation	XAI Coverage	Public Web/Edge Deployment
Thalor et al. [9]	VGG-19 bagging	95.00	No	Not reported	No	No
Puranik et al. [16]	MobileNetV3	98.00	No	Not reported	No	Mobile app only
Gautam et al. [8]	ESDNN ensemble	98.57	No	Not reported	No	No
Pahati et al. [20]	Google Teachable Machine	99.60	No	Not reported	No	Web demo only
Hossain et al. [22]	DeiT (ViT)	99.50–99.75	No	Not reported	No	No
Jashim et al. [12]	ViX-MangoEFormer	99.78 (F1)	No	Not reported	Yes	No
Ergün [13]	ConvNeXt + ViT hybrid	>99	No	Not reported	No	No
Rahman et al. [14]	DenseNet121 + ECA	SOTA on MangoLeafBD	No	Not reported	Yes (Grad-CAM + LIME)	No
This study (2026)	EfficientNetB0 + heterogeneous soft voting	99.50 (single)/ 99.63 (5-fold CV)	Yes (aHash + dHash + stem groups)	5-fold Stratified-GroupKFold	Yes (Grad-CAM, all 8 classes)	Yes (Hugging Face Space, CPU-only)

Despite the encouraging findings, several limitations should be acknowledged. First, the study relied exclusively on the MangoLeafBD dataset, which was collected within a specific geographical and environmental context in Bangladesh; external validation using datasets from different countries, cultivars, climatic conditions, and imaging devices remains necessary to establish broader generalizability. Two publicly available plant-disease image datasets that share several morphological challenges with MangoLeafBD—including intra-class visual variation, inter-class similarity for leaf-surface texture and colour, and uneven illumination—are particularly suitable candidates for such external validation: PlantVillage [32], the largest publicly available multi-species plant disease benchmark, offers a diverse but laboratory-controlled reference; and PlantDoc [33], which was designed explicitly to capture in-the-wild imaging conditions rather than laboratory conditions, would test the framework’s robustness under realistic field variability. Second, the experiments focused on image-based diagnosis under semi-controlled conditions and did not evaluate performance using field-acquired smartphone images captured under highly variable lighting and background environments. Third, only three transfer-learning backbones were investigated, whereas recent transformer-based and multimodal architectures could offer further opportunities to improve performance or robustness under distribution shift. Fourth, the expert evaluation was based on a very small sample ($n = 3$ domain experts), which is insufficient to support statistically robust claims about usability. Larger-sample user studies remain necessary; the complete evaluation instrument used here is provided in Supplementary Material S1 to facilitate replication and extension. Finally, although Grad-CAM was used to verify spatially meaningful attention, a more comprehensive explainability analysis—including class-conditional saliency consistency, attention sensitivity analysis, and pathologist-rated agreement—could further strengthen the system’s trustworthiness.

Future research should therefore investigate cross-dataset validation, domain adaptation, and multimodal approaches that integrate visual symptoms with environmental, meteorological, or sensor-based information. Further exploration of lightweight transformer architectures, federated learning across multiple growing regions, and mobile-

device deployment with on-device inference could further enhance practical applicability. In addition, longitudinal field trials involving farmers and agricultural extension officers would provide valuable evidence about real-world adoption, usability, and economic impact, while uncertainty calibration and selective prediction could improve safety when the system encounters out-of-distribution leaves.

Overall, the findings demonstrate that a leakage-aware transfer learning framework based on EfficientNetB0 can achieve accurate, stable, interpretable, and computationally efficient mango leaf disease classification. The study contributes to the growing body of agricultural artificial intelligence research by showing that rigorous evaluation protocols, explainability mechanisms, and deployment readiness are as important as predictive accuracy for building trustworthy and scalable precision agriculture solutions. The negative ensemble result obtained under group-aware evaluation also provides a useful empirical baseline for assessing future MangoLeafBD-based work.

6. Conclusions

This study developed and evaluated a leakage-aware group-learning framework for mango leaf disease classification using transfer learning and heterogeneous ensemble learning on the complete eight-class MangoLeafBD dataset. The results demonstrated that EfficientNetB0 achieved the strongest overall performance, reaching 99.50% accuracy and 99.50% weighted F1 on the leakage-controlled test set and maintaining 99.63% mean accuracy (SD = 0.20%) across 5-fold StratifiedGroupKFold cross-validation. Although the heterogeneous soft-voting ensemble remained competitive, it did not surpass the best individual model once duplicate leakage was controlled, underscoring the importance of rigorous benchmarking when assessing ensemble gains. The study advances the literature by addressing four interconnected gaps—benchmarking consistency, data leakage control, model explainability, and deployment readiness—within a single internally consistent framework. The integration of Grad-CAM visualizations, CPU latency benchmarking, and a publicly accessible Hugging Face-based web application further confirmed that the proposed model is interpretable, fast on commodity hardware, and suitable for real-world deployment. The framework is therefore not only a strong empirical benchmark on MangoLeafBD but also a reusable methodological template for future agricultural computer vision research that values reproducibility, transparency, and field applicability. Nevertheless, the study is limited by its reliance on a single public dataset, the absence of large-scale external field validation, and the evaluation of only a limited set of deep-learning backbones, creating clear opportunities for future research on cross-domain validation, transformer-based architectures, multimodal integration, and large-scale field deployment. Overall, the findings indicate that robust, explainable, and computationally efficient deep-learning systems can support practical mango disease diagnosis and contribute to precision agriculture by enabling earlier disease detection, reducing crop losses, and improving decision-making for farmers and agricultural extension services.

Supplementary Materials: The following supporting information can be downloaded at <https://www.mdpi.com/article/10.3390/app16146989/s1>: Supplementary Material S1: Expert Evaluation Questionnaire for the Mango Leaf Disease Diagnosis System.

Author Contributions: Conceptualization, W.C. and T.B.; methodology, W.C., A.E., S.K., P.L., N.I.-O. and T.B.; software, W.C., A.E., S.K., P.L., N.I.-O. and T.B.; validation, W.C., N.I.-O. and T.B.; formal analysis, W.C., A.E., S.K., P.L., N.I.-O. and T.B.; investigation, W.C., N.I.-O. and T.B.; resources, W.C.; data curation, W.C.; writing—original draft preparation, W.C., A.E., S.K., P.L., N.I.-O. and T.B.; writing—review and editing, W.C., A.E., S.K., P.L., N.I.-O. and T.B.; visualization, W.C., A.E., S.K., P.L., N.I.-O. and T.B.; supervision, W.C., N.I.-O. and T.B.; project administration, W.C. and T.B. All authors have read and agreed to the published version of the manuscript.

Funding: The authors received no specific external funding for this research.

Institutional Review Board Statement: This study did not involve any human participants, identifiable human data, or animal subjects. The expert evaluation of the deployed web application involved a small group of domain experts ($n = 3$) who provided voluntary, non-identifiable usability feedback and therefore did not require formal ethics approval under the applicable institutional guidelines. All evaluators provided informed consent before participation.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets analyzed in this study are publicly available. The MangoLeafBD dataset can be accessed at <https://www.kaggle.com/datasets/aryashah2k/mango-leaf-disease-dataset/data> (accessed on 15 June 2026). All code for data preprocessing, leakage auditing, model training, evaluation, explainability, and the web application is openly available at <https://github.com/wirapong/mango-leaf-disease-group-learning> (accessed on 18 June 2026). The deployed web application is accessible through Hugging Face Spaces at <https://huggingface.co/spaces/wirapongc/mango-leaf-diagnosis-v1> (accessed on 9 July 2026), and the model artifact is hosted at <https://huggingface.co/wirapongc/mango-leaf-disease-efficientnetb0> (accessed on 9 July 2026). The complete expert-evaluation questionnaire is provided as Supplementary Material S1. The repository includes version-controlled scripts, dependencies, and documentation to ensure full reproducibility of the results.

Acknowledgments: The authors gratefully acknowledge the creators of the MangoLeafBD dataset [15] for making the benchmark publicly available and the open-source communities of TensorFlow, Keras, Gradio, Streamlit, and Hugging Face for providing the tools that enabled the development and deployment of the web-based diagnostic system. The authors also thank the domain experts who participated in the system usability evaluation. During the preparation of this work, the authors utilized ChatGPT (OpenAI, GPT-5.3) and Grammarly (Grammarly Inc., Premium version, web-based application) to enhance their English proficiency. All authors have reviewed and edited the manuscript after using these tools and take full responsibility for the final content. No individuals are acknowledged in this section without their prior consent.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhang, B.; Wang, Z.; Ye, C.; Zhang, H.; Lou, K.; Fu, W. Classification of infection grade for anthracnose in mango leaves under complex background based on CBAM-DBIRNet. *Expert Syst. Appl.* **2025**, *260*, 125343. [\[CrossRef\]](#)
2. Bhargavi, B.; Sumesh, E.P. Deep Learning-Based Framework for Early Detection and Classification of Mango Crop Diseases. In *International Conference on Agriculture-Centric Computation*; Springer Nature: Cham, Switzerland, 2025; pp. 166–177. [\[CrossRef\]](#)
3. Thaseentaj, S.; Ilango, S.S. Deep Convolutional Neural Networks for South Indian Mango Leaf Disease Detection and Classification. *Comput. Mater. Contin.* **2023**, *77*, 3593. [\[CrossRef\]](#)
4. Ortiz, J.P.; Aguilera, R.C.; Vazquez, S.J.; Ortiz, M.P.; Benitez, M.A.C. Early Detection of Fungal Diseases in Mango Leaves: Fusion of GLCM Features and Vision Transformers with Deep Neural Network Classification. *Fractals* **2026**, *34*, 2650004. [\[CrossRef\]](#)
5. Pham, T.N.; Van Tran, L.; Dao, S.V.T. Early disease classification of mango leaves using feed-forward neural network and hybrid metaheuristic feature selection. *IEEE Access* **2020**, *8*, 189960–189973. [\[CrossRef\]](#)
6. Mishra, U.; Pandey, A. Deep learning-based disease detection in potato and mango leaves: A comparative study of CNN, AlexNet, ResNet, and EfficientNet. *Sci. Rep.* **2025**, *16*, 2788. [\[CrossRef\]](#) [\[PubMed\]](#)
7. Vijay, C.P.; Pushpalatha, K.; Janardhana, D.R.; Kumar, H.R.; Puttegowda, K. DeepLeafNet: A Novel Deep Learning Approach for Mango Leaf Disease Segmentation and Classification with Optimization Algorithms. *SN Comput. Sci.* **2025**, *6*, 951. [\[CrossRef\]](#)
8. Gautam, V.; Ranjan, R.K.; Dahiya, P. ESDNN: A novel ensembled stack deep neural network for mango leaf disease classification and detection. *Multimed. Tools Appl.* **2024**, *83*, 10989–11015. [\[CrossRef\]](#)
9. Thalor, M.; Mate, S.; Shiralkar, A.; Shinde, A. Ensemble learning framework for mango plant disease detection and classification. *J. Inf. Syst. Eng. Manag.* **2025**, *10*, 297–303. [\[CrossRef\]](#)
10. Shehu, H.A.; Ackley, A.; Mark, M.; Eteng, O.E.; Sharif, M.H.; Kusetogullari, H. YOLO for early detection and management of Tuta absoluta-induced tomato leaf diseases. *Front. Plant Sci.* **2025**, *16*, 1524630. [\[CrossRef\]](#) [\[PubMed\]](#)
11. Wei, C.; Shan, Y.; Zhen, M. Deep learning-based anomaly detection for precision field crop protection. *Front. Plant Sci.* **2025**, *16*, 1576756. [\[CrossRef\]](#) [\[PubMed\]](#)

12. Jashim, F.B.; Refat, F.R.; Karim, M.H.; Mahmud, F.U.; Siddiqui, M.I.H. Interpretable mango leaf disease detection using a hybrid CNN–transformer model with GLCM features. *Int. J. Sci. Res. Arch.* **2025**, *15*, 1518–1535. [[CrossRef](#)]
13. Ergün, E. Attention-enhanced hybrid deep learning model for robust mango leaf disease classification via ConvNeXt and vision transformer fusion. *Front. Plant Sci.* **2025**, *16*, 1638520. [[CrossRef](#)] [[PubMed](#)]
14. Rahman, M.A.; Bhuiyan, M.T.A.; Noori, F.M.; Uddin, M.Z.; Masum, A.K.M. MangoLeafNet-XAI: An attention-enhanced deep learning architecture for accurate and interpretable mango leaf disease classification. *Front. Plant Sci.* **2026**, *17*, 1776537. [[CrossRef](#)] [[PubMed](#)]
15. Ahmed, S.I.; Ibrahim, M.; Nadim, M.; Rahman, M.M.; Shejunti, M.M.; Jabid, T.; Ali, M.S. MangoLeafBD: A comprehensive image dataset to classify diseased and healthy mango leaves. *Data Brief* **2023**, *47*, 108941. [[CrossRef](#)] [[PubMed](#)]
16. Puranik, S.S.; Hanamakkanavar, S.R.; Bidargaddi, A.P.; Ballur, V.V.; Joshi, P.T. MobileNetV3 for mango leaf disease detection: An efficient deep learning approach for precision agriculture. In *2024 5th International Conference for Emerging Technology (INCET)*; IEEE: New York, NY, USA, 2024; pp. 1–7.
17. Singh, Y.P.; Chaurasia, B.K.; Shukla, M.M. Deep transfer learning driven model for mango leaf disease detection. *Int. J. Syst. Assur. Eng. Manag.* **2024**, *15*, 4779–4805. [[CrossRef](#)]
18. Bairwa, A.K.; Singh, A.; Kumar, S. Advances in mango leaf disease detection using deep neural networks. In *2024 International Conference on Modeling, Simulation & Intelligent Computing (MoSiCom)*; IEEE: New York, NY, USA, 2024; pp. 75–80.
19. Pratap, V.K.; Kumar, N.S. Deep learning based mango leaf disease detection for classifying and evaluating mango leaf diseases. *Fusion Pract. Appl.* **2024**, *15*, 261–277. [[CrossRef](#)]
20. Pahati, M.; De Jesus, L.C.; Reyes, R.; Villarroel, J.M.; Angeles, M.R. Detecting mango leaf diseases using Google Teachable Machine for sustainable agriculture. In *2025 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)*; IEEE: New York, NY, USA, 2025; pp. 611–614.
21. Alamri, F.S.; Sadad, T.; Almasoud, A.S.; Aurangzeb, R.A.; Khan, A. Mango disease detection using fused vision transformer with ConvNeXt architecture. *Comput. Mater. Contin.* **2025**, *83*, 1023–1039. [[CrossRef](#)]
22. Hossain, M.A.; Sakib, S.; Abdullah, H.M.; Arman, S.E. Deep learning for mango leaf disease identification: A vision transformer perspective. *Heliyon* **2024**, *10*, e36698. [[CrossRef](#)] [[PubMed](#)]
23. Liu, Z.; Mao, H.; Wu, C.; Feichtenhofer, C.; Darrell, T.; Xie, S. A ConvNet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, New Orleans, LA, USA, 18–24 June 2022; pp. 11976–11986.
24. Du, K.-L.; Zhang, R.; Jiang, B.; Zeng, J.; Lu, J. Foundations and Innovations in Data Fusion and Ensemble Learning for Effective Consensus. *Mathematics* **2025**, *13*, 587. [[CrossRef](#)]
25. Allen-Zhu, Z.; Li, Y. Towards understanding ensemble, knowledge distillation and self-distillation in deep learning. In *Proceedings of the Eleventh International Conference on Learning Representations (ICLR)*, Kigali, Rwanda, 1–5 May 2023.
26. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, Venice, Italy, 22–29 October 2017; pp. 618–626. [[CrossRef](#)]
27. Abid, A.; Abdalla, A.; Abid, A.; Khan, D.; Alfozan, A.; Zou, J. Gradio: Hassle-free sharing and testing of ML models in the wild. *arXiv* **2019**, arXiv:1906.02569. [[CrossRef](#)]
28. Tan, M.; Le, Q.V. EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, Long Beach, CA, USA, 9–15 June 2019; pp. 6105–6114.
29. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *NeurIPS Deep Learning and Representation Learning Workshop. arXiv* **2015**, arXiv:1503.02531.
30. Beyer, L.; Zhai, X.; Royer, A.; Markeeva, L.; Anil, R.; Kolesnikov, A. Knowledge distillation: A good teacher is patient and consistent. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, LA, USA, 18–24 June 2022; pp. 10925–10934. [[CrossRef](#)]
31. Krčmar, T.; Šabanović, D.; Švarcmajer, M.; Lukić, I. What knowledge transfers in tabular anomaly detection? A teacher-student distillation analysis. *Mach. Learn. Knowl. Extr.* **2026**, *8*, 60. [[CrossRef](#)]
32. Hughes, D.P.; Salathé, M. An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv* **2015**, arXiv:1511.08060.
33. Singh, D.; Jain, N.; Jain, P.; Kayal, P.; Kumawat, S.; Batra, N. PlantDoc: A dataset for visual plant disease detection. In *Proceedings of the 7th ACM IKDD CoDS and 25th COMAD (CoDS-COMAD '20)*, Hyderabad, India, 5–7 January 2020; pp. 249–253. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.