2025 28th International Conference on Computer and Information Technology (ICCIT)
19-21 December 2025, Cox's Bazar, Bangladesh

An Explainable AI based Plant Disease Identification using a Two-Stage Detection-Classification Pipeline with YOLO and ECA-NFNet Framework

Tahir Hasan
Dept. of Computer Science and Electronic Engineering
University of Surrey
Guildford, United Kingdom
tahir.hasan.thk@gmail.com

Md. Fatin Ilham
Dept. of Computer Science and Engineering
Varendra University
Rajshahi, Bangladesh
mdfatin.ilham@gmail.com

Md. Farhan Tanvir Nasim
Dept. of Computer Science and Engineering
Varendra University
Rajshahi, Bangladesh
farhan.nasim33@gmail.com

***Abstract*—Plant disease looms over global food security as a significant threat. Despite this, accurately identifying diseases from images taken in real-world field conditions remains a major challenge. Standard classification models often fail in scenarios with complex backgrounds, variable lighting, and image noise characteristic of datasets like PlantDoc. To address this, this study proposes a robust two-stage detection-classification pipeline. The first stage uses a YOLOv11n object detector that was trained to find and separate leaf areas from their messy surroundings. It got a mean Average Precision (mAP@0.5) of 92.9%. In the second stage, these cropped leaf images are put into an ECA-NFNet-L0 classification framework that uses a smart channel attention mechanism to find diseases in detail. On the hard-to-use PlantDoc dataset, our full pipeline gets a final classification accuracy of 78.5% and a weighted F1-score of 78.4%. This decoupled method, which separates localization from classification, makes the model much stronger and is a better way to diagnose plant diseases in the field.**

***Index Terms*—Plant Disease Identification, Computer Vision, Deep Learning, Object Detection, Image Classification, Two-Stage Pipeline, YOLO, ECA-NFNet, Explainable AI, Grad-CAM**

I. Introduction

Global food security is under increasing pressure, with plant diseases being a primary contributor to crop loss and economic instability. The Food and Agriculture Organization (FAO) says that plant diseases cost the world economy $220 billion every year. This makes food shortages worse, which affected 282 million people in 2023. These big losses show how important it is to find effective and scalable ways to manage diseases to protect the world's food supply [1]. While Computer Vision theoretically offers a solution to this problem, most of the existing models are lacking real world generalization capabilities due to being trained on clean, lab-based images.

To solve these problems, we suggest a strong two-stage detection-classification pipeline. The first step uses a You Only Look Once (YOLO) object detector to find and separate leaf areas from their complicated surroundings [2]. In the second stage, an Efficient channel attention - Normalization Free Network (ECA-NFNet) classification framework receives these cropped areas. This framework uses an efficient channel attention mechanism to identify diseases in detail [3]. Our method aims to improve diagnostic accuracy and robustness by separating localization and classification. This study confirms the pipeline's effectiveness on the difficult PlantDoc dataset, with the goal of creating a useful and automated tool for precision agriculture [4].

The two-stage pipeline uses object detection and classification methods to make it possible to automate the detection of plant diseases in a way that is both scalable and efficient.

II. Literature Review

The two-stage pipeline uses object detection and classification methods to make it possible to automate the detection of plant diseases in a way that is both scalable and efficient [5], [6]. Convolutional Neural Networks (CNNs), which automate feature extraction, have shown great success, with some models getting more than 99% accuracy on benchmark datasets [7], [8]. Much of this success has been on datasets like *PlantVillage*, which contains images taken in controlled, laboratory-like settings [9]. But datasets like PlantDoc and Plant Pathology are much harder to work with because they have pictures taken in real-world field conditions with complicated backgrounds and lighting that isn't always the same, as shown in Figure 1 [10], [11].

This distinction between lab-based and in-field data has created a significant performance gap. Table 1 shows that models that do well on clean datasets often do much worse when applied to the noisy, unpredictable images found in PlantDoc. To close this generalization gap, researchers are looking into a number of advanced strategies. These methods include mixing lab and field images during training [12] and using Few-Shot Learning (FSL) to get around data scarcity [13], and incorporating attention mechanisms to assist Models concentrate on diagnostically significant areas of the leaf [14], [15]. Furthermore, Explainable AI (XAI) techniques like LIME and LayerCAM are being used to improve model

979-8-3315-7867-1/25/$31.00 ©2025 IEEE

Fig. 1. Comparison of PlantVillage, Plant Pathology, and PlantDoc datasets.

interpretability, while the scope of research is expanding from simple classification to more nuanced tasks like assessing disease severity [16]–[18].

III. Methodology

In this study, we designed a two-stage detection–classification pipeline for plant disease identification, aimed at addressing the high variability and noise in the PlantDoc dataset [25]. The first stage involves leaf localization using a YOLO-based object detector, while the second stage performs fine-grained classification of the isolated leaf regions using an ECA-NFNet-L0 model [26], [27].

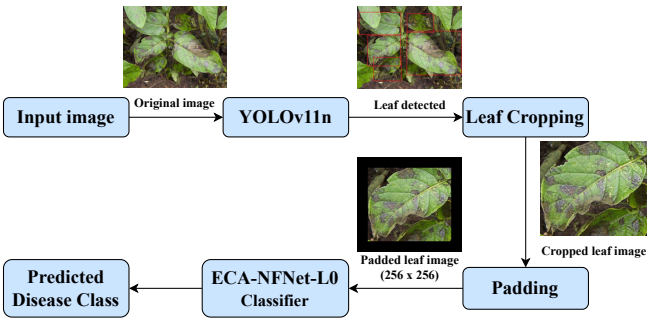

Fig. 2. Overview of the proposed two-stage detection–classification pipeline.

A. Dataset Preparation and Detection Stage

The PlantDoc dataset [25] presents significant challenges due to its heterogeneous backgrounds, uncontrolled lighting, and noisy image quality. Prior research has shown that standard classification models often achieve suboptimal performance on this dataset due to the high variability [20], [21]. For instance, models such as EfficientNet-B3 and ResNet50 with ViT achieve accuracy of 73.31% and 73.83% on the base PlantDoc dataset, highlighting the difficulty of directly classifying raw images with diverse backgrounds.

To address these issues and improve performance, we adopt a two-stage detection–classification pipeline. The first stage isolates leaves using an object detector, and the second stage performs fine-grained classification on cropped leaf regions. We used the PlantDoc-Object-Detection dataset [28], which provides bounding box annotations for leaf regions. Instead of treating each leaf-disease pair as a distinct class in the detection stage, all classes were merged into a single "leaf" category, making the detector responsible solely for isolating leaves from complex environments. Thus, the detector focuses purely on leaf localization.

We trained YOLOv8n and YOLOv11n models on this modified dataset. Let (x_i, y_i, w_i, h_i) denote the coordinates of a bounding box for leaf i, where (x_i, y_i) is the top-left corner and (w_i, h_i) are the width and height. The YOLO model predicts $\hat{x}_i, \hat{y}_i, \hat{w}_i, \hat{h}_i$ for each grid cell, and the loss function combines localization, confidence, and class components:

$$\mathcal{L}_{\text{YOLO}} = \lambda_{\text{coord}} \sum_{i=1}^{N} \sum_{v \in \{x,y,w,h\}} (v_i - \hat{v}_i)^2 + \lambda_{\text{conf}} \sum_{i=1}^{N} (C_i - \hat{C}_i)^2 + \lambda_{\text{cls}} \sum_{i=1}^{N} \sum_{c=1}^{C} (p_{i,c} - \hat{p}_{i,c})^2 \tag{1}$$

where C_i is the object confidence, $p_{i,c}$ is the class probability, and λ are weighting factors. Since all classes were merged into *leaf*, the classification term reduces to a binary presence of leaf.

Experimental evaluation showed that YOLOv11n outperformed YOLOv8n in terms of detection accuracy. However, the classification branch of the detection model performed poorly, motivating the need for a separate classification network. For this, the YOLOv11n detector was trained to crop leaf regions from the object detection dataset. Let I be an input image and $B = \{(x_i, y_i, w_i, h_i)\}$ be the set of detected bounding boxes. Each leaf crop I_i is obtained as:

$$I_i = I[y_i : y_i + h_i,\ x_i : x_i + w_i]. \tag{2}$$

The resulting cropped dataset consists of individual leaf images with reduced background clutter, which serve as input for the subsequent classification stage. Detected bounding boxes were then used to crop leaf regions, producing a new dataset consisting only of leaf patches.

Because these cropped patches often exhibited non-square aspect ratios, we developed a padding function that placed each crop into a square canvas before resizing to 224×224 pixels. This procedure ensured that aspect ratio distortions were avoided and that the images were consistent for downstream classification.

B. Data Augmentation and Sampling Strategy

To improve generalization capability, we applied strong data augmentation during training. Each cropped image was square-padded, resized, and then passed through `TrivialAugmentWide`, which automatically applies diverse transformations such as rotations, color jittering, and shearing. The resulting images were normalized to the ImageNet mean and variance to ensure compatibility with the

TABLE I
COMPARISON OF PLANT DISEASE IDENTIFICATION STUDIES ACROSS DIFFERENT DATASETS

Paper	Method	Dataset	Performance	Notes / Limitations
Other Datasets				
[19]	Hybrid CNN Ensemble + ViT	PlantVillage (subset)	99.24% (Apple), 98% (Corn)	High complexity, many parameters
[17]	LeafConvNeXt + Attention + XAI	PlantVillage	99.68%	Needs validation on real-world data
[14]	Attention-based Deep Network	PlantVillage	99.97%	May overlook subtle symptoms
[16]	EfficientNetB0 + LIME	Kaggle "New Plant Diseases"	99.69%	Post-hoc XAI may not fully reflect model logic
PlantDoc Dataset				
[20]	EfficientNet-B3	PlantDoc	73.31%	Data scarcity and real-world variation
[21].	ResNet50 and ViT models	PlantDoc	73.83%	Insufficient testing of cross-domain generalization
[22]	SVM + Siamese Twin Network	PlantDoc	62.76%	Limited capacity of classical ML
[23]	MobileNet	PlantDoc	60.14%	Lightweight network struggles with complex images
[24]	MobileNet-V2	PlantDoc	69.80%	Generalization issues on field images

TABLE II
YOLOV11N TRAINING SETUP AND HYPERPARAMETERS FOR LEAF DETECTION.

Property	Value
Model Architecture	YOLOv11n
Model Size	7.2M parameters
Input Resolution	640×640
Batch Size	16
Learning Rate	0.01 (cosine scheduler)
Optimizer	SGD with momentum 0.937
Number of Epochs	100
Weight Decay	0.0005
Augmentations	Mosaic, Mixup, HSV color space, Flip, Rotate
Anchor Boxes	Auto-learned from dataset
Loss Function	$\mathcal{L}_{\text{YOLO}}$ (coord + conf + class)

pretrained backbone. For validation and testing, only resizing and normalization were applied.

Class imbalance is a common challenge in PlantDoc, where certain diseases are heavily underrepresented. To counter this, we employed a Weighted Random Sampling strategy. If n_c denotes the number of samples in class c, the class weight is defined as

$$w_c = \frac{1}{n_c}, \tag{3}$$

and the probability of sampling an instance i with label y_i becomes

$$p_i = \frac{w_{y_i}}{\sum_{j=1}^{N} w_{y_j}}. \tag{4}$$

This sampling distribution ensures that rare classes are sampled more frequently, balancing their contribution during optimization.

C. Classification with ECA-NFNet-L0

Following the detection stage, the input to the classification stage would be the leaf patches that strictly contain each leaf with their pathology. This pre-processed images (discussed in subsection III-B) are then classified using a convolutional neural network. For this, the ECA-NFNet-L0 architecture was employed. NFNets are a family of convolutional networks that remove normalization layers entirely, instead stabilizing training through Scaled Weight Standardization and residual branch re-scaling. Specifically, the weight of a convolutional kernel W is standardized by its mean $\mu(W)$ and standard deviation $\sigma(W)$:

$$\hat{W} = \frac{W - \mu(W)}{\sigma(W)}, \tag{5}$$

and the output of the residual branch is scaled by a constant factor $\alpha \approx 0.2$ before being added to the shortcut connection:

$$y = x + \alpha F(x, \hat{W}), \tag{6}$$

where $F(x, \hat{W})$ denotes the residual transformation. This reparameterization removes the need for batch normalization while maintaining variance stability. Additionally, Adaptive Gradient Clipping (AGC) was applied to control gradient magnitudes relative to their corresponding weights.

To enhance representational power, ECA modules were integrated within NFNet blocks. The ECA mechanism first applies global average pooling to compute a channel descriptor:

$$z_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} x_c(i, j), \tag{7}$$

where x_c is the feature map of channel c. Unlike squeeze-and-excitation, ECA avoids dimensionality reduction by applying a lightweight 1D convolution with kernel size k across the aggregated channel descriptors:

$$a = \sigma(\text{Conv1D}_k(z)), \tag{8}$$

where σ denotes the sigmoid function. The final channel-wise attention reweights the input feature maps as

$$y_c = a_c \cdot x_c. \tag{9}$$

The L0 variant of NFNet, used here, is derived from NFNet-F0 but is optimized for computational efficiency by

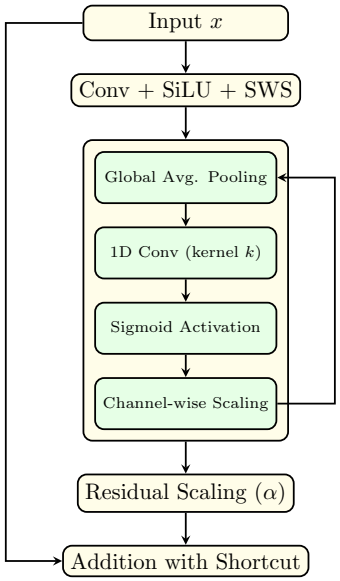

Fig. 3. Structure of an NFNet block with integrated ECA module.

reducing depth and width. It employs SiLU activations and was pretrained on ImageNet-1K, achieving 82.6% Top-1 accuracy. For this work, the final classification head was replaced with a Dropout layer with probability $p = 0.5$ followed by a fully connected layer sized to the number of plant disease categories.

D. Fine-Tuning Strategy

The fine-tuning procedure consisted of training the model using AdamW with a weight decay of 1×10^{-4}. Differential learning rates were applied, with 10^{-4} for the classification head and 10^{-6} for the backbone, ensuring that pretrained features were adapted conservatively. The loss function was cross-entropy with label smoothing, expressed as

$$\ell = -\sum_{k=1}^{K} \left((1-\epsilon) y_k + \frac{\epsilon}{K} \right) \log p_k, \tag{10}$$

where $\epsilon = 0.1$ is the smoothing factor, y_k is the one-hot encoded target, and p_k is the predicted probability for class k. A ReduceLROnPlateau scheduler was used to reduce the learning rate when validation loss failed to improve. Training was conducted for 50 epochs, with model weights checkpointed based on validation accuracy.

E. Hyperparameters and Training

Table III summarizes the ECA-NFNet-L0 training setup and key hyperparameters used for leaf disease classification. The model was fine-tuned from an ImageNet-1K pretrained backbone, with differential learning rates applied to the backbone and classification head. Weighted random sampling was used to mitigate class imbalance, and label smoothing ($\epsilon = 0.1$) was applied to the cross-entropy loss.

The model was trained for 50 epochs. Figure 4 shows the training and validation loss and accuracy curves. Initially, both

TABLE III
ECA-NFNET-L0 TRAINING SETUP AND HYPERPARAMETERS FOR LEAF DISEASE CLASSIFICATION.

Property	**Value**
Model Architecture	ECA-NFNet-L0
Pretrained Backbone	ImageNet-1K
Input Resolution	224×224
Batch Size	32
Optimizer	AdamW
Weight Decay	1×10^{-4}
Learning Rate	10^{-6} (backbone), 10^{-4} (head)
Scheduler	ReduceLROnPlateau
Loss Function	Cross-entropy with label smoothing ($\epsilon = 0.1$)
Number of Epochs	50
Dropout Probability	0.5
Activation Function	SiLU
ECA Kernel Size	Automatically determined (default from ECA module)

training and validation accuracies increased rapidly, reflecting effective learning from the pretrained features. The validation accuracy reached a maximum of 79.1%, demonstrating the model's generalization capability on the cropped leaf dataset.

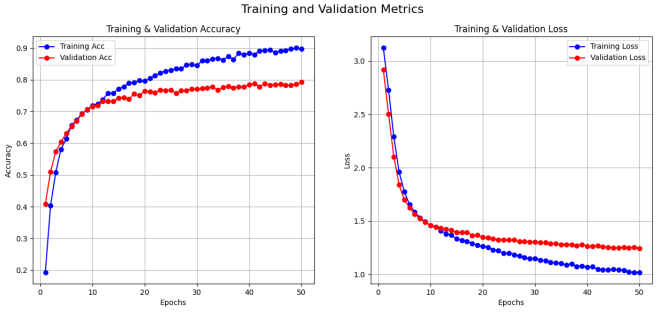

Fig. 4. Training and validation loss and accuracy curves for ECA-NFNet-L0 during fine-tuning on the cropped leaf dataset.

The learning curves indicate stable convergence, with the training loss gradually decreasing and the validation accuracy improving without overfitting. Early epochs show rapid improvement due to transfer learning from the ImageNet pretrained backbone, while later epochs refine the classifier with more subtle adjustments to the high-level features.

RESULTS AND DISCUSSION

F. Object Detection Results

This section presents the experimental evaluation of YOLO models for leaf detection. Performance is first compared between YOLOv8 and YOLOv11 on the base multi-class dataset, followed by YOLOv11 performance on the simplified binary-class merged leaf dataset.

Table IV summarizes YOLOv8 and YOLOv11 performance on the base dataset. YOLOv11 outperforms YOLOv8 across all metrics (mAP@0.5, mAP@0.5:0.95, F1-Score) indicating better detection accuracy and reliability.

YOLOv11 was retrained on the merged leaf dataset, reducing the detection task to binary classification ('leaf' vs

TABLE IV
PERFORMANCE COMPARISON ON BASE AND MERGED LEAF DATASETS.

Metric	YOLOv8	YOLOv11	YOLOv11 (Merged)
mAP@0.5	63.4	69.5	**92.9**
mAP@0.5:0.95	51.0	54.0	**72.0**
Precision	65.0	70.0	**90.0**
Recall	60.0	72.0	**85.0**
F1-Score	59.0	65.0	**87.0**

'background'). Table IV (last column) shows substantial gains in mAP@0.5, mAP@0.5:0.95, and F1-Score. Precision and Recall improvements indicate highly reliable detection.

Figures 5 and 6 present the PR curve and normalized confusion matrix for the merged dataset. The PR curve confirms high precision across recall levels. The confusion matrix demonstrates a strong True Positive Rate, validating model reliability. Note that here, the 'background' category represents false positives and false negatives. True negatives class is not included as per standard object detection metrics.

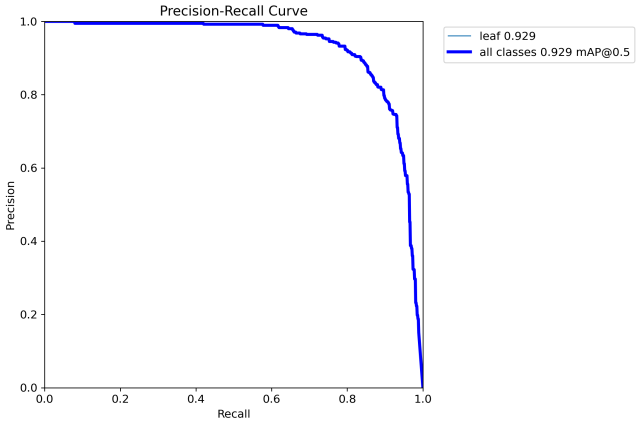

Fig. 5. YOLOv11 PR Curve on Merged Leaf Dataset (mAP@0.5 = 92.9%).

G. Classification Results

The trained leaf classification model was evaluated on the held-out test set, achieving an overall accuracy of 78.5% and a weighted F1-score of 78.4%. Table V presents detailed class-wise metrics.

1) Explainable AI: Grad-CAM Visualizations: Grad-CAM was applied to selected test images to interpret model decisions. Figures 7–8 show three representative cases. The highlighted regions confirm that the model focuses on relevant leaf areas, ensuring that predictions are based on meaningful features rather than background artifacts.

Overall, the classification model achieves robust multi-class performance, and Grad-CAM visualizations validate that decisions are made using physically meaningful leaf features.

IV. CONCLUSION

This study introduced a two stage detection-classification pipeline designed to overcome the challenges of plant disease identification in noisy, real-world environments. By first employing a YOLOv11n model to accurately localize and crop

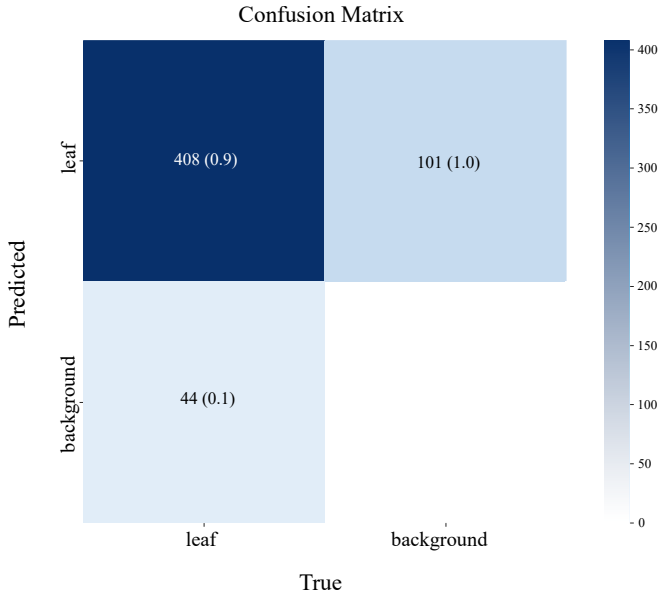

Fig. 6. Confusion Matrix for YOLOv11 with Normalized Values and Raw Counts.

TABLE V
CLASS-WISE PERFORMANCE ON TEST SET.

Class	Precision	Recall	F1-score	Support
Apple Scab Leaf	0.727	0.686	0.706	35
Apple Leaf	0.833	0.796	0.814	44
Apple Rust Leaf	0.771	0.844	0.806	32
Bell Pepper Leaf	0.851	0.864	0.857	66
Bell Pepper Leaf Spot	0.680	0.680	0.680	50
Blueberry Leaf	0.880	0.852	0.866	155
Cherry Leaf	0.840	0.824	0.832	51
Corn Gray Leaf Spot	0.500	0.385	0.435	13
Corn Leaf Blight	0.867	0.890	0.878	73
Corn Rust Leaf	0.875	0.955	0.913	22
Peach Leaf	0.920	0.853	0.885	122
Potato Leaf Early Blight	0.625	0.588	0.606	68
Potato Leaf Late Blight	0.596	0.705	0.646	44
Raspberry Leaf	0.934	0.892	0.912	111
Soyabean Leaf	0.639	0.867	0.736	45
Squash Powdery Mildew Leaf	0.932	0.982	0.957	56
Strawberry Leaf	0.894	0.955	0.923	88
Tomato Early Blight Leaf	0.553	0.539	0.546	39
Tomato Septoria Leaf Spot	0.602	0.663	0.631	89
Tomato Leaf	0.742	0.904	0.815	73
Tomato Leaf Bacterial Spot	0.417	0.333	0.370	45
Tomato Leaf Late Blight	0.756	0.633	0.689	49
Tomato Leaf Mosaic Virus	0.526	0.714	0.606	42
Tomato Leaf Yellow Virus	0.904	0.769	0.831	160
Tomato Mold Leaf	0.691	0.580	0.630	50
Grape Leaf	0.872	0.911	0.891	45
Grape Leaf Black Rot	0.818	0.750	0.783	24
Macro Avg	0.750	0.756	0.750	1691
Weighted Avg	0.789	0.785	0.784	1691
Accuracy		0.785		

leaf regions, our approach effectively mitigates the impact of complex backgrounds and variable lighting conditions, achieving a mean average precision (mAP@0.5) of 92.9%. The subsequent classification stage used an ECA-NFNet-L0 model, which achieved an overall accuracy of 78.5% and a weighted F1-score of 78.4% on the isolated leaf images. This result marks a notable improvement over models that

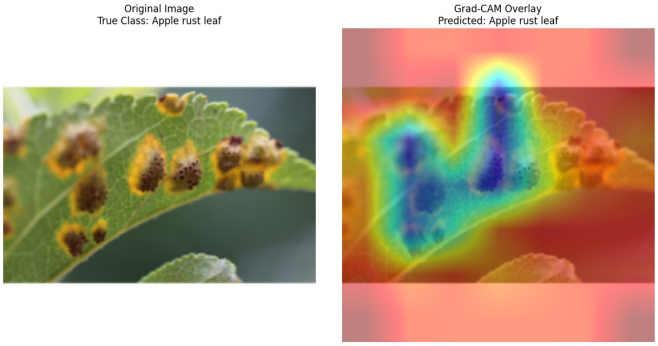

Fig. 7. Grad-CAM visualization for Apple Rust Leaf. The model focuses on characteristic diseased regions.

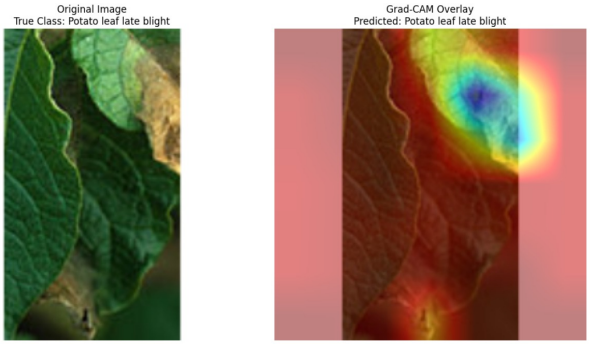

Fig. 8. Grad-CAM visualization for Potato Leaf Late Blight. The model correctly attends to relevant infected areas.

directly classify raw images from the challenging PlantDoc dataset. Furthermore, Grad-CAM visualizations confirmed that the model's predictions were based on relevant pathological features on the leaves, ensuring interpretability and trustworthiness.

The success of this two-stage framework demonstrates a practical and effective strategy for developing scalable and accurate agricultural tools. By decoupling the tasks of finding the object of interest and classifying it, our system provides a more resilient solution for real-world deployment.

References

[1] O. O. Alabi, S. A. Ajagbe, O. A. Adeaga, and M. O. Adigun, "Investigating fuel adulteration using arduino as an engine protection device (epd)," *Journal of Hunan University Natural Sciences*, vol. 50, no. 9, pp. 106–113, 2023.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[3] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pp. 6105–6114, 2020.

[4] S. Sushant and B. R. Pant, "Plantdoc: A large dataset for plant disease detection," *Journal of Agricultural Informatics*, vol. 12, no. 1, pp. 27–37, 2021.

[5] A. Bhargava *et al.*, "Plant leaf disease detection, classification, and diagnosis using computer vision and artificial intelligence: A review," *IEEE Access*, vol. 12, pp. 37443–37469, 2024.

[6] A. Upadhyay *et al.*, "Deep learning and computer vision in plant disease detection: A comprehensive review of techniques, models, and trends in precision agriculture," *Artificial Intelligence Review*, vol. 58, p. 92, 2025.

[7] M. Vardhan and S. Sharma, "Enhancing plant pathology with cnns: A hierarchical approach for accurate disease identification," in *2024 13th International Conference on Software and Computer Applications (ICSCA 2024)*, pp. 159–164, 2024.

[8] S. Natarajan, P. Chakrabarti, and M. Margala, "Robust diagnosis and meta visualizations of plant diseases through deep neural architecture with explainable ai," *Scientific Reports*, vol. 14, p. 13695, 2024.

[9] D. P. Hughes and M. Salathé, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," 2015. arXiv preprint arXiv:1511.08060.

[10] D. Singh *et al.*, "Plantdoc: A dataset for visual plant disease detection," 2019. arXiv preprint arXiv:1911.10317.

[11] R. Thapa, N. Snavely, S. Belongie, and A. Khan, "The plant pathology 2020 challenge dataset to classify foliar disease of apples," 2020. arXiv preprint arXiv:2004.11958.

[12] A. Ahmad, A. El Gamal, and D. Saraswat, "Toward generalization of deep learning-based plant disease identification under controlled and field conditions," *IEEE Access*, vol. 11, pp. 9042–9057, 2023.

[13] M. Rezaei *et al.*, "Plant disease recognition in a low data scenario using few-shot learning," *Computers and Electronics in Agriculture*, vol. 219, p. 108812, 2024.

[14] A. Bera, D. Bhattacharjee, and O. Krejcar, "An attention-based deep network for plant disease classification," *Machine Graphics & Vision*, vol. 33, no. 1, pp. 47–67, 2024.

[15] W. Yang *et al.*, "An effective image classification method for plant diseases with improved channel attention mechanism aecanet based on deep learning," *Symmetry*, vol. 16, no. 4, p. 451, 2024.

[16] N. Nigar *et al.*, "Improving plant disease classification with deep-learning-based prediction model using explainable artificial intelligence," *IEEE Access*, vol. 12, pp. 100005–100014, 2024.

[17] F. Lu *et al.*, "Leafconvnext: Enhancing plant disease classification for the future of unmanned farming," *Computers and Electronics in Agriculture*, vol. 233, p. 110165, 2025.

[18] R. Bandi, S. Swamy, and C. S. Arvind, "Leaf disease severity classification with explainable artificial intelligence using transformer networks," *International Journal of Advanced Technology and Engineering Exploration*, vol. 10, no. 100, pp. 278–302, 2023.

[19] S. Aboelenin, F. A. Elbasheer, M. M. Eltoukhy, W. M. El-Hady, and K. M. Hosny, "A hybrid framework for plant leaf disease detection and classification using convolutional neural networks and vision transformer," *Complex & Intelligent Systems*, vol. 11, p. 142, 2025.

[20] M. S. Krishna, P. Machado, R. I. Otuka, S. W. Yahaya, F. Neves dos Santos, and I. K. Ihianle, "Plant leaf disease detection using deep learning: A multi-dataset approach," *J*, vol. 8, no. 1, 2025.

[21] D. Aggarwal, Y. Mittal, and U. Kumar, "Advancing image classification through parameter-efficient fine-tuning: A study on loRA with plant disease detection datasets," in *The Second Tiny Papers Track at ICLR 2024*, 2024.

[22] M. Chandra, S. Redkar, S. Roy, and P. Patil, "Classification of various plant diseases using deep siamese network." Available online: https://www.researchgate.net/publication/341322315, 2024. Accessed on 14 December 2024.

[23] E. Moupojou, A. Tagne, F. Retraint, A. Tadonkemwa, W. Dongmo, H. Tapamo, and M. Nkenlifack, "Fieldplant: A dataset of field plant images for plant disease detection and classification with deep learning," *IEEE Access*, vol. 11, pp. 35398–35410, 2023.

[24] V. Menon, V. Ashwin, and R. K. Deepa, "Plant disease detection using cnn and transfer learning," in *Proceedings of the 2021 International Conference on Communication*, 2021.

[25] D. Singh, N. Jain, P. Jain, P. Kayal, S. Kumawat, and N. Batra, "Plantdoc: A dataset for visual plant disease detection," in *Proceedings of the 7th ACM IKDD CoDS and 25th COMAD*, pp. 249–253, 2020.

[26] A. Brock, S. De, S. L. Smith, and K. Simonyan, "High-performance large-scale image recognition without normalization," *arXiv preprint arXiv:2102.06171*, 2021.

[27] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "Eca-net: Efficient channel attention for deep convolutional neural networks," *arXiv preprint arXiv:1910.03151*, 2020.

[28] D. Singh, N. Jain, P. Jain, P. Kayal, S. Kumawat, and N. Batra, "Plantdoc: A dataset for visual plant disease detection," in *Proceedings of the 7th ACM IKDD CoDS and 25th COMAD*, CoDS COMAD 2020, (New York, NY, USA), p. 249–253, Association for Computing Machinery, 2020.