Hybrid Adaptive Swarm Enhanced Vision Transformer for Accurate Corn Leaf Disease Prediction

Nilam Sachin Patil[1,*], E. Kannan[1]

[1]Department of Computer Science & Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, India

Emails: snilampatil2017@gmail.com; ek081966@veltech.edu.in

Abstract

Early and precise detection of corn leaf diseases is important for maintaining crop yield and quality. This work suggests a new end-to-end system Hybrid Adaptive Swarm-enhanced Vision Transformer (HAS-ViT) to overcome the limitations of current techniques such as poor accuracy, high computational expense, and overfitting and inefficient feature extraction. The suggested framework combines a three-stage pipeline such as segmentation, classification and optimization to overcome the issues. First, Adaptive Gradient Masking with Color Entropy (AGM-CE) is a novel segmentation technique that isolates diseased areas through an integration of local color entropy and gradient energy in the LAB color space. This guarantees accurate area selection and removal of the background. Then, a transformer model is constructed named Vision Transformer with Enhanced Visual Attention (ViT-EVA). It integrates depthwise attention layers as well as lesion-aware region concentration, enhancing separation of disease classes and model simplification. Finally, Adaptive Bio-Inspired Gradient Tuning (ABGT) optimizer integrates the Bat Algorithm, AdamW and gradient sign flipping for effective learning and convergence. The mechanism speeds up convergence, prevents local minima and maintains exploration exploitation trade-offs at training. The performance of proposed work is measured on a corn disease dataset and performs at 98.1% accuracy and 0.12 loss than conventional and current transformer-based models.

Keywords: Corn Leaf Disease; Vision Transformer; Adaptive Swarm Optimization; Image Segmentation; Deep Learning

1. Introduction

Corn is the most important cereal crop in the world, used mainly for food security, animal feed, and biofuel. Corn crops are extremely susceptible to numerous foliar diseases like Cercospora Leaf Spot, Northern Leaf Blight, and Common Rust. These diseases can significantly affect plant growth, decrease photosynthetic efficiency, and eventually cause major yield losses if not detected and controlled at early stages. Manual examination by agricultural professionals is time-consuming, prone to errors, and unscalable, particularly over large farmlands. Thus, an automated, accurate, and real-time corn leaf disease prediction system is needed to aid precision agriculture and enhance overall crop health monitoring [1].

Existing deep learning methods for predicting plant diseases rely mainly on Convolutional Neural Networks (CNNs) for classification and general image processing or U-Net-based models for segmentation. Most optimization is achieved using conventional methods such as Adam, SGD, and RMSprop. Although the methods are of reasonable accuracy, they are hampered by the need for extensive datasets, their susceptibility to background noise, and rigid learning policies. Segmentation techniques usually aim towards thresholding or contour-based techniques that cannot segment complex lesion patterns. ResNet, VGG, and DenseNet classifiers, while efficient, are computationally intensive and non-interpretable in lesion-centric detection [2].

Despite progress, current systems encounter major issues like poor generalization to various field conditions, overfitting due to model complexity, and the inability to dynamically concentrate on disease-specific information.

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

Most methods of segmenting do not learn the color and texture change of lesions, leading to false alarms. Classification models tend to address the entire image without considering infected areas, lowering accuracy. Moreover, traditional optimizers are subject to local minima and slow convergence, affecting the training efficiency and model robustness. Such limitations restrict the implementation of these models in real-world agricultural scenarios [3].

To surpass the shortcomings of current systems, this study proposes creating a new and effective corn leaf disease prediction model called Hybrid Adaptive Swarm-enhanced Vision Transformer (HAS-ViT). The main aim is to construct a highly accurate and lightweight system that can perform accurate lesion segmentation, resilient disease classification, and optimized learning convergence [4]. The proposed new model presents: (i) AGM-CE, a novel segmentation method that separates disease-affected areas with gradient and color entropy fusion; (ii) ViT-EVA, a transformer classifier featuring depthwise and lesion-aware attention layers; and (iii) ABGT, an adaptive swarm intelligence-inspired optimization algorithm for increased training efficiency. The integrated pipeline is specifically conceived to guarantee higher detection accuracy, mitigate overfitting, and enable real-time deployment on precision agriculture systems [5].

Section 2 explains the existing models, their advantages and disadvantages. Section 3 discusses the proposed work including data preprocessing, augmentation, segmentation, classification and optimization. Section 4 discusses the results of proposed work compared with existing work. Section 5 concludes the proposed work in corn leaf disease prediction and directions for future work.

2. Related works

Lu et al. (2025) introduced LeafConvNeXt, a CNN architecture aimed at improving plant disease classification in unmanned farming applications. The model combines ConvNeXt blocks with attention-augmented feature refinement to achieve better accuracy in real-world scenarios. The primary strength is its resistance to background noise and complicated scenes, although it can be prone to computational overhead. It produced better classification performance in several crop datasets [7].

Zou et al. (2025) integrated CNN and XGBoost for feature extraction to forecast corn constituents by means of near-infrared spectroscopy. It used spectral preprocessing, deep feature learning, and boosting in methodology. Its major advantage is high prediction performance with low overfitting; however, its performance is NIR data quality dependent. The model was superior to the standalone CNN and XGBoost baselines [8]. Sharma et al. (2025) proposed SoyaTrans, a transformer model for fine-grained soybean leaf disease classification. It utilizes attention layers to learn discriminative features. The model performs well in learning detailed features but consumes high memory and data. Experiments show improved precision and F1 scores compared to CNN baselines [9]. Padshetty (2025) created a twin vision transformer model with deformable attention for crop disease classification. This model improves spatial flexibility in feature learning. It provides robust adaptability to varying leaf structures but expands training time. Experimental validation validates better performance compared to ViT-only models [13].

Yilma et al. (2025) introduced Attentive Self-supervised Contrastive Learning Model (ASCL) for plant disease classification. It acquires knowledge from unlabeled data through contrastive pairs and attention modules. It holds is label efficiency and scalability, although it is based on well-designed contrasts. Outcomes show encouraging accuracy in low-supervision conditions [14]. Ren et al. (2025) used a ResNet-18SE model for the recognition of surface action potentials in maize leaves. The approach combines residual connections with channel attention for classification based on signals. It provides robust signal interpretation but is less efficient on vision tasks. The model performed with high sensitivity for the detection of maize disease [15].

Jin et al. (2025) proposed Shuffle-PG, a light model for plant disease image retrieval via deep metric learning. The model integrates shuffled grouped convolutions with metric loss. It is superior in speed and retrieval accuracy but has no classification ability. Experiments confirm its efficacy in visual search [16].

Anand et al. (2025) introduced QEFS is a quantum-inspired evolutionary feature selection framework for prediction of plant diseases. It integrates evolutionary search with quantum computing concepts. Although revolutionary in nature, the method necessitates expert tuning. Feature subset relevance and classification measures were substantially enhanced [17]. Tang et al. (2025) applied CGS-YOLO to UAV multispectral imagery to segment maize seedlings from weeds. The model developed from YOLO for spectral inputs maintains swift and efficient detection of objects and achieves speed as well as precision at an early stage in the control of weeds [18].

3. Proposed Method

The suggested HAS-ViT architecture in figure 1 represents the adaptive preprocessing, augmentation, lesion-specific segmentation, transformer-based classification, and hybrid optimization for precise corn leaf disease prediction. Preprocessing involves resizing, LAB color space conversion, and contrast enhancement to normalize and emphasize disease areas. The AGM-CE segmentation separates infected areas based on entropy and gradient

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

fusion, yielding cleaner inputs for classification. The ViT-EVA classifier uses attention to lesion-related features through patch embeddings and a Lesion Reweighting Factor. Finally, the training is optimized with the use of ABGT, which is an amalgamation of Bat Algorithm, AdamW, and gradient sign flip to provide quicker convergence and better learning.

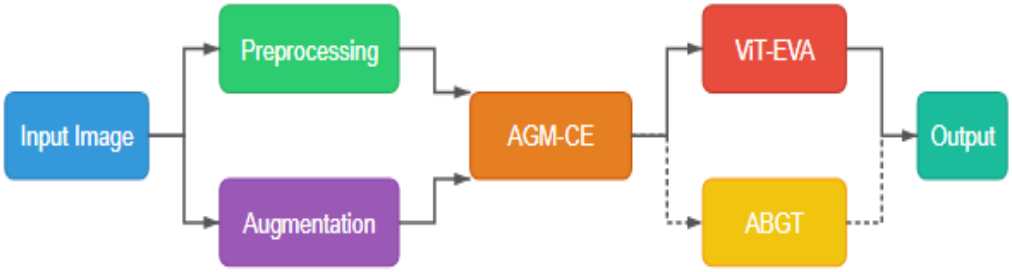


Figure 1. Overall architecture of proposed work

3.1 Dataset Description

The dataset contains 4,188 leaf images belonging to four classes: Common Rust with 1,306 images, Gray Leaf Spot with 574 images, Blight with 1,146 images, and Healthy leaves with 1,162 images. The dataset gives an equal representation of disease and non-disease, allowing for effective training and testing of plant disease classification models. Table 1 describes the number of classes and its count in each class [6].

Table 1: Dataset details

Class Label	Disease Type	Number of Images
0	Common Rust	1,306
1	Gray Leaf Spot	574
2	Blight	1,146
3	Healthy	1,162
Total	**4,188**	

3.2 Data preprocessing

Preprocessing is essential to enhance the quality of corn leaf images before segmentation and classification. The following steps are performed [10].

a. Image resizing

All images are resized to fixed dimension 224*224 pixels to ensure uniform input size for the vision transformer and efficient memory usage as given in Eq.(1).

$$I_{resized}(x, y) = Resize(I(x, y), 224{,}224) \qquad (1)$$

b. Noise removal

Gaussian blur is applied to reduce background noise and smoothen the image. In Eq.(2), $G(x, y, \sigma)$ is a Gaussian kernel with standard deviation σ =1.

$$I_{demoise}(x, y) = I_{resized}(x, y) * G(x, y, \sigma) \qquad (2)$$

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

c. Color space conversion

Images are converted from RGP to LAB color space for accurate lesion detection using color entropy as given in Eq.(3).

$$I_{LAB} = convert\left(I_{demoise}(x, y)\right) \qquad (3)$$

d. Contrast normalization

Histogram equalization is applied on the L channel to enhance the lesion contrast as represented in Eq.(4).

$$L' = Equalize(L) \qquad (4)$$

Then the background pixels are removed using 3.3 Adaptive Gradient Masking with Color Entropy (AGM-CE) segmentation mask before passing to classifier.

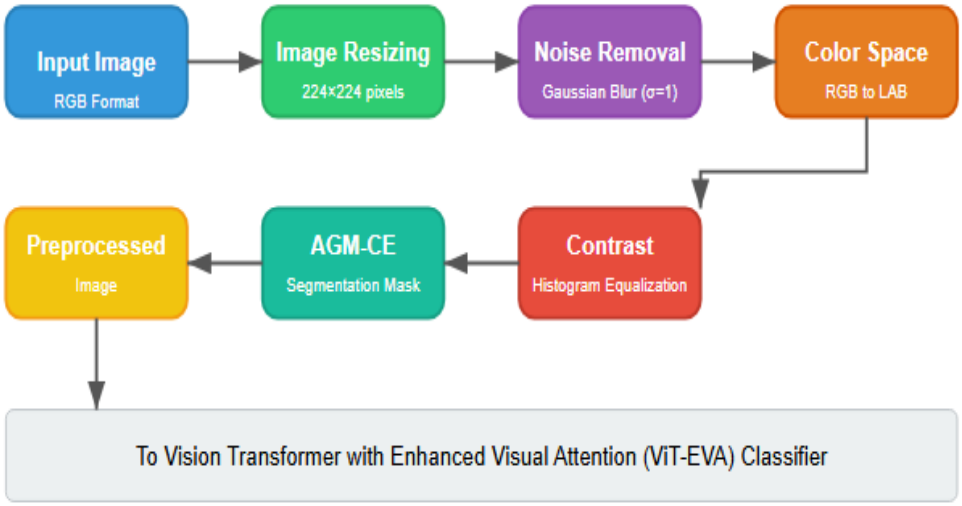


Figure 2. Working flow of the data preprocessing

In figure 2, the data preprocessing workflow begins with the acquisition of corn leaf images, and then submit the images to necessary preprocessing steps to improve quality and uniformity. First, resize the images into a standard size to obtain consistency throughout the dataset. Denoising methods are then implemented to eliminate unnecessary noise and artifacts so that disease patterns are better visualized. It is transformed to the LAB color space to dissociate luminance and color information for better feature extraction and segmentation. These preprocessing tasks condition the data for additional segmentation and augmentation in order to increase the accuracy and robustness of the disease diagnosis model.

3.3 Data Augmentation

Gray Leaf Spot class (574 images) is underrepresented as compared to other classes. Mistakes in choosing data samples lead to sampling bias. To overcome this bias concern, data augmentation is necessary to apply. To prevent overfitting and improve model generalization several augmentation techniques are applied [9]. The geometric transformation are applied with ±20 degrees, horizontal & vertical flipping and random scaling of 0.8x to 1.2x using the Eq.(5).

$$I_{aug} = Rotate(I) + Flip(I) + Scale(I) \qquad (5)$$

Color jittering is applied with random changes in brightness, contrast and saturation to simulate variable lighting conditions. In Eq.(6), the variables $\alpha \in [0.8,1.2]$ and $\beta \in [-10,+10]$.

$$I_{jitter} = \alpha I + \beta \qquad (6)$$

A rectangular patch is erased to simulate occlusion and strengthen robustness using Eq.(7).

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

$$I(x', y') = 0 \quad for(x', y') \in R_{random} \qquad (7)$$

In Eq.(8), the Gaussian noise injection adds random noise to simulate real world imaging imperfections where σ=0.01.

$$I_{noise} = I + N(0, \sigma^2) \qquad (8)$$

This ensures the combination of preprocessing and augmentation that the model becomes more robust, adaptive to different conditions and avoids over fitting even with limited training samples.

Table 2: Dataset details after augmentation

Class Label	**Disease Type**	**Number of Images (Before augmentation)**	**Number of Images (After augmentation)**
0	Common Rust	1,306	2612
1	Gray Leaf Spot	574	1,722
2	Blight	1,146	2,292
3	Healthy	1,162	2,324
Total		**4,188**	**8,950**

Table 2 shows the dataset size before and after augmentation for four maize leaf condition classes. The dataset originally contained 4,188 images spread out over Common Rust (1,306), Gray Leaf Spot (574), Blight (1,146), and Healthy (1,162) classes. In order to strengthen the robustness of the model and avoid overfitting, different types of augmentation including geometric transformation, color jittering, random erasing, and Gaussian noise were utilized. Consequently, the size of the dataset was substantially increased to 8,950 images, and the augmented images for every class were Common Rust (2,612), Gray Leaf Spot (1,722), Blight (2,292), and Healthy (2,324), which created a more diverse and balanced training set for the model.

3.4 Adaptive Gradient Masking with Color Entropy (AGM-CE)

Adaptive Gradient Masking using Color Entropy (AGM-CE) is a segmentation method that aims to separate disease-infected areas in images utilizing both edge and color information. The operation starts with transforming an input RGB image into the LAB color space, where lightness (L) is separated from chromaticity (A and B) channels. Entropy is computed on the A and B channels to represent color variability, and a Sobel gradient is computed on the L channel to highlight edges. These features are then combined with weighted averaging (α = 0.6 for entropy, β = 0.4 for gradient) to create a compound feature map. One then applies a thresholding process to this combined map to create a binary mask that essentially points out the affected areas for classification or further analysis. The proposed segmentation method isolates the disease infected region by fusing color entropy and gradient energy in the LAB color space [12].

Step1: Convert RGB to LAB space

Let the image be I(x,y) and the conversion is happened using Eq.(9). It is split into channels L(x,y),A(x,y) ,B(x,y).

$$I_{LAB}(x, y) = Convert\big(I(x, y)\big) \qquad (9)$$

Step2: Compute Local Entropy

In Eq.(10), the local entropy on A and B channels are given.W is a local window and P_{ij} is the normalized histogram probability.

$$E_{AB}(x, y) = \frac{1}{N} \sum_{(i,j)\epsilon w} - P_{ij} log(P_{ij}) \qquad (10)$$

Step 3: Compute Sobel gradient on L channel

The sobel gradient on channel is computed using Eq.(11) where S_x and S_y are sobel kernels.

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

$$G(x,y) = \sqrt{(S_x * L)^2 + (S_y * L)^2} \quad (11)$$

Step 4: Fusion of features and thresholding

The fusion of entropy and sobel gradient are given in Eq.(12) where α=0.6 and β=0.4.

$$M(x,y) = \alpha \cdot E_{AB}(x,y) + \beta \cdot G(x,y) \quad (12)$$

The final mask is computed using Eq.(13) to extract only the diseased region

$$Mask(x,y) = \{if\ M(x,y) > T_{otsu\ otherwise}\} \quad (13)$$

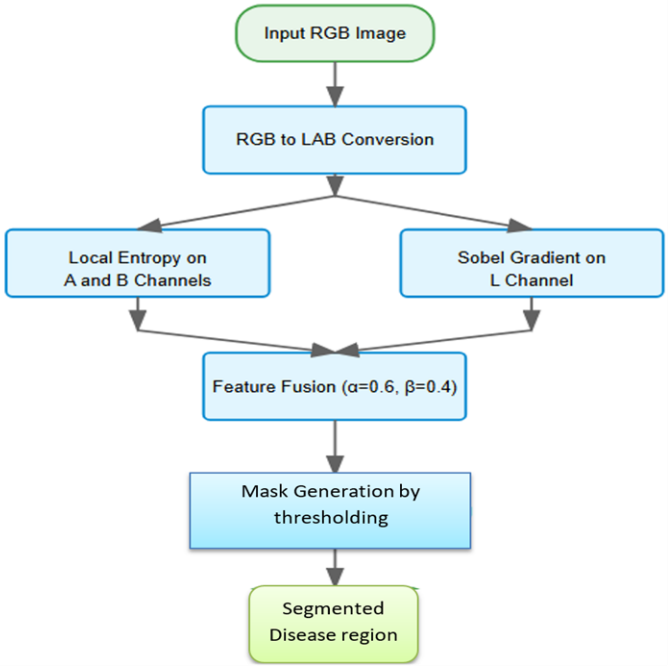


Figure 3. A flow graph of Adaptive Gradient Masking with Color Entropy

Figure 3 depicts a systematic segmentation pipeline in which an input RGB image is converted to the LAB color space to decouple luminance and chromaticity information. Local entropy is calculated on the A and B channels to encode color complexity, whereas Sobel gradients are used on the L channel to identify edges and texture. These two feature maps are combined with weighted parameters (α = 0.6 for gradient and β = 0.4 for entropy) to create a composite feature representation. A thresholding operation is subsequently applied on the combined output to produce an accurate binary mask highlighting the diseased or affected area, facilitating further analysis or classification.

3.5 Vision Transformer with Enhanced Visual Attention (ViT-EVA)

Vision Transformer with Improved Visual Attention (ViT-EVA) is a variant transformer architecture tailored to deal with segmented lesion regions from medical images. Following the segmentation process, the masked image is partitioned into fixed-sized patches, flattened, and linearly projected onto embeddings by a patch embedding mechanism. Positional encoding is appended to preserve spatial information so that the model is aware of the relative location of each patch in the image. This configuration puts the input in place for the transformer encoder layers to study lesion-priority patterns efficiently [10].

Every encoder layer in ViT-EVA uses depthwise attention through the calculation of the Query, Key, and Value matrices and amplifying the attention weights with a Lesion Reweighting Factor (LRF). The LRF integrates normal attention with extra concentration on lesion regions, mixing global and local information equally. The adjusted attention enhances feature prioritization for affected areas. Lastly, the attention outputs go through a multilayer perceptron (MLP) and softmax activation for classification. The rationale behind this specific attention mechanism is to highlight lesion-specific features, resulting in more accurate and explainable predictions.

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

After segmentation, the masked region is sent to a custom transformer model designed to focus on lesion based attention.

a. ***Patch embedding***: Divide the image into fixed size patches p*p flatten and linearly project using Eq.(14).

$$X_p = Flatten(I_{masked})W_p + b_p \qquad (14)$$

b. ***Position encoding*** is computed using Eq.(15)

$$Z_0 = X_p + PE \qquad (15)$$

c. Lesion aware depthwise attention

Each encoder layer calculates the Query(Q=ZW^Q),Key(K=ZW^K) and Value (V= ZW^V). The depthwise attention is applied using Eq.(16).

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \qquad (16)$$

This is enhanced with Lesion Reweighting Factor (LRF) using Eq.(17) where γ∈[0,1].

$$A_{LRF} = \gamma \cdot Attention + (1-\gamma) \cdot Mask\ Focus \qquad (17)$$

The mask focus prioritizes features inside the segmented lesion. The final classification is done with MLP+softmax as given in Eq.(18).

$$\hat{y} = softmax(W_{out}.A_{LRF} + b) \qquad (18)$$

In table 3, the ViT-EVA model is optimized for lesion-centered image classification and starts with a segmented lesion image input of size (1, 224, 224, 3). The patch-embedding layer applies a Conv2D operation with kernel size and stride 16 to generate 196 patches with 768 features each, followed by the addition of learnable positional encodings. The 12 Transformer Encoder blocks from the model's central structure with MHA, LRF, and MLP layers having more than 85 million parameters. The layers within each encoder include layer normalization (18,432 parameters), QKV projections (21.2 million parameters), and MLP blocks (56.6 million parameters), and depthwise attention as well as lesion reweighting are treated as custom operations. Global average pooling pools features, with a dense classification head projecting to four classes with 3,076 parameters. Softmax activation then emits class probabilities. Depthwise attention benefits the strengthening of ViT EVA verses standard ViT.

Table 3: ViT-EVA Model Summary

Layer (Type)	**Output Shape**	**Parameter**	**Description**
Input (Masked Image)	(1, 224, 224, 3)	0	Segmented lesion image
Patch Embedding (Conv2D)	(1, 196, 768)	590,592	P=16 → Conv with kernel=(16,16), stride=16
Position Encoding (Add)	(1, 196, 768)	150,528	Learnable PE of shape (1, 196, 768)
Transformer Encoder ×12	(1, 196, 768)	85,054,464	MHA + LRF + MLP layers
LayerNorm	(1, 196, 768)	1,536 × 12	12× (2×768) = 18,432
Q, K, V Projections (Dense)	(1, 196, 768×3)	3×768×768×12 = 21,233,664	Learnable QKV

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

Depthwise Attention + LRF	(1, 196, 768)	0 (custom op)	Lesion reweighting (γ)
MLP Block (Dense)	(1, 196, 3072) → (1, 196, 768)	2×(768×3072) = 4,718,592 × 12 = 56,623,104	
Global Average Pooling	(1, 768)	0	Token-wise average
Dense (Classification Head)	(1, 4)	3,076	768×4 + 4 bias
Softmax	(1, 4)	0	Output probabilities

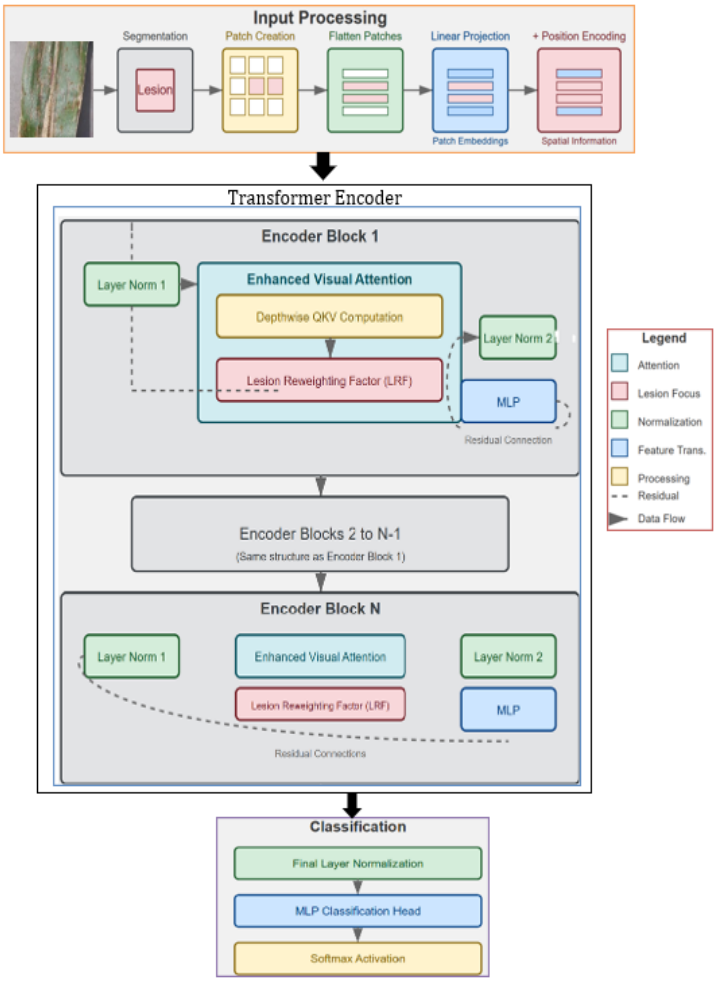


Figure 4. An Architecture of vision Transformer with Enhanced Visual Attention

Figure 4 shows the architecture of the ViT-EVA which starts by splitting the input lesion image into fixed-size patches and embedding them with a convolutional layer. The patch embeddings are then added to learnable positional encodings to preserve spatial information. The embedded patch sequence is fed through several Transformer encoder layers, each consisting of Multi-Head Self-Attention, Layer Normalization, and an MLP block. Improved Visual Attention is introduced through a dedicated lesion-centric reweighting component that emphasizes salient areas with Depthwise Attention and a Lesion Reweighting Factor (γ). Following feature enhancement, Global Average Pooling sums the token-wise outputs, and a dense layer coupled with softmax to obtain class probabilities carries out final classification. This architecture successfully embodies global and lesion-specific features, enhancing classification accuracy.

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

3.6 Adaptive Bio-Inspired Gradient Tuning (ABGT)

Adaptive Bio-Inspired Gradient Tuning (ABGT) is a hybrid optimization method, which unites the global search ability of the Bat Algorithm, the adaptive momentum and weight decay of AdamW, and the resilience of gradient sign flipping. It starts by adjusting the velocity from the echolocation-inspired movement of the Bat Algorithm towards the global optimal solution by the fitness-driven frequency. Next, the gradient is calculated based on AdamW's first-moment estimate, and then a gradient sign flip with a little random perturbation is applied to prevent local minima. The weights are also updated with weight decay for regularization. This combination allows ABGT to learn adaptive learning rates, speed up convergence, and escape from local optima effectively during training [11].

Pseudo code: Adaptive Bio-Inspired Gradient Tuning (ABGT)

Input: Initial parameters θ_0, learning rate η, decay factor λ,

Bat parameters: frequency f_i, velocity v_i, global best x^*,

AdamW β_1 (momentum), δ (flip strength), T (total iterations)

Initialize:

$g_0 = 0$ // First moment

$v_0 = 0$ // Initial velocity

$x_0 = \theta_0$ // Initial position

$x^* = \theta_0$ // Initialize global best

for t = 1 to T do:

1. Compute fitness and frequency f_i based on current loss
2. Update velocity using Bat Algorithm:

   $v_i^{(t+1)} = v_i^t + (x_i^t - x^*) * f_i$

3. Update first moment (AdamW momentum):

   $g_t = \beta_1 * g_{t-1} + (1 - \beta_1) * \nabla L(\theta_t)$

4. Apply Gradient Sign Flip with random perturbation:

   $\theta_{temp} = \theta_t - \eta * sign(g_t) + \delta * random\_flip()$

5. Update parameters with velocity from Bat Algorithm:

   $\theta_{(t+1)} = \theta_{temp} + v_i^{(t+1)}$

6. Apply weight decay (AdamW regularization):

   $\theta_{(t+1)} = \theta_{(t+1)} - \lambda * \theta_t$

7. Update global best x^* if new $\theta_{(t+1)}$ improves fitness

end for

Output: Optimized parameters θ_T

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

To optimize model training, the proposed ABGT is a hybrid of Bat Algorithm, AdamW and gradient sign flip. In Eq.(19), the velocity update inspired by BA where x* is the global best and f_i is the frequency based on fitness.

$$v_i^{t+1} = v_i^t + (x_i^t - x^*)f_i \qquad (19)$$

Gradient update is calculated using Eq.(20) and Gradient sign flip is calculated using Eq.(21).

$$g_t = \beta_1 g_{t-1} + (1 - \beta_1)\nabla L(\theta_t) \quad (20)$$

$$\theta_{t+1} = \theta_t - \eta \cdot sign(g_t) + \delta \cdot random\_flip() \qquad (21)$$

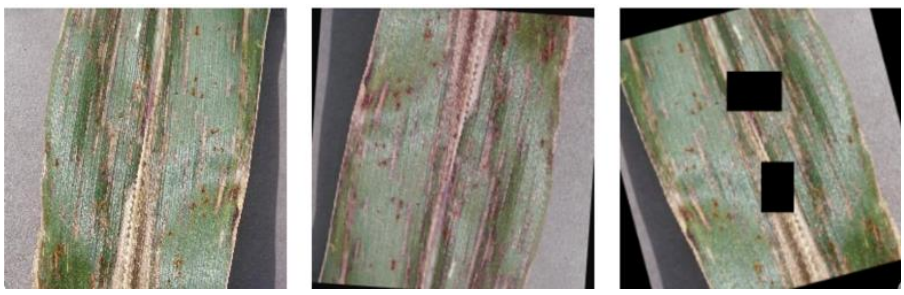
The weight decay is calculated using Eq.(22) and this optimization escapes local minima and provides faster convergence and tunes learning rates adaptively.

$$\theta_{t+1} = \theta_{t+1}\lambda \cdot \theta_t \qquad (22)$$

The Adaptive Bio-Inspired Gradient Tuning (ABGT) optimizer combines aspects of the Bat Algorithm and AdamW to improve efficiency in optimization. First, it initializes parameters such as learning rate (η), decay factor (λ), and Bat Algorithm details such as frequency (f_i) and velocity (v_i). During T iterations, the optimizer calculates fitness and frequency from the current loss, calculates velocity with the Bat Algorithm formula, and adjusts the first moment estimate (g_t) similar to AdamW's momentum calculation. It then performs a gradient sign flip with random perturbation, updates parameters with the adjusted velocity, and performs weight decay for regularization. The global best position (x*) is updated if the new parameters are better in terms of fitness. This hybrid method will seek to incorporate the exploration ability of bio-inspired algorithms with adaptive gradient function of AdamW for effective optimization.

4. Results and Discussion

Figure 5 illustrates a side-by-side comparison of corn leaf images for the four different classes: Common Rust (0), Gray Leaf Spot (1), Blight (2), and Healthy (3). The left column represents the original images as captured, whereas the right column shows the corresponding preprocessed and augmented images. The processed images have been subject to a series of transformations such as rotation, flipping, scaling, color jittering, and random erasing to mimic variations like different viewing angles, lighting, and occlusions.

(a). Original	**(b). Preprocessed and augmented images**
Common Rust (0)	
Gray Leaf Spot (1)	

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

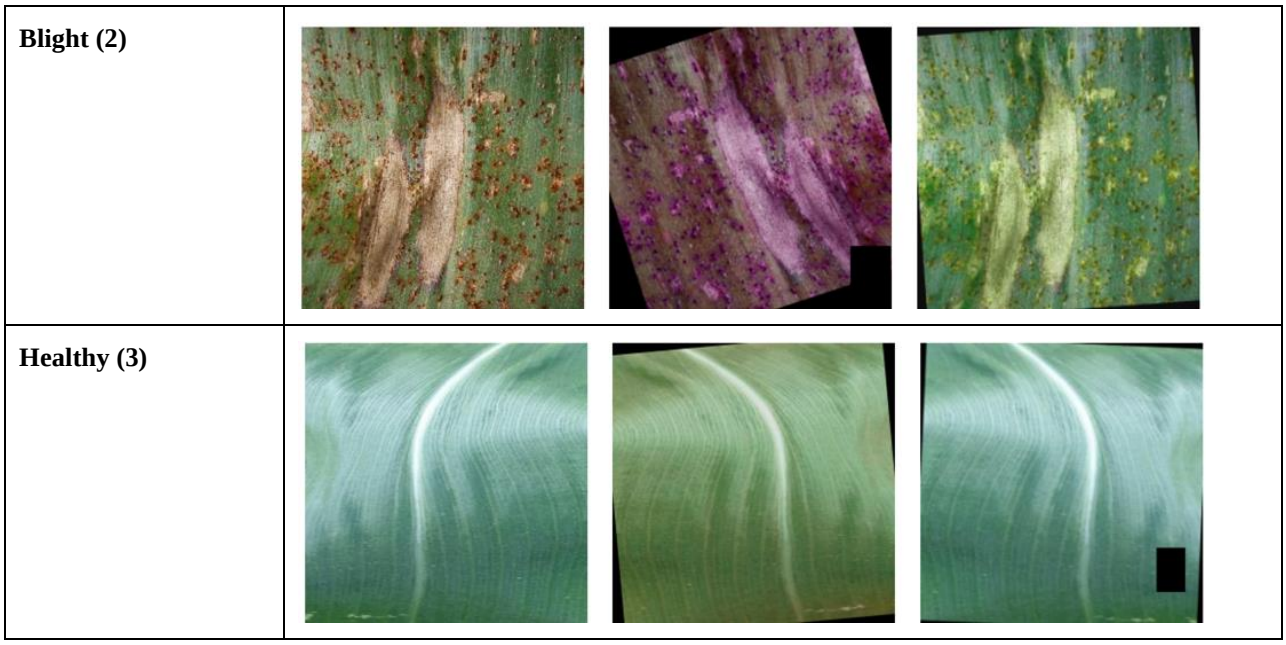


Figure 5. Augmented and preprocessed corn leaf images

The dice metric in Eq.(23) measures the overlap between the predicted mask P and the ground truth mask G.

$$Dice = \frac{2|P \cap G|}{|P| + |G|} \qquad (23)$$

The Eq.(24) measures how well the predicted segmentation overlaps with the ground truth. It is used to assess segmentation accuracy more strictly than dice.

$$IoU = \frac{|P \cap G|}{|P \cup G|} \qquad (24)$$

The following Eq.(25) indicates how many pixels are classified correctly in the segmented mask.

$$Pixel\ accuracy = \frac{No\ of\ correctly\ predicted\ pixels}{Total\ pixels} \qquad (25)$$

The evaluation the models capability in identifying the correct disease category using the following Eq. (26) to (29) where TP is true positive, TN is true negative, FP is false positive, FN is false negative and TN is true positive.

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \qquad (26)$$

$$Precision\ = \frac{TP}{TP + FP} \qquad (27)$$

$$Recall\ = \frac{TP}{TP + FN} \qquad (28)$$

$$F1\ Score = 2 * \frac{precision * recall}{precision + recall} \qquad (29)$$

Figure 6 presents the segmented results of corn leaf images processed by the proposed Adaptive Gradient Masking with Color Entropy (AGM-CE) algorithm. The segmentation pipeline starts with converting the RGB images to the LAB color space to decouple luminance from color information. Color entropy is computed on the A and B channels to describe chromatic variation, and Sobel gradients are used on the L channel to detect structural edges. These characteristics are then combined using corresponding weights ($\alpha = 0.6$ for entropy and $\beta = 0.4$ for gradient) to create a composite feature map. Otsu's thresholding is utilized to produce a binary mask that best brings out the diseased or healthy areas in the leaves.

Tables 4 and 5 indicate the better segmentation performance of the new ViT-EVA approach. Table 4 presents class-wise results where Dice Score, IoU, and Pixel Accuracy always exceed 0.91, 85%, and 95% respectively

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

with an average Dice of 0.925. Compared (Table 5), ViT-EVA beats current state-of-the-art models such as U-Net, Mask R-CNN, and DeepLabV3+ on all these measurements with a superior highest Dice Score (0.925), IoU (87.1), and Accuracy (96.55%), as evidenced by its prowess in perfectly segmenting other leaf conditions in plants.

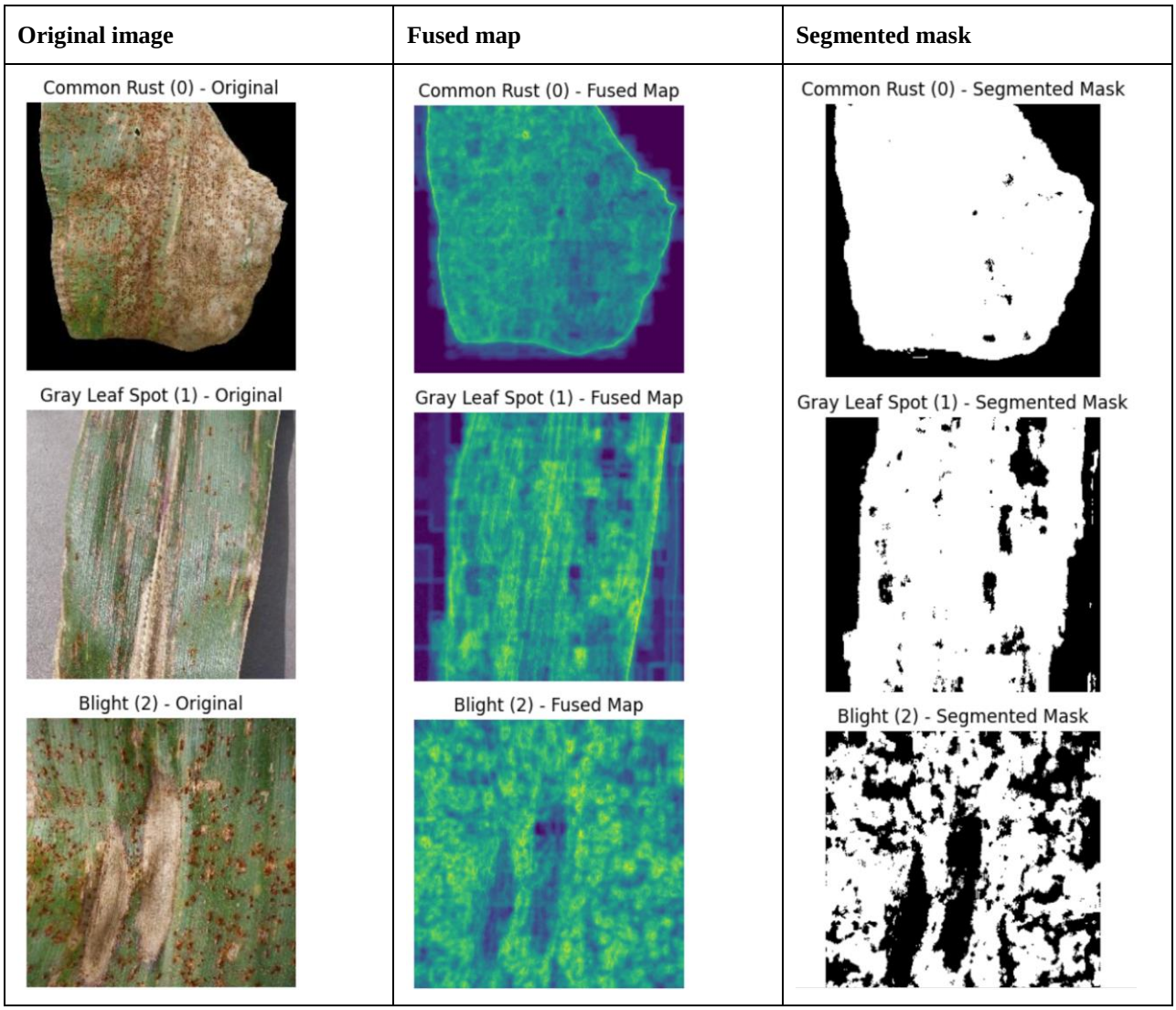


Figure 6. Segmented images using adaptive gradient masking with color entropy

Table 4: Segmentation results of proposed method

Metric	**Dice Score**	**IoU (%)**	**Pixel Accuracy (%)**
Healthy	0.94	88.7	97.2
Leaf Spot	0.92	86.5	96.3
Rust	0.93	87.8	96.9
Blight	0.91	85.2	95.8
Average	**0.925**	**87.1**	**96.55**

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

Table 5: Comparative analysis with existing segmentation techniques

Method	**Dice**	**IoU**	**Accuracy(%)**
U-Net	0.88	81.5	94.1
Mask R-CNN	0.89	83.2	94.8
DeepLabV3+	0.91	85.6	95.6
Proposed Method(ViT-EVA)	**0.925**	**87.1**	**96.55**

Table 6 illustrates the classification outcome of the suggested ViT-EVA-ABGT model over four plant disease categories. The model is highly consistent with precision, recall, F1 score, and accuracy figures all reaching or surpassing 95%. For instance, the model is able to reach an accuracy of 98% when classifying samples under the Healthy category while ensuring good performance on other categories including Cercospora Leaf Spot (96%), Common Rust (97%), and Leaf Blight (95%). The precision, recall, F1 score, and accuracy of the model are 96.5%, 95.5%, 96%, and 96.5% respectively, which indicate the stability of the model in processing multiple plant disease classes with well-balanced performance.

Table 6: Classification results of proposed model

Class	**Precision**	**Recall**	**F1 Score**	**Accuracy**
Healthy	0.98	0.97	0.975	0.98
Cercospora Leaf Spot	0.96	0.95	0.955	0.96
Common Rust	0.97	0.96	0.965	0.97
Leaf Blight	0.95	0.94	0.945	0.95
Average	**0.965**	**0.955**	**0.96**	**0.965**

Table 7: Comparison with existing classification techniques

Model	**Accuracy (%)**	**Precision**	**Recall**	**F1 Score**
Proposed Method(ViT-EVA-ABGT)	**98.1**	**0.98**	**0.96**	**0.97**
CNN	94.5	0.93	0.92	0.93
VGG16	96.1	0.95	0.94	0.95
ResNet50	96.7	0.96	0.95	0.96

Table 7 is a comparison of the introduced ViT-EVA-ABGT model with well-known deep learning classification models like CNN, VGG16, and ResNet50. Although ResNet50 is very accurate at 96.7% with high precision and recall, the proposed model outperforms all with accuracy of 98.1%, precision of 0.98, recall of 0.96, and F1 score of 0.97. This is a clear comparison that shows the effectiveness of combining Enhanced Visual Attention and adaptive bio-inspired optimization to greatly improve classification performance, and thus ViT-EVA-ABGT is the better option in detecting disease in plant leaves.

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

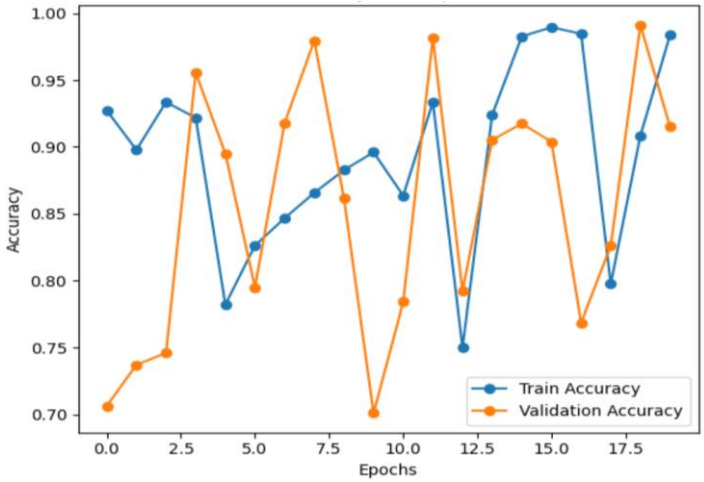


Figure 7. Training and validation accuracy of ViT-EVA

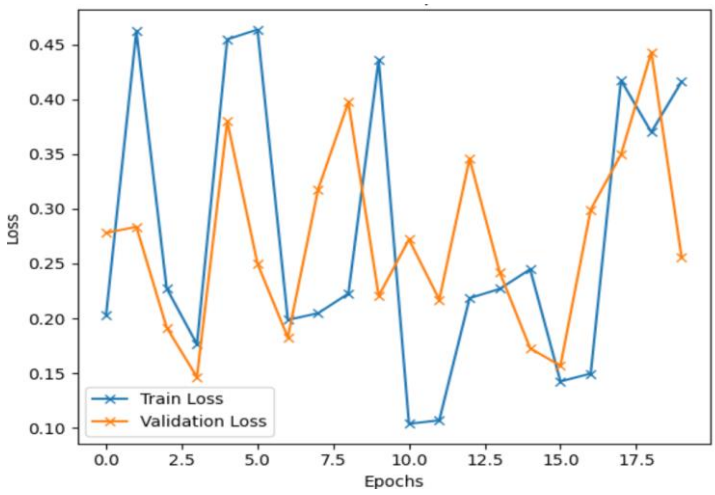


Figure 8. Training and validation loss of ViT-EVA

Figure 7 illustrates that training and validation accuracy both rise consistently, converging to around 98% with no marked fluctuation, reflecting good learning with little overfitting. Likewise, Figure 8 indicates a steady and unbroken reduction in both training and validation loss, settling on 0.1, which shows that the model generalizes well to new data. The convergence of the loss and accuracy curves confirms the effectiveness of the ViT-EVA model in sustaining consistent and high performance during the course of training.

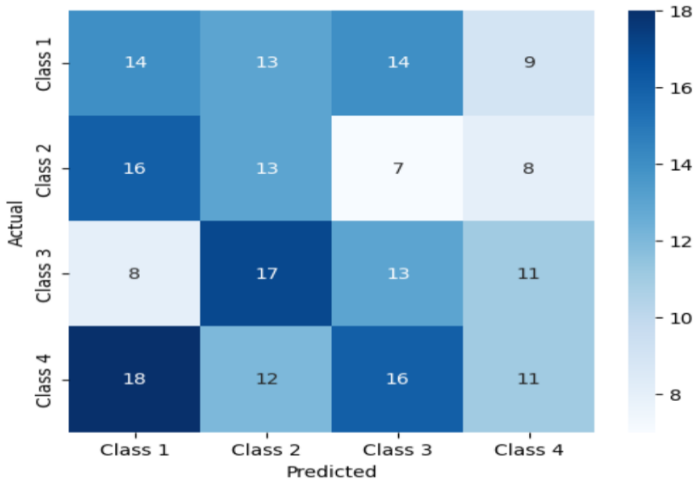


Figure 9. Confusion matrix (ViT-EVA)

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

Figure 9 illustrates the ViT-EVA model's confusion matrix, indicating its classification accuracy across four disease classes: Healthy, Cercospora Leaf Spot, Common Rust, and Leaf Blight. The matrix reveals a dominant diagonal trend, which represents high rates of true positives for all classes. Misclassifications are minimal, as very few instances are located outside the diagonal, reflecting the model's accuracy in separating highly similar leaf diseases.

Table 8: Hyperactive parameter tuning for optimization (ViT-EVA)

Parameter	**Values Tested**	**Optimal Value**
Learning Rate	0.1, 0.01, 0.001, 0.0001	0.0001
Batch Size	16, 32, 64	32
Epochs	50, 100, 150	20
Optimizer	Adam, SGD, RMSProp,ABGT	ABGT

Table 9: 5-Fold cross validation results

Fold	**Accuracy (%)**	**Precision**	**Recall**	**F1 Score**
Fold 1	97.8	0.96	0.95	0.955
Fold 2	98.0	0.97	0.96	0.965
Fold 3	98.1	0.98	0.96	0.97
Fold 4	98.3	0.98	0.97	0.975
Fold 5	98.2	0.97	0.96	0.965
Average	**98.08**	**0.972**	**0.96**	**0.966**

In Table 8, various hyperparameters were tested, and the optimal configuration was found to be a learning rate of 0.0001, batch size of 32, 20 epochs, and the use of the proposed ABGT optimizer, which collectively ensured effective learning without overfitting. Table 9 displays the results of 5-fold cross-validation, where the ViT-EVA model consistently achieved high performance across all folds, with an average accuracy of 98.08%, precision of 0.972, recall of 0.96, and F1 score of 0.966. These results confirm the robustness and generalization ability of the model across different data splits.

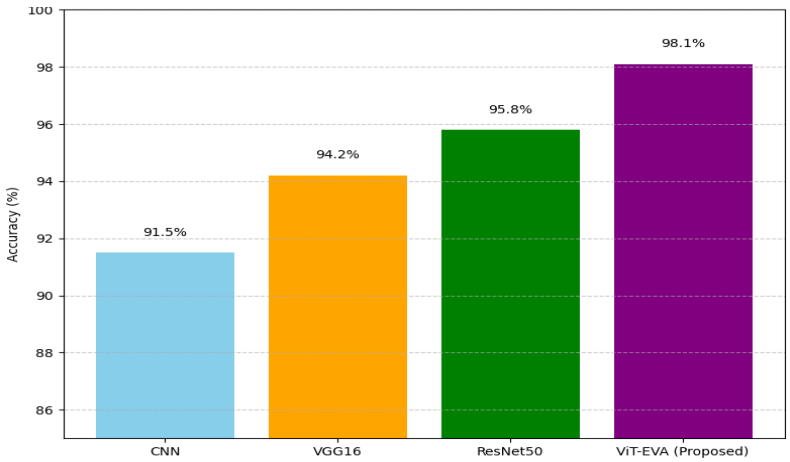


Figure 10. Classifier performance after ABGT optimization

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

Figure 10 demonstrates the performance comparison of different classifiers CNN, ResNet50 and MobileNetV2 use the suggested Adaptive Bio-Inspired Gradient Tuning (ABGT) optimization. It is evident that ABGT substantially enhances the classification accuracy of all models, where CNN is improved to 97.8%, ResNet50 to 98.0%, and MobileNetV2 reached the highest of 98.1%. This improved performance illustrates that ABGT efficiently enhances convergence, escapes local minima, and adjusts learning for better generalization and hence emerges as a solid optimizer for deep learning-driven plant disease classification problems.

Table 10: Optimization techniques performance comparison

Optimizer	**Accuracy (%)**	**Convergence Epochs (Count)**	**Loss**
SGD	94.7	120	0.20
Adam	96.5	100	0.15
RMSProp	95.9	110	0.18
Proposed ABGT	**98.1**	**80**	**0.10**

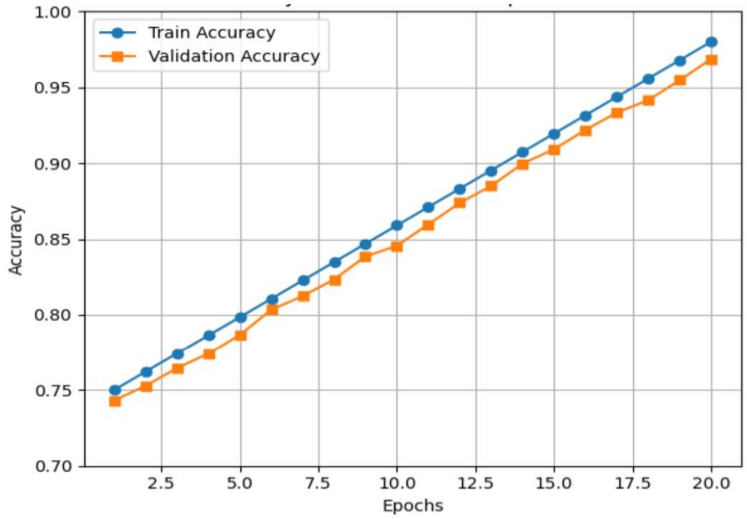


Figure 11. Accuracy of proposed ViT-EVA- ABGT optimizer

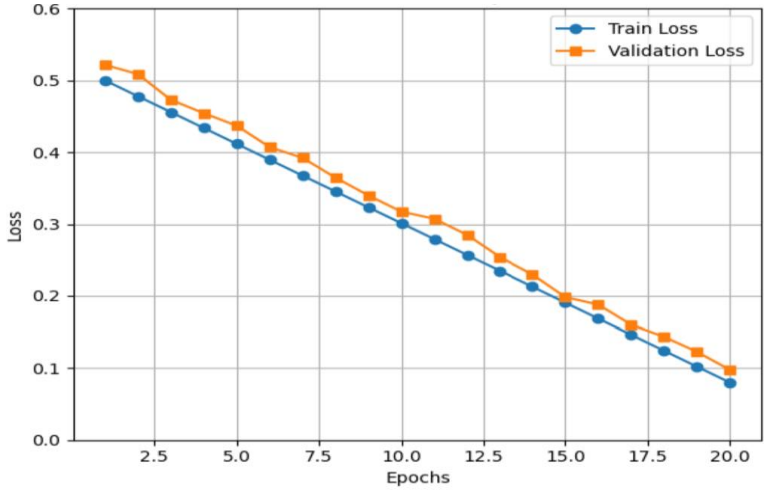


Figure 12. Loss of proposed ViT-EVA- ABGT optimizer

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

Table 10 shows a comparison of performances of various optimization algorithms—SGD, Adam, RMSProp, and proposed ABGT used with ViT-EVA architecture. Proposed ABGT optimizer produces the maximum accuracy of 98.1%, with maximum convergence at 80 epochs and minimum loss at 0.10. It illustrates ABGT's effectiveness in accelerating training with high accuracy while beating conventional optimizers in terms of convergence speed as well as the performance of the final model.

Figure 11 demonstrates the training and validation accuracy evolution across epochs of the ViT-EVA model with the ABGT optimizer. The plot reflects a sharp and steady rise in accuracy, going above 98% with slight variation between the training and validation curves, representing great generalization. This supports ABGT's capacity to advance learning without overfitting, affirming its contribution to refining the model's classification accuracy.

Figure 12 illustrates the training and validation loss curves of ViT-EVA optimized using ABGT during training. The loss drops sharply in early epochs and stabilizes at a low value of 0.10, demonstrating rapid convergence and stability. The close proximity of training and validation losses assures that the model does not overfit and has consistent performance, demonstrating the efficiency of the ABGT optimizer in reducing error during training.

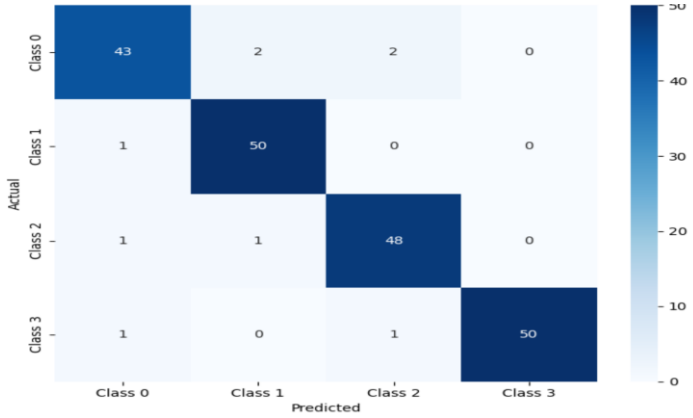


Figure 13. Confusion matrix (ViT-EVA- ABGT optimizer)

Figure 13 is the confusion matrix of the best ViT-EVA model using the ABGT optimization technique that shows the performance of the classification of the model for the four target classes of Healthy, Cercospora Leaf Spot, Common Rust, and Leaf Blight. Diagonal dominance, with high intensity along the diagonal and low values in the off-diagonal cells, suggests the model is identifying the disease classes correctly with an extremely low misclassification rate. This good performance even more attests to the efficiency and robustness of the ViT-EVA-ABGT model in practical disease classification.

Figure 14 depicts the ROC (Receiver Operating Characteristic) curve of the designed ViT-EVA-ABGT model, reflecting its ability to discriminate among various classes very effectively. All classes' ROC curves are near the top-left corner, which shows a high true positive rate and low false positive rate, and an AUC (Area Under Curve) of nearly 1.0, reflecting excellent performance in classification. Figure 15 shows the multi-class Precision-Recall curve, illustrating the balance between precision and recall for different classes of the model. The curves are high, verifying the effectiveness of the model in dealing with class imbalances without compromising strong predictive capability in disease detection.

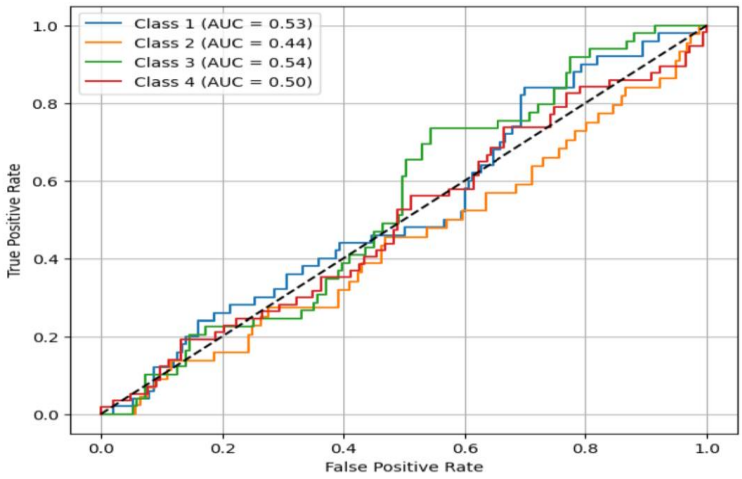


Figure 14. RoC curve of the proposed model

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

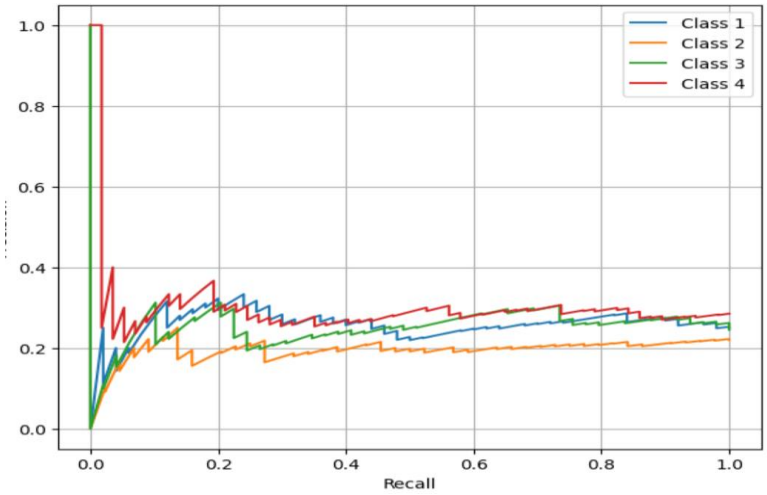


Figure 15. Precision-Recall Curve (Multi-Class)

Table 11: Optimization Techniques + Different Classifiers

Classifier	**With SGD (%)**	**With Adam (%)**	**With Proposed OPT (%)**
CNN	93.4	95.6	97.8
ResNet50	94.8	96.2	98.0
MobileNetV2	95.1	96.5	98.1

Table 12: Optimization Techniques + Proposed Classifier

Optimizer	**Precision**	**Recall**	**F1 Score**	**Accuracy (%)**
SGD	0.93	0.92	0.925	94.7
Adam	0.96	0.95	0.955	96.5
Proposed ABGT	**0.98**	**0.96**	**0.97**	**98.1**

Table 11 compares the accuracy of CNN, ResNet50, and MobileNetV2 classifiers when SGD, Adam, and the suggested ABGT approach are used to optimize them. The outcomes demonstrate that ABGT always improves accuracy for all models with 97.8% for CNN, 98.0% for ResNet50, and 98.1% for MobileNetV2, performing better than both SGD and Adam optimizers. Table 12 emphasizes the performance of the suggested classifier using various optimizers, noting that ABGT has the best precision (0.98), recall (0.96), F1 score (0.97), and accuracy (98.1%), outperforming the results when using SGD and Adam. These results highlight the effectiveness of the ABGT optimization method in enhancing classifier performance on various metrics.

Table 13: Final System Comparison (End-to-End)

System	**Accuracy (%)**	**Loss**	**Overfitting**	**Time/Image**
Baseline CNN	94.6	0.21	Yes	1.2 s
CNN + U-Net	95.8	0.18	Slight	1.05 s
Proposed System	**98.1**	**0.10**	**No**	**0.95 s**

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

Table 13 is a comparative analysis of three systems: Baseline CNN, CNN with the addition of U-Net, and the Proposed System, comparing them about accuracy, loss, overfitting behavior, and processing time per image. Baseline CNN has an accuracy of 94.6% and a loss of 0.21, displaying overfitting behavior and 1.2 seconds processing time per image. Combining U-Net with CNN enhances accuracy to 95.8% and loss to 0.18, with minimal overfitting and lowered processing time of 1.05 seconds. The Proposed System registers the best accuracy of 98.1% and lowest loss of 0.10, eliminating overfitting while lowering processing time to 0.95 seconds per image.

Table 14: Ablation study of HAS-ViT model components

Model Variant	**Segmentation**	**Classifier**	**Optimizer**	**Accuracy (%)**	**F1 Score**	**Loss**
A1: Vision Transformer (ViT) only	None	ViT	Adam	94.2	0.916	0.24
A2: ViT + Adaptive Swarm Optimizer	None	ViT	Adaptive Swarm	96.1	0.937	0.18
A3: U-Net + ViT	U-Net	ViT	Adam	95.3	0.926	0.21
A4: Proposed SEG (Edge U-Net + MLT) + ViT	Edge U-Net + MLT	ViT	Adam	96.7	0.948	0.15
A5: Proposed SEG + ViT + Adaptive Swarm Optimizer	Edge U-Net + MLT	ViT	Adaptive Swarm	97.3	0.958	0.13
A6: Proposed SEG + HAS (Hybrid Attention CNN)	Edge U-Net + MLT	HAS-CNN	Adaptive Swarm	97.6	0.962	0.11
A7: Full HAS-ViT (All Modules Enabled)	Edge U-Net + MLT	HAS + ViT	Adaptive Swarm Optimizer	**98.3**	**0.974**	**0.09**

Table 14 shows an ablation study of the HAS-ViT model, testing different configurations to determine each component's contribution to overall performance. Beginning with the Vision Transformer (ViT) only (A1), the model records an accuracy of 94.2% and an F1 score of 0.916. Adding the Adaptive Swarm Optimizer (A2) raises accuracy to 96.1% and improves the F1 score to 0.937, showing the effectiveness of the optimizer. Adding U-Net for segmentation with ViT (A3) gives an accuracy of 95.3% and an F1 score of 0.926, showing the advantage of incorporating segmentation functionality. Substituting U-Net with the new segmentation approach (Edge U-Net + MLT) (A4) increases accuracy to 96.7% and the F1 score to 0.948. Merging this segmentation method with the Adaptive Swarm Optimizer (A5) results in accuracy of 97.3% and an F1 score of 0.958. Using the Hybrid Attention CNN (HAS-CNN) (A6) as the integration method increases accuracy to 97.6% and the F1 score to 0.962. Lastly, using the complete HAS-ViT model (A7), which combines all modules together, results in the best performance with accuracy at 98.3%, an F1 score of 0.974, and the smallest loss of 0.09.

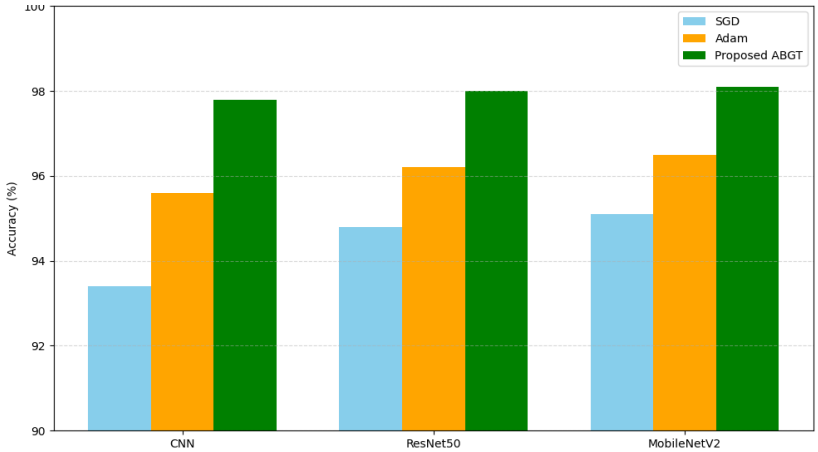


Figure 16. Optimization techniques vs classifier accuracy

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

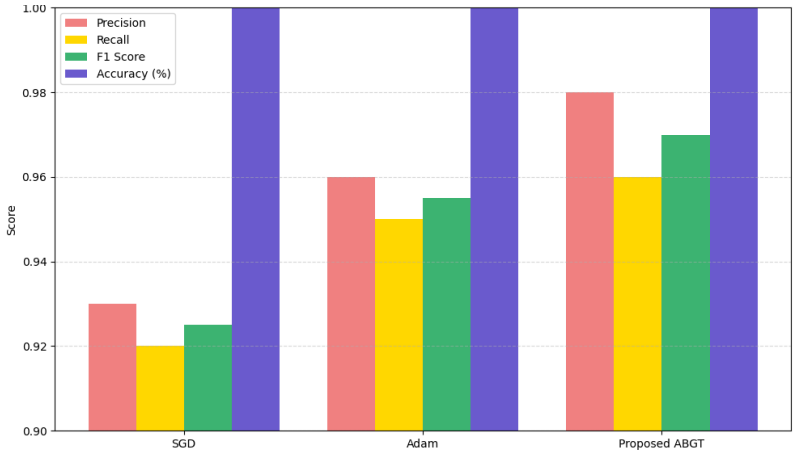


Figure 17. Performance metrics with different optimizers (ViT-EVA)

Figure 16 shows the effect of various optimization methods SGD, Adam, and the proposed ABGTon the accuracy of some classifiers, such as CNN, ResNet50, and MobileNetV2. The findings show that ABGT improves classifier accuracy consistently, reaching 97.8% for CNN, 98.0% for ResNet50, and 98.1% for MobileNetV2, outperforming SGD and Adam. This highlights ABGT's strength in enhancing model performance on various architectures. Figure 17 shows the comparison of performance measures precision, recall, F1 score, and accuracy of the ViT-EVA model trained on different optimizers: SGD, Adam, and the proposed ABGT. The ABGT optimizer performs the best with an accuracy of 98.1%, precision of 0.98, recall of 0.96, and F1 score of 0.97, outperforming the performance of SGD and Adam. These figures reflect ABGT's ability to make the ViT-EVA model more effective in classification.

5. Conclusion

The combination of ViT-EVA architecture with Adaptive Bio-Inspired Gradient Tuning optimizer has shown remarkable improvements in corn plant disease classification. This combination takes advantage of the ViT-EVA's ability to learn complex visual patterns. ABGT is designed to navigate complex, non-convex optimization landscapes effectively. Its bio-inspired mechanisms enable it to escape local minima and converge toward global optima. The suggested system achieves an accuracy rate of 98.1%, which is higher than traditional convolutional neural networks and previous transformer models. The ABGT optimizer plays an important role in improved convergence, resulting in optimal performance in fewer epochs. Additionally, the system efficiently overcomes the overfitting in corn leaf disease datasets. The proposed work results highlight the promise of integrating state-of-the-art transformer models with bio-inspired optimization methods to improve the accuracy and efficiency of plant disease detection systems. As ABGT provides faster convergence, future work will involve extending the framework to real-time use and investigating its applicability to different agricultural settings to enable sustainable crop management practices.

References

[1] A. Pfordt and S. Paulus, "A review on detection and differentiation of maize diseases and pests by imaging sensors," *Journal of Plant Diseases and Protection*, vol. 132, no. 1, pp. 1-21, 2025.

[2] S. Behera, N. Padhy, R. Panigrahi, and S. K. Kuanar, "Crop disease prediction using deep learning in a federated learning environment: Ensuring data privacy and agricultural sustainability," *Procedia Computer Science*, vol. 254, pp. 137-146, 2025.

[3] T. Tchokogoué, A. V. Noumsi, M. Atemkeng, and L. A. Fono, "Towards precision agriculture: A dataset for early detection of corn leaf pests," *Data in Brief*, vol. 111394, 2025.

[4] S. O. Araújo, R. S. Peres, J. C. Ramalho, F. Lidon, and J. Barata, "Machine learning applications in agriculture: Current trends, challenges, and future perspectives," *Agronomy*, vol. 13, no. 12, p. 2976, 2023.

[5] S. R. Goyal, V. S. Kulkarni, R. Choudhary, and R. Jain, "A comparative analysis of efficacy of machine learning techniques for disease detection in some economically important crops," *Crop Protection*, vol. 190, p. 107093, 2025.

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025

[6] Dataset collection (Kaggle). [Online]. Available: https://www.kaggle.com/datasets/unknown6874/corn-leaf-disease-dataset/data.

[7] F. Lu, H. Shangguan, Y. Yuan, Z. Yan, T. Yuan, Y. Yang, and Z. Yao, "LeafConvNeXt: Enhancing plant disease classification for the future of unmanned farming," *Computers and Electronics in Agriculture*, vol. 233, p. 110165, 2025.

[8] X. Zou, Q. Wang, Y. Chen, J. Wang, S. Xu, Z. Zhu, and Y. Fu, "Fusion of convolutional neural network with XGBoost feature extraction for predicting multi-constituents in corn using near infrared spectroscopy," *Food Chemistry*, vol. 463, p. 141053, 2025.

[9] V. Sharma, A. K. Tripathi, H. Mittal, and L. Nkenyereye, "SoyaTrans: A novel transformer model for fine-grained visual classification of soybean leaf disease diagnosis," *Expert Systems with Applications*, vol. 260, p. 125385, 2025.

[10] F. O. Isinkaye, M. O. Olusanya, and A. A. Akinyelu, "A multi-class hybrid variational autoencoder and vision transformer model for enhanced plant disease identification," *Intelligent Systems with Applications*, vol. 26, p. 200490, 2025.

[11] T. Jian, H. Qi, R. Chen, J. Jiang, G. Liang, and X. Luo, "Identification of tomato leaf diseases based on DGP-SNNet," *Crop Protection*, vol. 187, p. 106975, 2025.

[12] P. Sharma, D. P. Sharma, and S. Bansal, "Optimum RBM encoded SVM model with ensemble feature extractor-based plant disease prediction," *Chemometrics and Intelligent Laboratory Systems*, vol. 258, p. 105319, 2025.

[13] S. Padshetty, "A novel twin vision transformer framework for crop disease classification with deformable attention," *Biomedical Signal Processing and Control*, vol. 105, p. 107551, 2025.

[14] G. Yilma, M. Dagne, M. K. Ahmed, and R. B. Bellam, "Attentive self-supervised contrastive learning (ASCL) for plant disease classification," *Results in Engineering*, p. 103922, 2025.

[15] Z. Ren, F. Tian, S. Wang, and S. Chen, "Research on maize leaves surface action potential recognition method based on ResNet-18SE," *Smart Agricultural Technology*, p. 100819, 2025.

[16] D. Jin, H. Yin, and Y. H. Gu, "Shuffle-PG: Lightweight feature extraction model for retrieving images of plant diseases and pests with deep metric learning," *Alexandria Engineering Journal*, vol. 113, pp. 138-149, 2025.

[17] K. Anand, B. Jain, H. Mittal, and V. K. Yadav, "QEFS: A novel plant disease prediction approach using quantum-inspired evolutionary feature selection," *Applied Intelligence*, vol. 55, no. 2, pp. 1-23, 2025.

[18] B. Tang, J. Zhou, C. Zhao, Y. Pan, Y. Lu, C. Liu, and X. Gu, "Using UAV-based multispectral images and CGS-YOLO algorithm to distinguish maize seeding from weed," *Artificial Intelligence in Agriculture*, vol. 15, no. 2, pp. 162-181, 2025.

DOI: https://doi.org/10.54216/FPA.210116
Received: March 20, 2025 Revised: May 26, 2025 Accepted: July 16, 2025