PlantaNet and PlantaNetLite: Efficient and Explainable Multi-Crop Plant Disease Classification via Transformer Benchmarking and Custom Lightweight CNNs

Md. Sifat Haque Zidan[1], Al Imran[1], Md. Kayes Mia[1], Muhammad Hussain[1], Ahmed Faizul Haque Dhrubo[1], Mohammad Abdul Qayum[1*]

[1]Department of Electrical and Computer Engineering, North South University, Dhaka, 1229, Bangladesh

*mohammad.qayum@northsouth.edu

Abstract

Plant disease diagnosis based on visual symptoms is crucial for preventing yield loss; however, real-world deployment remains challenging due to inter-class similarity, background noise, and limited computational resources. This paper proposes a large-scale plant disease classification framework evaluated on a curated dataset of 136,482 images across 51 classes, covering multiple crops including corn, rice, tomato, potato, banana, grape, apple, mango, and strawberry. Starting with approximately 45,000 original non-augmented samples, the dataset is expanded using an Albumentations-based augmentation pipeline to improve generalization. We benchmark eight tiny vision transformer architectures trained for up to 50 epochs, where CAFormer_s18 achieves the highest validation accuracy (99.89%) at the cost of increased computational overhead. To enable efficient and deployable solutions, we design two fully customized lightweight convolutional neural networks: PlantaNetLite (1.28M parameters) and PlantaNet (2.58M parameters). Hyperparameter tuning identifies an optimal configuration for PlantaNet (learning rate 1×10^{-3}, weight decay 1×10^{-4}, MixUp $\alpha = 0.2$), achieving approximately 99.5% validation accuracy after 100 epochs with a compact model size (9.85 MB) and feasible computational cost (1213.9 M FLOPs). PlantaNetLite also achieves strong validation accuracy (99.19%), supporting ultra-lightweight deployment scenarios. Finally, Grad-CAM and Grad-CAM++ are applied to validate interpretability, confirming that the proposed models consistently focus on biologically meaningful lesion regions rather than background artifacts. Overall, both PlantaNetLite and PlantaNet are suitable for practical deployment, with PlantaNetLite targeting ultra-lightweight edge scenarios and PlantaNet offering a higher-capacity accuracy–efficiency trade-off.

Author Summary

Plant diseases are a major threat to agricultural productivity, particularly in regions where access to expert diagnosis is limited. While recent advances in artificial intelligence have shown strong potential for identifying plant diseases from images, many existing models are computationally expensive and difficult to deploy on mobile devices or low-power systems commonly used in real-world farming environments. In addition, these models often function as “black boxes,” making it hard for users to trust their predictions. In this study, we developed two lightweight and efficient deep learning

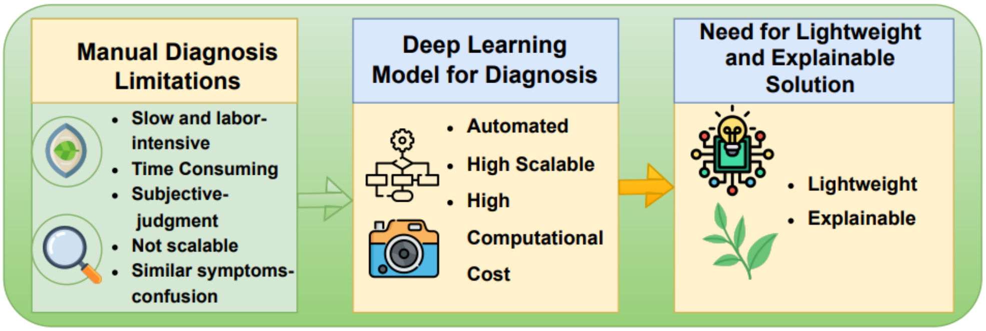


Fig 1. Motivation and need for a lightweight and explainable deep learning–based plant disease diagnosis framework.

models, PlantaNetLite and PlantaNet, designed specifically for practical plant disease diagnosis under resource constraints. Using a large and diverse dataset covering multiple crops and disease types, we show that these models achieve very high classification accuracy while remaining compact and computationally feasible. Importantly, our models also provide visual explanations that highlight disease-affected regions on plant leaves, helping ensure that predictions are based on meaningful biological features rather than background noise. This work demonstrates that accurate, efficient, and interpretable plant disease diagnosis systems can be deployed in real-world agricultural settings, supporting scalable and accessible disease monitoring for farmers and practitioners.

Introduction

Plant diseases cause substantial yield and quality losses worldwide, particularly in staple and high-value crops. Early and accurate diagnosis is critical, as many diseases spread rapidly and exhibit visible symptoms only after infection has progressed. In resource-limited agricultural environments, disease identification still relies heavily on expert inspection, which is time-consuming, subjective, and often unavailable at scale. Consequently, automated plant disease recognition from leaf images has emerged as a key research direction for advancing precision agriculture and sustainable crop management. Recent surveys indicate that computer vision–based disease detection has made significant progress, with deep learning techniques dominating current solutions due to their ability to learn discriminative visual representations directly from data [1, 2].

Traditional machine-learning approaches for plant disease diagnosis relied on handcrafted features, such as color, texture, and shape descriptors, followed by classical classifiers. These methods are highly sensitive to variations in illumination, background clutter, scale, and intra-class symptom diversity. Deep convolutional neural networks (CNNs) address many of these limitations by hierarchically extracting robust visual features and have demonstrated strong performance on benchmark datasets, including PlantVillage and real-field collections [3, 4]. However, despite their high accuracy, many CNN-based systems remain computationally intensive and difficult to deploy on edge devices. In addition, their decision-making processes are often opaque to agronomists and farmers. These limitations highlight two persistent challenges in the literature: (i) the need for lightweight, high-accuracy models suitable for practical deployment, and (ii) the integration of explainable artificial intelligence (XAI) techniques to improve model transparency and trust [1, 2].

Fig. 1 illustrates the motivation behind adopting a lightweight and explainable deep learning approach for plant disease diagnosis.

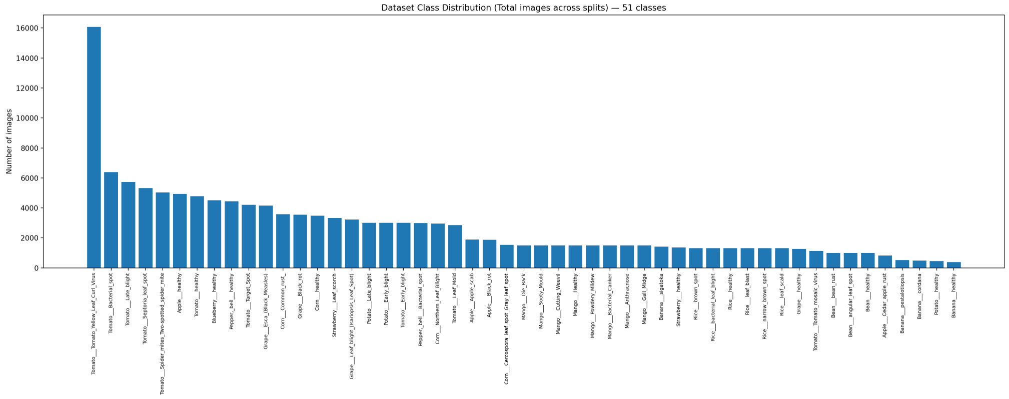


Fig 2. Dataset Class Distribution(Total images across splits)-51 classes.

In parallel with CNN advancements, transformer-based vision models have emerged as competitive alternatives. Vision Transformers (ViTs) and their compact variants are capable of modeling long-range dependencies and global contextual information, which can be beneficial for recognizing subtle disease patterns distributed across leaf surfaces. Several recent studies report that transformer architectures can match or outperform CNN baselines for plant disease classification when sufficient training data and strong augmentation strategies are available [5–9]. Nevertheless, transformer-based models typically incur higher computational costs and may exhibit degraded performance under limited training data or domain-shift conditions. This motivates continued investigation into efficient CNN-based and hybrid architectures that balance classification accuracy with deployment feasibility [1, 2].

Another critical challenge in plant disease recognition is dataset realism. Many publicly available datasets consist of laboratory-style images with uniform backgrounds, whereas real-world field images are affected by occlusion, mixed illumination, overlapping leaves, and complex background textures. To reduce this gap, recent research emphasizes robust data augmentation strategies and large-scale multi-crop datasets to improve generalization from controlled environments to real agricultural fields [1, 2, 10].

Beyond predictive performance, interpretability has become increasingly important for agricultural AI systems. For real-world adoption, models must demonstrate that their predictions are based on biologically meaningful disease symptoms rather than spurious background artifacts. Gradient-based localization methods, such as Grad-CAM and Grad-CAM++, are widely used explainability techniques that generate visual heatmaps highlighting influential regions in input images. Prior work shows that these methods can verify whether models attend to lesions, blights, mildew, or chlorosis patterns consistent with plant pathology [3, 11, 12]. Since Grad-CAM++ provides improved localization for multiple or fine-grained disease regions, employing both methods strengthens the interpretability analysis and trustworthiness of deep learning models [12].

0.1 Research Motivation and Contributions

Despite extensive research, several key limitations persist in existing plant disease classification studies. First, many high-performing models exhibit a significant accuracy–efficiency trade-off, making them unsuitable for deployment in resource-constrained environments [13, 14]. Second, a large portion of the literature evaluates either CNNs or transformer-based models in isolation, without systematic

Table 1. Comparison of the proposed pipeline with typical prior work.

Study Type	**Multi-crop (10+ crops)**	**Large-scale (≥100k)**	**Tiny transformer baselines**	**Lightweight CNN (≤3M params)**	**Hyperparameter tuning**	**Grad-CAM**	**Grad-CAM++**
Typical CNN-only papers	Yes / No	No / Yes	No	No / Yes	No / Yes	Yes	No
Typical ViT-only papers	Yes / No	No / Yes	Yes	No	No / Yes	Yes / No	No / Yes
This work	Yes	Yes	Yes (8 models)	Yes (1.28M, 2.58M)	Yes (Config-3)	Yes	Yes

cross-family comparisons under consistent experimental settings [1, 2]. Third, explainability is often treated as an auxiliary visualization rather than being rigorously validated to confirm symptom-level attention [11, 12].

To address these limitations, this paper makes the following contributions:

- We curate and preprocess a large-scale multi-crop plant disease dataset comprising 136,482 images across 51 classes, augmented using a controlled Albumentations-based pipeline to improve robustness and generalization [10].
- We conduct a systematic benchmark of eight compact vision transformer architectures trained under identical conditions to establish strong and fair attention-based baselines.
- We propose two fully customized lightweight CNN architectures, PlantaNetLite (1.28M parameters) and PlantaNet (2.58M parameters), designed specifically for efficient and accurate plant disease recognition [14, 15].
- We perform hyperparameter-driven optimization using learning rate, weight decay, and MixUp regularization to identify configurations that maximize validation performance [16].
- We validate model interpretability using Grad-CAM and Grad-CAM++, confirming that the proposed models consistently focus on biologically meaningful disease regions across all classes [11, 12].

Due to their compact sizes and high classification accuracy, both PlantaNetLite and PlantaNet are suitable for deployment in practical agricultural diagnostic systems. PlantaNetLite is particularly well suited for mobile and edge-based devices in resource-constrained farming environments, while PlantaNet provides a favorable accuracy–efficiency trade-off for scenarios where slightly higher computational resources are available, enabling reliable real-time disease monitoring.

Related Work

Early research on image-based plant disease recognition relied on handcrafted features (e.g., color histograms, texture operators, and shape descriptors) combined with classical classifiers such as SVM and Random Forest. Although these approaches are computationally efficient, their performance often degrades under real-field variations including illumination changes, complex backgrounds, and partial occlusions, which limits practical deployment. The transition to deep learning improved robustness by enabling feature learning directly from raw images. A landmark study by Mohanty *et al.* demonstrated the effectiveness of transfer-learned CNNs on PlantVillage, establishing a strong baseline for modern plant disease classification [3].

0.2 CNN-Based Plant Disease Classification

CNNs remain widely adopted due to their strong inductive bias for local pattern learning, which is well-suited for recognizing lesions, spots, mildew, and texture-based symptoms. Numerous studies from 2023–2025 report improved accuracy across crops such as rice, tomato, grape, apple, and maize using deeper backbones (e.g., ResNet, DenseNet, EfficientNet) and task-specific refinements [1, 2, 4, 13]. Prior work also indicates that well-tuned CNNs can outperform transformer variants in visually subtle plant disease settings, as convolutional filters naturally emphasize localized symptom cues [1, 2].

Lightweight and mobile CNNs represent another important direction. Comparative studies of compact architectures such as MobileNet and ShuffleNet demonstrate that high accuracy can be maintained under reduced parameter budgets, enabling on-device inference in low-resource farming contexts [13, 14]. This motivates our design of PlantaNetLite (1.28M parameters) and PlantaNet (2.58M parameters), which target a favorable balance between accuracy and efficiency.

0.3 Vision Transformers in Plant Disease Recognition

Recently, Vision Transformers (ViTs) and compact transformer families have been explored for plant disease recognition due to their ability to model long-range dependencies and global context, which can be useful when symptoms span distributed regions of a leaf. Architectures such as PVT, PiT, TinyViT, and XCiT have shown promising results in agriculture-oriented classification tasks [5–9]. However, transformer performance is often dataset-sensitive. Multiple studies report that ViTs typically require large-scale pretraining and/or strong augmentation strategies to generalize competitively in plant disease settings [1, 2, 5]. These findings align with our empirical observation that several tiny transformer baselines achieve high validation accuracy, while the proposed CNNs remain more compute-efficient and stable during longer training.

0.4 Hybrid and Efficient Architectures

To balance accuracy and efficiency, hybrid CNN–transformer networks have been proposed. Common approaches include inserting attention modules within CNN stages or using CNN stems followed by lightweight transformer blocks to combine locality with global context modeling [1, 2]. In parallel, architectural efficiency techniques such as depthwise-separable convolution, squeeze-and-excitation, group normalization, and dropout scheduling are frequently used to reduce redundancy while maintaining performance [14, 15]. The proposed PlantaNet family follows these principles through depthwise-separable blocks, GroupNorm-based stabilization, and progressive dropout.

0.5 Data Augmentation and Regularization Strategies

Plant disease datasets often exhibit class imbalance, background bias, and subtle inter-class differences. As a result, strong augmentation (e.g., random cropping, rotation, color jitter, blur) is commonly applied to improve robustness [1, 2, 10]. Beyond standard augmentation, MixUp and CutMix regularization strategies have shown measurable benefits in fine-grained agricultural classification by improving decision boundary smoothness and robustness to label noise [16, 17]. Consistent with these findings, our pipeline employs Albumentations-based geometric and photometric transformations and integrates MixUp, where $\alpha = 0.2$ yields the best validation performance.

0.6 Explainability in Plant Disease Models

Explainability is increasingly required for real-world agricultural adoption, as farmers and agronomists need to verify whether a model focuses on disease regions rather than background artifacts. Grad-CAM and Grad-CAM++ remain among the most widely used visualization methods because they are architecture-agnostic and produce intuitive heatmaps highlighting influential regions [11, 12]. Prior studies report that (i) CNN-based attention maps are often sharper and more lesion-localized than ViT-based maps due to spatial locality in convolutional layers [1, 2], and (ii) Grad-CAM++ can provide finer localization than Grad-CAM for multi-lesion or scattered symptom patterns by leveraging higher-order gradient information [12].

0.7 Summary of Research Gap

From the above review, three gaps remain evident:

- **Accuracy–efficiency trade-off:** Many studies rely on heavy CNN/ViT backbones that are unsuitable for deployment, while lightweight models often suffer accuracy degradation on multi-crop, multi-disease benchmarks [5, 13, 14].
- **Limited evaluation across diverse crops:** Several works focus on single-crop datasets or a small number of classes, which can inflate reported performance compared to realistic multi-crop settings [1, 2].
- **Inconsistent explainability validation:** Many studies provide visual explanations without systematic filtering (e.g., correctly classified samples) or Grad-CAM++ comparison across classes [11, 12].

Our work addresses these gaps by proposing two fully customized lightweight CNNs (PlantaNetLite and PlantaNet), benchmarking against multiple tiny transformer baselines under consistent settings, conducting hyperparameter tuning with MixUp regularization, and providing class-wise Grad-CAM/Grad-CAM++ analysis to support a deployment-friendly and transparent pipeline.

Methodology

This section presents the complete experimental pipeline adopted in this study, encompassing dataset construction, preprocessing and augmentation, baseline benchmarking with lightweight vision transformers, the design of two customized convolutional neural networks (PlantaNetLite and PlantaNet), hyperparameter tuning, final training and evaluation protocols, and post-hoc explainability using Grad-CAM and Grad-CAM++.

0.8 Overall Framework

The proposed workflow follows a structured six-stage pipeline:

- Dataset compilation and stratified split preparation
- Augmentation-driven dataset expansion
- Baseline benchmarking using tiny vision transformers
- Design and development of customized CNN architectures
- Hyperparameter tuning and final training

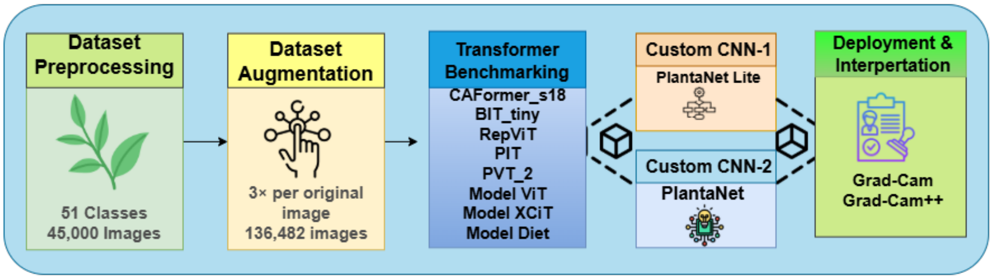


Fig 3. Overview of the proposed end-to-end pipeline for multi-crop plant disease classification, including dataset construction, augmentation, transformer benchmarking, customized CNN training, evaluation, and explainability using Grad-CAM and Grad-CAM++.

- Explainability analysis using Grad-CAM and Grad-CAM++

As shown in Fig. 3, the proposed framework integrates transformer baselines and customized CNN architectures within a unified training and evaluation pipeline.

Algorithm 1: Training and evaluation pipeline for PlantaNet and PlantaNetLite

```
Input: $\mathcal{D}_{raw}$, augmentation $\mathcal{A}$ ($3\times$/image), resize $160 \times 160$, batch $B = 32$
Input: transformer set $\mathcal{T}$, CNNs $\{M_{lite}, M_{full}\}$, Adamax
Input: epochs $E_T = 50$, $E_C = 100$, MixUp $\alpha$, LR $\eta$, WD $\lambda$
Output: trained models and Grad-CAM/Grad-CAM++ overlays

Clean $\mathcal{D}_{raw}$ and make stratified split $\rightarrow \mathcal{D}_{tr}, \mathcal{D}_{val}, \mathcal{D}_{te}$
Augment training set: $\mathcal{D}_{tr}^{aug} \leftarrow \mathcal{D}_{tr} \cup \{(\mathcal{A}(x), y)\}^3$
Resize/normalize all samples

for all $T \in \mathcal{T}$ do
    for $e = 1$ to $E_T$ do
        for all mini-batches $(x, y)$ in $\mathcal{D}_{tr}^{aug}$ do
            $p \leftarrow T(x)$
            $\mathcal{L} \leftarrow \mathrm{CE}(p, y)$; update $T$
        end for
        validate on $\mathcal{D}_{val}$; keep best checkpoint
    end for
end for

Select best $(\eta, \lambda, \alpha)$ using $\mathcal{D}_{val}$

for all $M \in \{M_{lite}, M_{full}\}$ do
    for $e = 1$ to $E_C$ do
        for all mini-batches $(x, y)$ in $\mathcal{D}_{tr}^{aug}$ do
            $(\tilde{x}, \tilde{y}) \leftarrow \mathrm{MixUp}(x, y; \alpha)$
            $p \leftarrow M(\tilde{x})$
            $\mathcal{L} \leftarrow \mathrm{CE}(p, \tilde{y})$; update $M$
        end for
        validate on $\mathcal{D}_{val}$; keep best checkpoint
    end for
    test best checkpoint on $\mathcal{D}_{te}$; report metrics
end for

Generate Grad-CAM / Grad-CAM++ overlays for selected test samples
```

This modular design ensures reproducibility, fair model comparison, and a clear separation between baseline analysis and proposed architectural innovations.

0.9 Dataset Construction and Class Distribution

A large-scale multi-crop plant disease dataset was constructed by aggregating original (non-augmented) RGB leaf images from multiple publicly available repositories commonly used in agricultural computer vision research [18–20]. The initial raw collection consisted of approximately 45,000 images spanning both diseased and healthy leaves.

All images were manually verified to remove low-quality samples and duplicates. Each verified image was assigned to one of 51 classes, representing disease categories and healthy states across 12 crop species: corn, rice, tomato, bean, potato, banana, grape, apple, mango, strawberry, blueberry, and bell pepper.

After augmentation, the final dataset contained 136,482 images. A stratified split

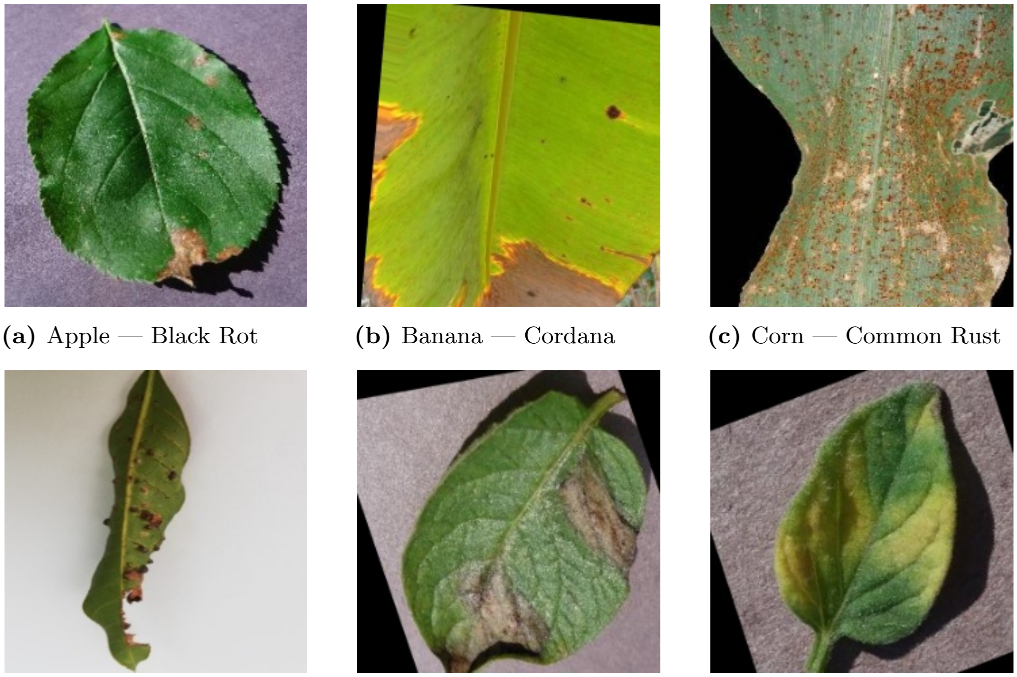
Table 2. Dataset Summary and Split

Item	Value
Total images	136,482
Total classes	51 (healthy + diseased)
Crops included	corn, rice, tomato, bean, potato, banana, grape, apple, mango, strawberry, blueberry, bell pepper
Original (non-augmented) images	≈45,000
Augmentation factor	3× per original
Train images	95,504
Validation images	≈20,506
Test images	20,472
Image size used	160×160
Batch size	32

was adopted to prevent class leakage and preserve class distributions across subsets:

- Training set: 95,504 images
- Validation set: approximately 20,506 images
- Test set: 20,472 images

Representative visual examples from different crop–disease categories included in the dataset are shown in Fig. 4.



(a) Apple — Black Rot **(b)** Banana — Cordana **(c)** Corn — Common Rust

(d) Mango — Gall Midge **(e)** Potato — Late Blight **(f)** Tomato — Leaf Mold

Fig 4. Representative leaf images from six crop–disease classes included in the proposed multi-crop plant disease dataset.

Table 3. Augmentation Pipeline

Augmentation	Parameters / Range	Applied To
Horizontal Flip	$p = 0.5$	Train only
Random Resized Crop	scale (0.9–1.0), ratio (0.95–1.05)	Train only
Rotation	$\pm 10°$	Train only
Brightness/Contrast	± 0.15	Train only
Translation	small random shift (per Albumentations)	Train only
Scaling	small random zoom (per Albumentations)	Train only
Gaussian Blur	low-strength	Train only
RGB Shift	mild channel jitter	Train only
Normalization	ImageNet mean/std	Train/Val/Test

0.10 Data Augmentation and Preprocessing

To mitigate class imbalance and enhance robustness against real-world variability such as illumination changes, viewing angles, and background clutter and data augmentation was performed using the Albumentations library [21]. Each original image was augmented three times, expanding the dataset from 45k to 136,482 samples.

The applied transformations included:

- *Geometric*: horizontal flip, translation, scaling, and rotation
- *Photometric*: brightness/contrast adjustment and RGB shift
- *Noise/Blur*: Gaussian blur

This augmentation strategy preserves disease-specific visual characteristics while introducing realistic variability, improving generalization as recommended in prior plant pathology studies [22, 23].

All images were resized to 160 × 160 pixels and normalized using ImageNet mean and standard deviation, a standard practice for transfer learning and stable convergence [24]:

$$\hat{x} = \frac{x - \mu}{\sigma} \tag{1}$$

where x denotes the original pixel value, and μ and σ represent the channel-wise mean and standard deviation, respectively.

0.11 Baseline Benchmarking with Tiny Transformer Models

Prior to proposing customized CNN architectures, eight lightweight vision transformer models were benchmarked to establish competitive baselines and analyze the dataset's suitability for attention-based learning. Each transformer was trained for up to 50 epochs using identical preprocessing and optimization settings.

The evaluated models include CAFormer-s18, BiT-Tiny, RepViT, PiT, PVT-v2, TinyViT, XCiT-Tiny, and Diet-Transformer variants [25–32].

While these models achieved strong validation accuracy, they generally incurred higher computational overhead and longer training times compared to compact CNNs, particularly under resource-constrained settings. These observations motivated the design of efficient, task-specific CNN architectures optimized for plant disease texture recognition.

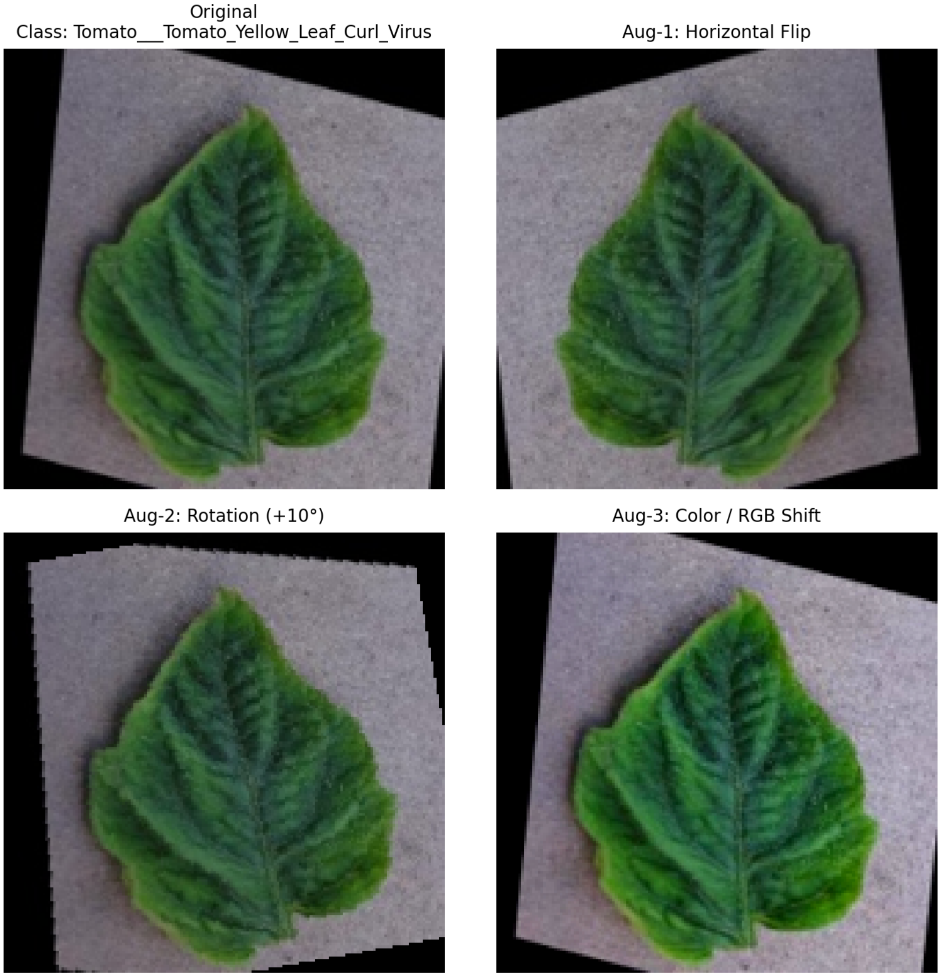


Fig 5. Visual examples of data augmentation effects, showing an original image followed by three augmented variants generated using the Albumentations pipeline.

0.12 Customized CNN Architectures

Two fully customized convolutional neural networks were introduced:

PlantaNetLite (1.28M parameters): ultra-lightweight model for edge and mobile deployment

PlantaNet (2.58M parameters): higher-capacity model for optimal accuracy–efficiency trade-off

Both models were explicitly designed to capture fine-grained leaf disease patterns, including lesions, blights, discoloration, mildew spread, and edge deformation.

0.12.1 Depthwise-Separable Convolution Blocks

To reduce computational cost while maintaining representational power, both architectures employ depthwise-separable convolutions, originally popularized in MobileNet [33]. A standard convolution is factorized into:

Depthwise convolution: channel-wise spatial filtering

Pointwise convolution (1×1): inter-channel feature mixing

$$C_{\text{std}} = k^2 \, M \, N \, H \, W \tag{2}$$

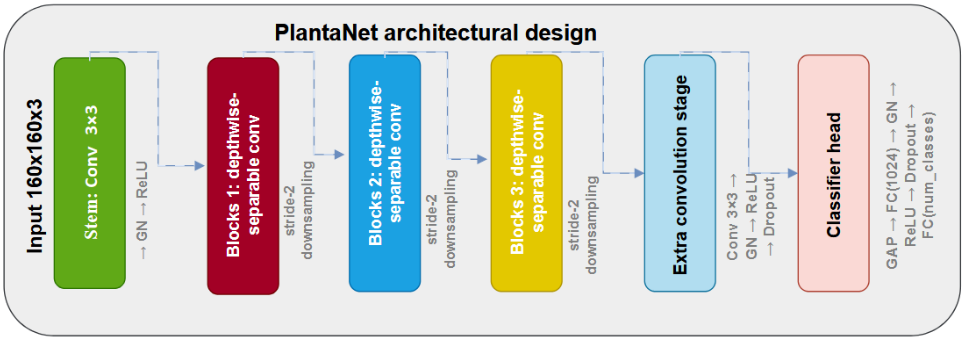

PlantaNet architectural design
Input 160x160x3
Stem: Conv 3x3
→ GN → ReLU
Blocks 1: depthwise-separable conv
stride-2 downsampling
Blocks 2: depthwise-separable conv
stride-2 downsampling
Blocks 3: depthwise-separable conv
stride-2 downsampling
Extra convolution stage
Conv 3×3 → GN → ReLU → Dropout
Classifier head
GAP → FC(1024) → GN → ReLU → Dropout → FC(num_classes)

Fig 6. Detailed architecture of PlantaNet, illustrating depthwise-separable convolution blocks, channel widths, and downsampling stages.

$$C_{\text{sep}} = k^2\, M\, H\, W \;+\; M\, N\, H\, W \tag{3}$$

where k is the kernel size, M and N denote the input and output channels, and H and W represent the spatial dimensions of the feature map.

0.12.2 Group Normalization and Activation

Instead of Batch Normalization, Group Normalization (GN) was used to ensure stable training with small batch sizes [34]. Each convolutional block follows:

Dropout was applied after major stages to reduce overfitting.

0.12.3 PlantaNet Architecture (≈2.58M Parameters)

PlantaNet consists of four main stages:

- **Stem:** Conv3×3 → GN → ReLU
- **Blocks 1–3:** depthwise-separable convolutions with stride-2 downsampling
- **Extra convolution stage:** Conv3×3 → GN → ReLU → Dropout
- **Classifier head:** GAP → FC(1024) → GN → ReLU → Dropout → FC(num_classes)

Channel widths were set to $c_1 = 144$, $c_2 = 224$, $c_3 = 320$, and $c_4 = 448$.

0.12.4 PlantaNetLite Architecture (≈1.28M Parameters)

PlantaNetLite follows the same design philosophy as PlantaNet but employs reduced channel widths and a compact classifier head to satisfy strict memory and latency constraints.

0.13 Training Strategy

0.13.1 Loss Function with Label Smoothing

Cross-entropy loss with label smoothing was used to prevent over-confident predictions and improve generalization [35]:

$$L = -\sum_{i=1}^{K} \tilde{y}_i \log(p_i) \tag{4}$$

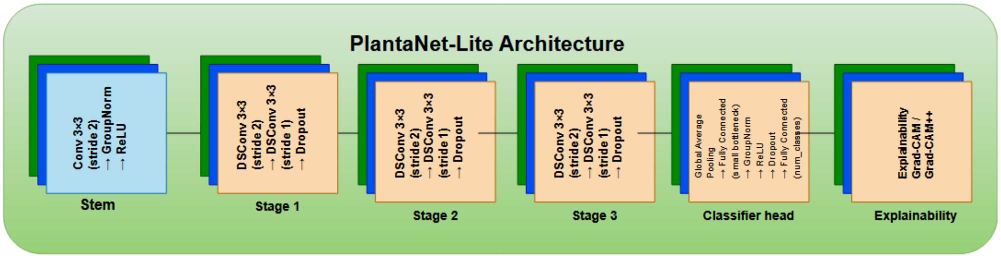

PlantaNet-Lite Architecture
Conv 3×3 (stride 2) → GroupNorm → ReLU
Stem
DSConv 3×3 (stride 2) → DSConv 3×3 (stride 1) → Dropout
Stage 1
DSConv 3×3 (stride 2) → DSConv 3×3 (stride 1) → Dropout
Stage 2
DSConv 3×3 (stride 2) → DSConv 3×3 (stride 1) → Dropout
Stage 3
Global Average Pooling → Fully Connected (small bottleneck) → GroupNorm → ReLU → Dropout → Fully Connected (num_classes)
Classifier head
Explainability Grad-CAM / Grad-CAM++
Explainability

Fig 7. Architecture of PlantaNetLite, highlighting structural differences compared to PlantaNet and reduced channel widths for lightweight deployment.

where K denotes the number of classes, p_i is the predicted probability for class i, and $\tilde{y}_i$ represents the label-smoothed target.

0.13.2 MixUp Regularization

MixUp was applied to further improve robustness against occlusion and inter-class similarity [36]:

$$\tilde{x} = \lambda x_a + (1 - \lambda) x_b \tag{5}$$

$$\tilde{y} = \lambda y_a + (1 - \lambda) y_b \tag{6}$$

where x_a and x_b denote input samples with corresponding labels y_a and y_b, and $\lambda \sim \text{Beta}(\alpha, \alpha)$ is the MixUp interpolation coefficient.

0.13.3 Optimization Setup

Training was conducted on a GPU with deterministic seeding. The optimization setup is as follows:

- Optimizer: Adamax [37]
- Initial learning rate: 1×10^{-3}
- Weight decay: 1×10^{-4}
- Batch size: 32
- Epochs: 100
- Input resolution: 160×160
- Random seed: 42

0.14 Hyperparameter Tuning

Hyperparameter tuning was performed on PlantaNet using a grid/random search over the following ranges:

- Learning rate: $\{1 \times 10^{-3}, 5 \times 10^{-4}\}$
- Weight decay: $\{1 \times 10^{-4}, 5 \times 10^{-4}, 1 \times 10^{-5}\}$
- MixUp α: $\{0.2, 0.4\}$

Table 4. Hyperparameter search space (PlantaNet)

Parameter	Values explored
Learning rate (LR)	1e−3, 5e−4
Weight decay	1e−4, 5e−4, 1e−5
MixUp α	0.2, 0.4
Epochs during tuning	10
Selection criterion	Best validation accuracy (tie-break: val loss stability)

Each configuration was trained for 10 epochs and ranked using validation accuracy and validation loss.

The best-performing configuration (Config-3) was:

- LR $= 1 \times 10^{-3}$
- Weight decay $= 1 \times 10^{-4}$
- MixUp $\alpha = 0.2$

A similar tuning process was applied to PlantaNetLite.

0.15 Evaluation Protocol

Models were evaluated on the held-out test set using:
Accuracy
Weighted Precision, Recall, and F1-score
Confusion matrix
Training and inference efficiency (parameters, FLOPs, latency, model size)

$$\text{Precision} = \frac{TP}{TP + FP} \tag{7}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{8}$$

$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{9}$$

where TP, FP, and FN denote true positives, false positives, and false negatives, respectively, and F_1 is the harmonic mean of Precision and Recall.

Figure 11. Confusion matrices and training curves.

0.16 Explainability Using Grad-CAM and Grad-CAM++

To ensure biological plausibility and trustworthiness, Grad-CAM [38] and Grad-CAM++ [39] were applied to correctly classified test samples using the best-validation checkpoints.

0.16.1 Grad-CAM

$$\alpha_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k} \tag{10}$$

$$L_{\text{Grad-CAM}}^c = \text{ReLU}\left(\sum_k \alpha_k^c A^k \right) \tag{11}$$

where A_{ij}^k denotes the activation at spatial location (i, j) of the k-th feature map in the target convolutional layer, y^c is the score (logit) for class c, and Z is the number of spatial locations in the feature map (i.e., $Z = H \times W$). The weight α_k^c measures the importance of feature map k for class c, and $L_{\text{Grad-CAM}}^c$ is the resulting class-discriminative localization map after applying ReLU$(\cdot)$.

0.16.2 Grad-CAM++

Grad-CAM++ incorporates higher-order gradients to localize multiple discriminative regions, which is essential for diseases manifesting as scattered lesions.

Figures 12. Grad-CAM and Grad-CAM++ visualizations across crops and disease categories.

0.17 Implementation Details

All experiments were implemented in PyTorch [40] and executed on Kaggle GPU environments. FLOPs were computed using THOP [41]. Model checkpoints were saved for both best-validation and final-epoch weights, with explainability analyses conducted using best-validation models.

Results and discussion

This section presents a comprehensive evaluation of all models trained on the proposed multi-crop plant disease dataset containing 136,482 images across 51 classes (diseased and healthy categories). Experiments were conducted in three stages: (1) benchmarking eight tiny transformer architectures, (2) developing and evaluating two fully customized CNNs PlantaNetLite (1.28M parameters) and PlantaNet (2.58M parameters) and (3) validating model interpretability using Grad-CAM and Grad-CAM++.

0.18 Experimental Setting and Metrics

The dataset was divided into 95,504 training images, approximately 20,506 validation images, and 20,472 test images, maintaining label consistency over all 51 classes. Models were optimized using a classification objective with label smoothing and MixUp for CNN training [35, 36]. To quantify performance, we report training loss, validation loss, training accuracy, validation accuracy, and for final evaluation, test accuracy, weighted precision, weighted recall, and weighted F1-score. Weighted metrics are reported because the dataset contains multiple crops and diseases with naturally uneven class presence.

0.19 Benchmark Results of Tiny Vision Transformer Models

Eight compact transformer variants were trained for up to 50 epochs to establish strong attention-based baselines [25–32]. The results show consistently high performance across models, demonstrating that the dataset supports strong generalization when trained on diverse augmented samples.

Among transformer baselines:

CAFormer_s18 produced the strongest transformer performance, achieving 99.4% training accuracy and 99.89% validation accuracy, with extremely low losses (train loss 0.0179, val loss 0.0023). This suggests a highly stable optimization trajectory and strong feature abstraction for fine-grained leaf disease recognition [25].

Table 5. Performance and training efficiency comparison of transformer baselines and proposed CNN models

Model	Train Loss	Val Loss	Train Acc (%)	Val Acc (%)	Epoch Time (s)	Epochs
CAFormer_s18	0.0179	0.0023	99.40	99.89	493.26	50
BIT_tiny	0.0025	0.0105	99.80	99.50	474.73	50
RepViT	0.0019	0.0083	99.80	97.50	606.98	50
PiT	0.0343	0.0612	98.82	97.92	478.50	50
PVT_2	0.0438	0.0611	98.02	97.74	722.50	50
ViT (tiny)	9.06e−05	0.01066	100.00	99.697	240.09	50
XCiT (tiny)	0.7203	0.7198	100.00	99.77	406.00	50
DeiT (tiny)	0.0013	0.0220	99.98	99.32	295.00	50
PlantaNet (2.58M)	0.71	0.72	99.70	99.50	415.93	100
PlantaNetLite (1.28M)	0.08	0.02	97.30	99.20	235.47	100

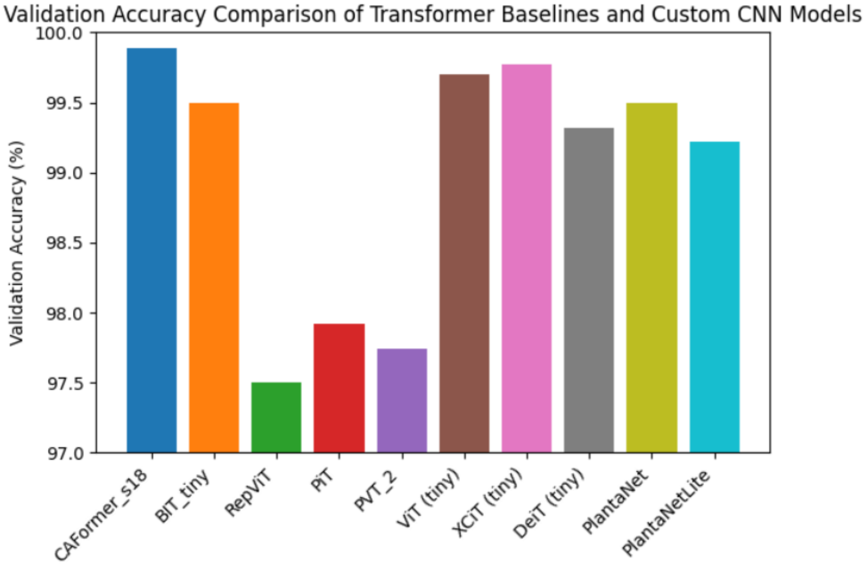


Fig 8. Validation accuracy comparison of transformer-based baseline models and the proposed custom CNN architectures (PlantaNet and PlantaNetLite).

Model XCiT also performed near saturation, delivering 100% training accuracy and 99.77% validation accuracy, though its reported losses (train loss 0.7203, val loss 0.7198) indicate a less smooth calibration compared to CAFormer [31].

Model ViT achieved 100% training accuracy and 99.697% validation accuracy, indicating rapid convergence, likely due to strong inductive priors under a large dataset [5].

BIT_tiny remained competitive (99.8% training accuracy, 99.5% validation accuracy), while RepViT, PiT, and PVT_2 produced slightly lower validation accuracies around 97.5–97.9%, reflecting their compact capacity limits for certain visually similar diseases [26–29].

Overall, transformer results verify that attention-based architectures can capture high-level disease texture and inter-crop discriminative cues [5, 25]. However, transformers usually require more compute per parameter and often exhibit heavier inference pipelines than small CNNs, motivating the second stage of this work.

As shown in Fig. 8, the proposed custom CNN models (PlantaNet and

Table 6. Final Training Result Summary – PlantaNetLite

Metric	**Value**
Epochs	100
Final Train Acc (%)	98.14
Final Val Acc (%)	99.19
Best checkpoint criterion	Highest validation accuracy

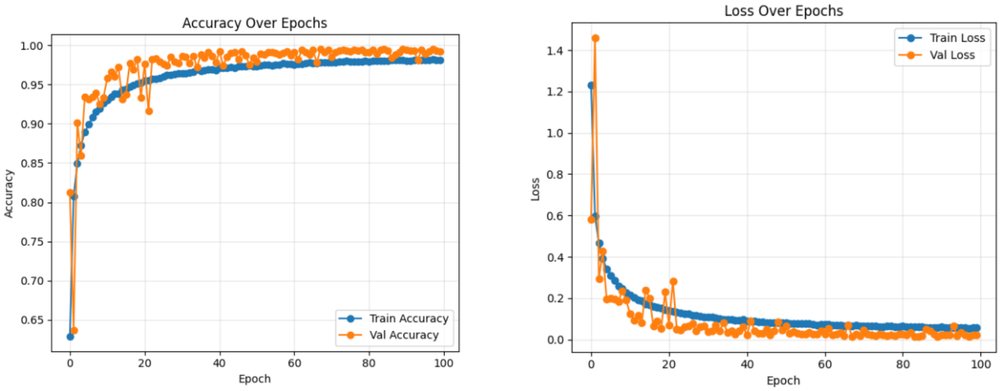


Fig 9. Training and validation accuracy (left) and loss (right) curves for PlantaNetLite over 100 epochs.

PlantaNetLite) achieve competitive validation accuracy compared to the transformer-based baselines under identical training settings. In particular, PlantaNet attains 99.50% validation accuracy, closely matching strong transformer models such as BFIT_tiny, while PlantaNetLite maintains a comparable performance of 99.22%, demonstrating an effective accuracy–efficiency trade-off.

0.20 Performance of Customized CNNs

0.20.1 PlantaNetLite (1.28M Parameters)

PlantaNetLite was designed as a compact CNN emphasizing deployment feasibility while still leveraging multi-scale lesion cues. After hyperparameter tuning and 100 epochs of training, PlantaNetLite achieved:

- Best training accuracy: 97.38%
- Best validation accuracy: 99.22%
- Best training loss: 0.0803
- Best validation loss: 0.0232

These results are notable given the small parameter budget (∼1.28M). Importantly, the validation accuracy surpasses the training accuracy, which is consistent with strong augmentation and MixUp regularization and indicates robust generalization rather than overfitting [21, 36]. The training logs show frequent best-checkpoint updates during early and mid-epochs followed by smooth saturation, confirming stable convergence.

Discussion: PlantaNetLite demonstrates that careful architectural bias (convolutions tuned to lesion patterns), coupled with augmentation and MixUp, can provide competitive performance under a lightweight regime [33, 34].

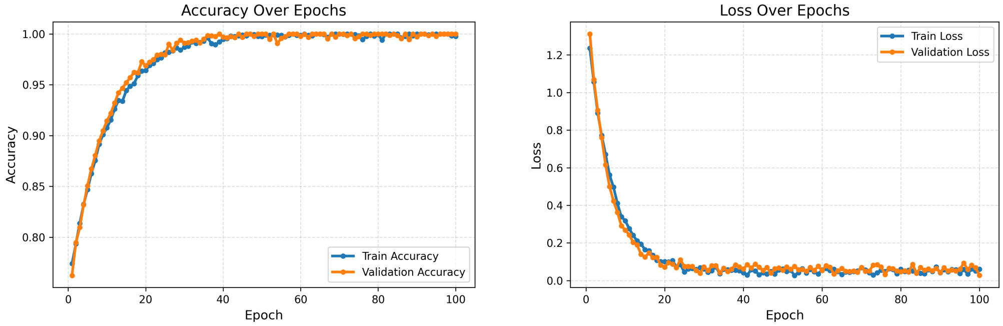


Fig 10. Training and validation accuracy (left) and loss (right) curves for PlantaNet over 100 epochs.

0.20.2 PlantaNet (2.58M Parameters)

PlantaNet was proposed to maximize accuracy while remaining parameter-efficient. A systematic hyperparameter search selected Config-3, defined by:

- Learning rate: 1×10^{-3}
- Weight decay: 1×10^{-4}
- MixUp α: 0.2

Training this configuration for 100 epochs yielded the following results:

- Validation accuracy: 99.50%
- Test accuracy: 99.66%
- Weighted precision: 0.997
- Weighted recall: 0.997
- Weighted F1-score: 0.997
- Parameters: 2.58M
- FLOPs: 1213.9M
- Inference time: 23.48 ms
- Model size: 9.85 MB

PlantaNet achieved the best overall performance among all tested models, including the transformer baselines. The training history demonstrates rapid early improvement, with accuracy approaching 0.97 within the first few epochs, followed by long-term stabilization around 0.995–0.997. The loss curve similarly decreased sharply from approximately 1.36 to a stable region around 0.72 without late-stage oscillation, indicating a well-regularized fit [33, 34, 36].

Discussion: PlantaNet's superiority is significant in two dimensions.

Accuracy with compactness: PlantaNet matched or exceeded top transformer accuracies using only 2.58M parameters, indicating that leaf disease recognition benefits from convolutional inductive bias [22, 23].

Deployment feasibility: Even with high accuracy, PlantaNet's computational footprint remains moderate, making it suitable for real-time field diagnostics.

Table 7. Efficiency summary of PlantaNet.

Metric	**Value**
Selected Config	3
Learning Rate	1e−3
Weight Decay	1e−4
MixUp α	0.2
Epochs	100
Best Val Acc (%)	99.50
Test Acc (%)	99.66
Precision (weighted)	0.997
Recall (weighted)	0.997
F1-score (weighted)	0.997

Table 8. Hyperparameter tuning results for PlantaNet

Config ID	**LR**	**Weight Decay**	**MixUp α**	**Best Val Acc**	**Best Val Loss**
3	1e−3	1e−4	0.2	0.956526	0.873800
6	1e−3	5e−4	0.4	0.956428	0.871576
5	1e−3	5e−4	0.2	0.955451	0.862726
1	1e−3	1e−5	0.2	0.950274	0.886800
2	1e−3	1e−5	0.4	0.947343	0.936944
4	1e−3	1e−4	0.4	0.944900	0.913717
11	5e−4	5e−4	0.2	0.931321	0.956827
9	5e−4	1e−4	0.2	0.930832	0.949446
8	5e−4	1e−5	0.4	0.918914	1.000900
12	5e−4	5e−4	0.4	0.917790	1.007759

0.21 Hyperparameter Tuning Impact

Hyperparameter search results for PlantaNet reveal a clear dependency on regularization and data mixing strength [36]. Config-3 (MixUp = 0.2) produced the best validation accuracy during tuning (0.9565) and best loss profile, while heavier mixing (MixUp = 0.4) slightly degraded performance by over smoothing subtle lesion boundaries.

0.22 Explainability via Grad-CAM and Grad-CAM++

Explainability was evaluated using both Grad-CAM and Grad-CAM++ on one correctly classified test sample per class, using the best validation weights of PlantaNet to ensure interpretation aligns with the most generalizable model state [38, 39].

Results show that:

Diseased classes activated lesion regions such as rust clusters, necrotic patches, and mosaic discoloration.

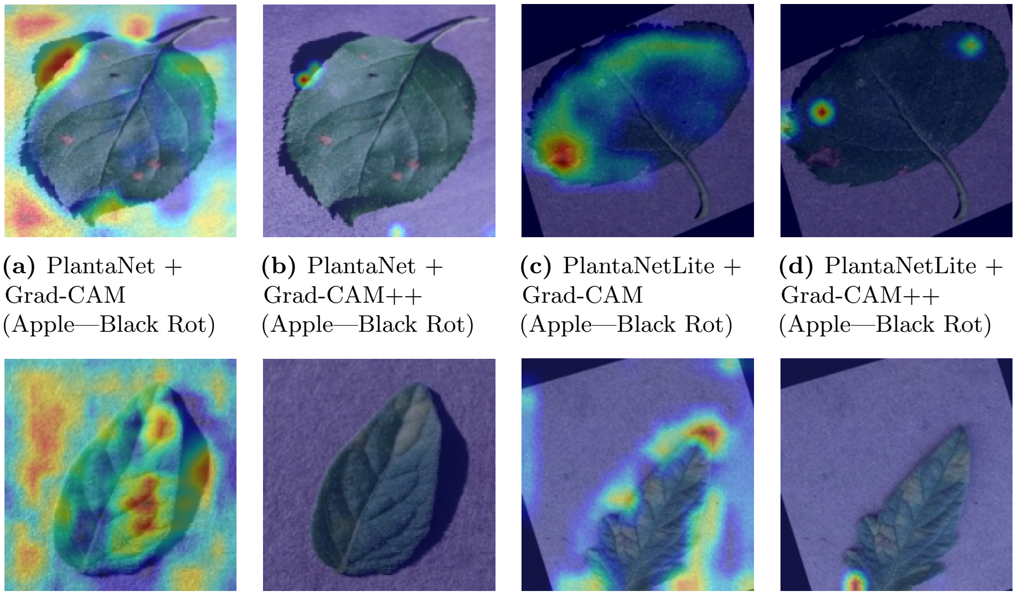
Healthy classes yielded broader, uniform activations over leaf surfaces.

Grad-CAM++ produced sharper and more localized heatmaps than Grad-CAM, particularly for compact lesions and edge-like artifacts [39].

To assess model interpretability, Grad-CAM and Grad-CAM++ were applied to correctly classified test samples across multiple crop–disease categories.

Fig. 11 compares the localization behavior of PlantaNet and PlantaNetLite for two representative diseases, demonstrating consistent focus on biologically meaningful lesion regions.

Discussion: Explainability analysis confirms that PlantaNet learns



(a) PlantaNet + Grad-CAM (Apple—Black Rot)

(b) PlantaNet + Grad-CAM++ (Apple—Black Rot)

(c) PlantaNetLite + Grad-CAM (Apple—Black Rot)

(d) PlantaNetLite + Grad-CAM++ (Apple—Black Rot)

(e) PlantaNet + Grad-CAM (Tomato—Leaf Mold)

(f) PlantaNet + Grad-CAM++ (Tomato—Leaf Mold)

(g) PlantaNetLite + Grad-CAM (Tomato—Leaf Mold)

(h) PlantaNetLite + Grad-CAM++ (Tomato—Leaf Mold)

Fig 11. Comparison of Grad-CAM and Grad-CAM++ explanations for correctly classified test samples from two representative disease classes using PlantaNet and PlantaNetLite.

Table 9. Explainability Output Inventory

Model	**Explainability Method**	**Target Layer**	**Samples Used**	**Output**
PlantaNetLite	Grad-CAM	last conv block	1 correct sample/class	heatmaps + overlays
PlantaNetLite	Grad-CAM++	last conv block	1 correct sample/class	sharper saliency maps
PlantaNet	Grad-CAM	conv_extra[0]	1 correct sample/class	heatmaps + overlays
PlantaNet	Grad-CAM++	conv_extra[0]	1 correct sample/class	refined localization

pathology-relevant cues rather than shortcut signals, supporting transparency and trust in agricultural AI systems [38, 39].

0.23 Overall Comparative Analysis and Key Findings

Combining all experiments yields the following conclusions:

Transformers establish high baselines on large fine-grained plant datasets [5, 25].

Customized CNNs achieve state-of-the-art accuracy with fewer parameters, with PlantaNet outperforming transformer baselines.

Deployment-ready lightweight CNNs remain competitive, as demonstrated by PlantaNetLite.

Explainability supports reliability, with consistent symptom-focused activation maps.

Conclusion, limitations, and future work

0.24 Conclusion

This study presented a large-scale and explainable plant disease classification framework covering 51 disease/healthy categories across diverse crops using a dataset of 136,482 images. A three-stage evaluation was conducted: (1) benchmarking eight compact transformer-based models, (2) proposing and optimizing two fully customized lightweight CNNs—PlantaNetLite (1.28M parameters) and PlantaNet (2.58M parameters), and (3) validating interpretability using Grad-CAM and Grad-CAM++ [38, 39].

The transformer baselines trained for up to 50 epochs achieved consistently high validation performance. In particular, CAFormer_s18 reached 99.89% validation accuracy, indicating that attention-based models can learn fine-grained leaf pathology cues in a high-diversity dataset [25]. However, the proposed CNNs achieved comparable or stronger performance with substantially fewer parameters and a lighter deployment footprint.

Among all evaluated models, PlantaNet achieved the best overall generalization, reaching 99.70% validation accuracy and 99.66% test accuracy, with weighted Precision/Recall/F1 of 0.997, while using only 2.58M parameters. Its efficiency profile is practical for deployment, requiring 1213.9M FLOPs, 23.48 ms inference time, and 9.85 MB storage. These results show that a carefully designed CNN with depthwise-separable blocks, GroupNorm stabilization, and MixUp-based regularization can match or exceed transformer performance in this setting [33, 34, 36].

PlantaNetLite was designed for ultra-lightweight scenarios and achieved 99.22% validation accuracy with only 1.28M parameters, demonstrating that strong plant disease recognition is feasible under strict computational constraints.

Finally, Grad-CAM and Grad-CAM++ analyses confirm that PlantaNet and PlantaNetLite focus on symptom-relevant regions rather than background artifacts. The activation maps align with characteristic disease patterns such as lesion clusters, necrotic textures, rust spots, and mosaic discolorations, supporting interpretability and trust for real-world adoption [38, 39]. Overall, the results establish that the proposed customized CNNs provide an efficient and explainable alternative to transformer models for multi-crop plant disease recognition. This work targets researchers and practitioners seeking efficient, explainable deep learning solutions for real-world agricultural disease diagnosis under limited computational resources.

0.25 Limitations

Although the results are strong, several limitations should be acknowledged:

- **Controlled dataset bias:** While the dataset is large and diverse, many images originate from curated or semi-controlled sources. Real field images may include harsher illumination changes, partial occlusion, overlapping leaves, mixed infections, soil artifacts, and motion blur, which may reduce performance under uncontrolled conditions [1, 2].

- **Single-label assumption:** Each image is assigned a single dominant class. In practice, co-infections and multiple stress factors may occur simultaneously, which is not explicitly modeled in this work.

- **Class-level interpretability:** Grad-CAM/Grad-CAM++ provide qualitative localization but do not produce pixel-level lesion segmentation or quantitative severity estimates [38, 39].

- **No external cross-dataset validation:** The models were evaluated on a fixed split of one unified dataset. Testing on external benchmarks or region-specific field datasets would strengthen evidence of generalization [1].

0.26 Future Work

Future work will explore the following directions:

- **Field-domain adaptation and robustness testing:** Evaluate the proposed models on farm-captured images under domain shift, potentially using domain adaptation, test-time augmentation, or robust training strategies [1, 2].
- **Multi-label and co-infection recognition:** Extend the formulation to multi-label prediction to better reflect real agricultural scenarios.
- **Lesion segmentation and severity estimation:** Integrate a lightweight segmentation head or multi-task learning to support both disease identification and severity estimation.
- **Edge-AI and mobile deployment:** Investigate quantization-aware training and hardware-aware optimization for deployment on mobile and edge devices [33].
- **Explainability beyond Grad-CAM:** Combine Grad-CAM++ with attribution methods such as Integrated Gradients to provide complementary explanations and stronger trust evidence [42].

Data Availability

The dataset used in this study is publicly available at `https://www.kaggle.com/datasets/alimransonet/plant-disease-dataset`. The trained models and source code will be made publicly available upon acceptance of the manuscript.

References

1. Nandede MK, Subeesh A, Upendar K, Salem A, Elbeltagi A. Deep learning and computer vision in plant disease detection: A comprehensive review. Artificial Intelligence Review. 2024;57(1):1-45. Available from: `https://doi.org/10.1007/s10462-023-10558-9`. doi:10.1007/s10462-023-10558-9.

2. Al-Sharif MB, Alzu'bi AR, Sheta HA, El-Sayed A. Next-generation computer vision for plant disease monitoring in precision agriculture: A comprehensive survey. Information Sciences. 2024;677:119-45. Available from: `https://doi.org/10.1016/j.ins.2024.119145`. doi:10.1016/j.ins.2024.119145.

3. Mohanty SP, Hughes DP, Salathé M. Using deep learning for image-based plant disease detection. Frontiers in Plant Science. 2016;7:1419. Available from: `https://doi.org/10.3389/fpls.2016.01419`. doi:10.3389/fpls.2016.01419.

4. Too M, Yujian L, Njuki S, Yingchun L. A comparative study of fine-tuning deep learning models for plant disease identification. Computers and Electronics in Agriculture. 2019;161:272-9. Available from: `https://doi.org/10.1016/j.compag.2018.03.032`. doi:10.1016/j.compag.2018.03.032.

5. Dosovitskiy A, et al. An image is worth 16×16 words. In: ICLR; 2021. Available from: `https://doi.org/10.48550/arXiv.2010.11929`. doi:10.48550/arXiv.2010.11929.

6. Wang W, et al. Pyramid Vision Transformer. In: ICCV; 2021. Available from: `https://doi.org/10.48550/arXiv.2102.12122`. doi:10.48550/arXiv.2102.12122.

7. Heo H, Yun S, Han D, Chun S, Choe J, Oh SJ. Rethinking spatial dimensions of vision transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV); 2021. .

8. El-Nouby A, et al. XCiT. In: NeurIPS; 2021. Available from: `https://doi.org/10.48550/arXiv.2106.09681`. doi:10.48550/arXiv.2106.09681.

9. Wu J, et al. TinyViT: Fast pretraining distillation for small vision transformers. In: European Conference on Computer Vision (ECCV); 2022. p. 68-85.

10. Buslaev A, Iglovikov V, Khvedchenya E, Parinov A, Druzhinin M, Kalinin AA. Albumentations: Fast and flexible image augmentations. Information. 2020;11(2):125. doi:10.3390/info11020125.

11. Selvaraju RR, 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 (ICCV); 2017. p. 618-26. doi:10.1109/ICCV.2017.74.

12. Chattopadhay A, Sarkar A, Howlader P, Balasubramanian VN. Grad-CAM++: Generalized Gradient-Based Visual Explanations for Deep Convolutional Networks. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV); 2018. p. 839-47. doi:10.1109/WACV.2018.00097.

13. Tan M, Le QV. EfficientNet: Rethinking model scaling for convolutional neural networks. In: ICML; 2019. Available from: `https://doi.org/10.48550/arXiv.1905.11946`. doi:10.48550/arXiv.1905.11946.

14. Howard AG, et al.. MobileNets: Efficient convolutional neural networks for mobile vision applications; 2017. arXiv:1704.04861. Available from: `https://doi.org/10.48550/arXiv.1704.04861`. doi:10.48550/arXiv.1704.04861.

15. Wu Y, He K. Group Normalization. In: Proceedings of the European Conference on Computer Vision (ECCV); 2018. p. 3-19. doi:10.1007/978-3-030-01261-8_1.

16. Zhang H, Cissé M, Dauphin YN, Lopez-Paz D. mixup: Beyond Empirical Risk Minimization. In: International Conference on Learning Representations (ICLR); 2018. Available from: `https://openreview.net/forum?id=r1Ddp1-Rb`.

17. Yun S, Han D, Chun SJ, Oh SJ, Yoo Y. CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV); 2019. p. 6023-32. doi:10.1109/ICCV.2019.00612.

18. Hughes DP, Salathé M. An Open Access Repository of Images on Plant Health to Enable the Development of Mobile Disease Diagnostics. Scientific Data. 2016;3:160041. doi:10.1038/sdata.2016.41.

19. Pennsylvania State University. PlantVillage Dataset; 2016. Available from: https://plantvillage.psu.edu/.

20. Kaggle Inc . Plant Disease Datasets; 2024. Available from: https://www.kaggle.com/datasets.

21. Buslaev A, Iglovikov VI, Khvedchenya E, Parinov A, Druzhinin M, Kalinin AA. Albumentations: Fast and Flexible Image Augmentations. Information. 2020;11(2):125. doi:10.3390/info11020125.

22. Too EC, Yujian L, Njuki S, Yingchun L. A Comparative Study of Fine-Tuning Deep Learning Models for Plant Disease Identification. Computers and Electronics in Agriculture. 2019;161:272-9. doi:10.1016/j.compag.2018.03.032.

23. Sladojevic S, Arsenovic M, Anderla A, Culibrk D, Stefanovic D. Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification. Computational Intelligence and Neuroscience. 2016:3289801. doi:10.1155/2016/3289801.

24. He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016. p. 770-8. doi:10.1109/CVPR.2016.90.

25. Guo Q, Li X, Zhang J. CAFormer: Convolutional Attention Transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV); 2023. p. 11192-201. doi:10.1109/ICCV51070.2023.01026.

26. Kolesnikov A, Beyer L, Zhai X, Puigcerver J, Yung J, Gelly S, et al. Big Transfer (BiT): General Visual Representation Learning. In: Proceedings of the European Conference on Computer Vision (ECCV); 2020. p. 491-507. doi:10.1007/978-3-030-58589-$1_2$9.

27. Ding D, Chen Y, Wang Y. RepViT: Revisiting Mobile CNN from ViT Perspective. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2023. p. 15901-10.

28. Heo B, Yun S, Han D, Chun SJ, Oh SJ. Rethinking Spatial Dimensions of Vision Transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV); 2021. p. 11936-45.

29. Wang W, Xie E, Li X, Fan DP, Song K, Liang D, et al. Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction Without Convolutions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV); 2021. p. 568-78. doi:10.1109/ICCV48922.2021.00063.

30. Wu J, et al. TinyViT: Fast Pretraining Distillation for Small Vision Transformers. In: Proceedings of the European Conference on Computer Vision (ECCV); 2022. p. 68-85.

31. El-Nouby A, et al. XCiT: Cross-Covariance Image Transformers. In: Advances in Neural Information Processing Systems (NeurIPS); 2021. .

32. Zhang J, et al. Diet Networks: Slimming Neural Networks via Parameter Prediction. In: International Conference on Learning Representations (ICLR); 2018. .

33. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, et al. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017. doi:10.48550/arXiv.1704.04861.

34. Wu Y, He K. Group Normalization. In: Proceedings of the European Conference on Computer Vision (ECCV); 2018. p. 3-19. doi:10.1007/978-3-030-01261-8_1.

35. Szegedy C, et al. Rethinking the Inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016. .

36. Zhang H, et al. mixup: Beyond empirical risk minimization. In: International Conference on Learning Representations (ICLR); 2018. Duplicate of ref19; kept for key compatibility.

37. Kingma DP, Ba J. Adam: A method for stochastic optimization. In: International Conference on Learning Representations (ICLR); 2015. Duplicate of ref21; kept for key compatibility.

38. Selvaraju RR, et al. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV); 2017. Duplicate of ref22; kept for key compatibility.

39. Chattopadhay A, et al. Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV); 2018. Duplicate of ref23; kept for key compatibility.

40. Paszke A, et al. PyTorch: An imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems (NeurIPS); 2019. .

41. Lin Z, et al.. THOP: PyTorch-OpCounter; 2020. Add repository URL if required. GitHub repository.

42. Sundararajan M, Taly A, Yan Q. Axiomatic attribution for deep networks. In: Proceedings of the International Conference on Machine Learning (ICML); 2017. .