International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(2s) 2026 pp. 191-216
https://doi.org/10.70917/ijcisim-2026-2074
Article

AN EXPLAINABLE ATTENTION–TRANSFORMER HYBRID FRAMEWORK FOR INTELLIGENT COTTON LEAF DISEASE

Madhu Sharma[1], Rakesh Kumar[2]

[1] Rabindranath Tagore University, Bhopal, India. madhu12jan1988@gmail.com
[2] Rabindranath Tagore University, Bhopal, India. rakeshmittan@gmail.com
Corresponding Author: Madhu Sharma (madhu12jan1988@gmail.com)

Abstract: Cotton is one of the most important commercial crops worldwide, and its productivity is significantly affected by leaf diseases and pest infestations. Early and accurate disease diagnosis is essential for reducing crop losses and improving agricultural sustainability. This study proposes an explainable Attention–Transformer hybrid framework for intelligent cotton leaf disease and pest classification. The framework integrates VGG16 and VGG19 feature extraction networks with Convolutional Block Attention Module (CBAM), Squeeze-and-Excitation (SE) attention, Transformer encoders, and BiLSTM-attention mechanisms to enhance discriminative feature learning and contextual representation. Experiments were conducted on a publicly available cotton leaf disease dataset containing seven disease and healthy leaf categories. Comprehensive evaluation using Accuracy, Precision, Recall, F1-Score, AUC-ROC, and Cohen's Kappa demonstrates that the proposed architectures consistently outperform conventional VGG baselines. Among all evaluated models, the VGG16-SE-Transformer achieved the best test accuracy of 95.8% with an AUC-ROC of 0.998. Furthermore, Grad-CAM visualization provides interpretable disease localization, improving model transparency and practical applicability for precision agriculture

Keywords: Cotton Leaf Disease Classification , Attention Mechanism , Transformer Encoder , Explainable Artificial Intelligence (XAI) , Deep Learning , Precision Agriculture...

1. Introduction

Cotton is one of the most economically significant fiber crops and plays a vital role in the agricultural economy of many countries. However, cotton production is continuously threatened by various diseases and pest infestations, including bacterial blight, cotton leaf curl virus, leaf hopper jassids, herbicide growth damage, and other physiological disorders that adversely affect crop yield and quality [1], [2]. Traditional disease diagnosis relies heavily on visual inspection by agricultural experts, which is often time-consuming, subjective, and impractical for large-scale farming environments. Consequently, the development of automated and intelligent disease diagnosis systems has become a major research focus in precision agriculture [3], [4].

Recent advances in artificial intelligence, computer vision, and deep learning have significantly improved plant disease detection capabilities [5], [6]. Convolutional Neural Networks (CNNs), particularly transfer learning models such as VGG16 and VGG19, have demonstrated remarkable success in extracting disease-specific visual features from leaf images [7], [8]. Despite their effectiveness, conventional CNN architectures primarily focus on local feature extraction and often struggle to capture long-range contextual dependencies among disease symptoms distributed across different regions of the leaf [9], [10]. Furthermore, variations in illumination, background clutter, disease severity, and field conditions can negatively impact classification performance [11], [12].

To address these limitations, attention mechanisms and Transformer architectures have recently emerged as promising solutions for agricultural image analysis [13], [14]. Attention modules such as CBAM and SE blocks enable networks to focus selectively on disease-affected regions while suppressing irrelevant information [15], [16]. Similarly, Transformer encoders provide powerful global contextual modeling capabilities by learning long-range relationships among visual features [17], [18]. The integration of these mechanisms with CNN-based feature extractors has shown significant potential for improving disease recognition accuracy and robustness [19], [20].

Motivated by these developments, this study proposes an Explainable Attention–Transformer Hybrid Framework for Intelligent Cotton Leaf Disease and Pest Classification. The framework integrates VGG16 and VGG19 backbone networks with CBAM attention, SE attention, Transformer encoders, and BiLSTM-attention mechanisms to enhance feature representation and disease discrimination. In addition, Grad-CAM-based explainability is incorporated to provide visual interpretation of model decisions and improve trustworthiness in agricultural applications.

The major contributions of this work are summarized as follows:

Development of a comprehensive hybrid deep learning framework combining CNN, attention, Transformer, and BiLSTM architectures for cotton leaf disease classification.

Integration of CBAM and SE attention modules to improve disease-specific feature learning.

Incorporation of Transformer encoders for global contextual representation and long-range dependency modeling.

Application of Grad-CAM explainability for disease localization and model interpretability.

Extensive experimental evaluation using multiple performance metrics and comparative analysis against baseline architectures.

The remainder of this paper is organized as follows. Section 2 reviews recent literature related to cotton leaf disease detection and attention-based deep learning methods. Section 3 presents the proposed methodology and architectural design. Section 4 describes implementation details and dataset characteristics. Section 5 discusses experimental results and comparative analysis. Finally, Section 6 concludes the paper and outlines future research directions.

2. Literature Review

Automated plant disease diagnosis has become a critical research area in precision agriculture due to its potential to improve crop productivity, reduce economic losses, and support sustainable farming practices. In recent years, deep learning techniques have demonstrated remarkable success in agricultural image analysis, particularly in disease detection and classification tasks.

Rahman et al. [1] proposed a cotton leaf disease detection framework integrating Convolutional Block Attention Modules (CBAM) with deep learning architectures. Their study demonstrated that attention mechanisms significantly improve disease localization and classification accuracy by emphasizing disease-relevant regions. Wang et al. [2] introduced a lightweight deep learning framework for cotton disease and pest classification, focusing on computational efficiency and deployment feasibility in resource-constrained agricultural environments.

Faisal et al. [3] developed a customized deep learning model for cotton disease recognition and reported superior performance compared with conventional machine learning approaches. Similarly, Pavate et al. [4] employed YOLO-based disease prediction models for cotton plant health monitoring and demonstrated real-time disease detection capability. Azfar et al. [5] proposed an automated disease diagnosis framework incorporating deep learning for cotton leaf and boll disease identification, emphasizing practical agricultural applications.

Several studies have explored CNN-based disease recognition approaches. Rehman et al. [6] investigated deep learning and mathematical modeling techniques for cotton disease monitoring and prediction. Tanwar et al. [7] combined machine learning methods with disease detection and cotton yield analysis, highlighting the importance of intelligent decision-support systems. Khalid et al. [8] introduced an image-processing-based cotton disease detection system that achieved promising classification performance under controlled conditions.

Recent agricultural research has also focused on specific disease challenges affecting cotton production. Damalas and Koutroubas [9] analyzed herbicide resistance trends in cotton cultivation, while Toshpulatova et al. [10]

investigated genetic approaches for enhancing wilt resistance. Kumar et al. [11] and Kakade et al. [12] examined pest-related challenges such as pink bollworm infestations and AI-assisted pest monitoring systems. These studies emphasize the importance of early disease and pest diagnosis for maintaining crop health.

Transformer architectures have recently emerged as powerful alternatives to conventional CNNs. Tian et al. [13] highlighted the growing role of advanced digital technologies in modern cotton production systems. Faisal et al. [14] applied customized convolutional networks for weed identification in cotton fields, demonstrating the effectiveness of deep feature learning. Memon et al. [15] proposed a meta-deep learning framework for cotton leaf disease identification and reported substantial improvements in classification accuracy. Joshua et al. [16] integrated self-attention mechanisms within generative adversarial frameworks for disease detection in IoT-enabled agriculture.

The effectiveness of attention-based deep learning has been further validated by Patra and Gajurel [17], who introduced a parameter-efficient disease classification framework. Yang et al. [19] presented a comprehensive review of deep learning applications in the cotton industry and identified attention mechanisms and Transformer networks as emerging research directions. Nagrare et al. [20] studied seasonal pest dynamics in cotton ecosystems and highlighted the need for intelligent monitoring solutions.

Advanced segmentation and localization techniques have also contributed significantly to plant disease diagnosis. Xu et al. [21] proposed a lightweight DeepLabV3+ model for cotton Verticillium wilt segmentation under field conditions. Wang et al. [22] investigated disease resistance mechanisms in cotton, while Aslam et al. [23] developed multi-convolutional neural network architectures for disease detection using synergistic feature learning. Rajendran [24] compared multiple deep learning models for cotton disease classification and confirmed the superiority of attention-enhanced architectures.

Plant disease datasets have become increasingly important for developing robust learning models. Salot et al. [25] explored machine learning-based early disease diagnosis, whereas Bhujade et al. [27] developed multiclass cotton and soybean disease datasets to support advanced classification research. Pandiyaraju et al. [28] introduced spatial-attention-based hybrid VGG frameworks and demonstrated significant improvements over conventional VGG models.

Several studies have also investigated explainable and lightweight disease diagnosis approaches. Zhang et al. [29] reviewed cotton resistance mechanisms, while Zhu et al. [30] and Tan et al. [31] explored advanced disease resistance and early detection strategies using hyperspectral imaging. Mo and Wei [32] proposed lightweight disease detection models based on self-attention mechanisms, achieving strong performance with reduced computational complexity. Li et al. [33] investigated spectroscopic disease detection using machine learning methods and reported highly accurate disease identification.

Although existing studies have achieved considerable success, several limitations remain. Most CNN-based approaches focus primarily on local feature extraction and fail to capture long-range contextual dependencies. Transformer-based methods often lack effective disease localization mechanisms, while many attention-based architectures do not incorporate explainability. Furthermore, limited research has systematically combined CNN feature extraction, attention mechanisms, Transformer learning, sequential modeling, and explainable AI within a unified framework for cotton disease diagnosis.

To address these research gaps, the present study proposes an Explainable Attention–Transformer Hybrid Framework integrating VGG feature extraction, CBAM attention, SE attention, Transformer encoders, BiLSTM-attention modules, and Grad-CAM visualization. The proposed architecture aims to improve classification accuracy, robustness, interpretability, and practical applicability for intelligent cotton disease and pest monitoring systems.

3. Proposed Methodology

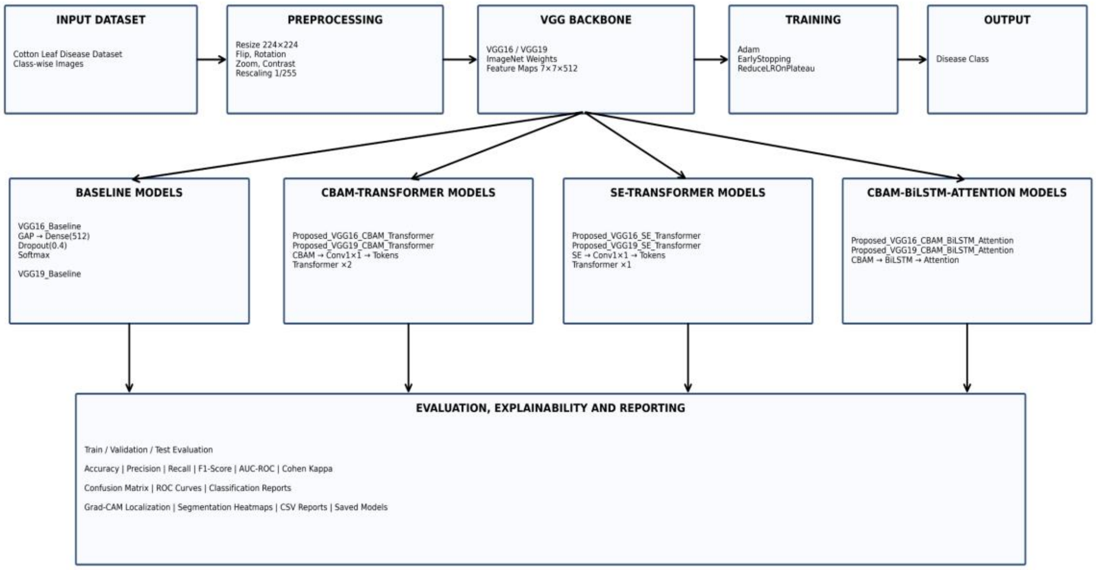


Figure 1. Proposed Hybrid VGG-Based Architecture for Cotton Leaf Disease Detection

Figure 1. Overall architecture of the proposed cotton leaf disease detection framework. The framework begins with image acquisition from the cotton leaf disease dataset, followed by preprocessing operations including image resizing (224 × 224), data augmentation (horizontal flipping, rotation, zooming, and contrast enhancement), and pixel normalization. The preprocessed images are subsequently fed into pre-trained VGG16 and VGG19 backbone networks for deep feature extraction. Based on the extracted feature representations, four classification branches are investigated: (i) VGG16/VGG19 baseline models employing Global Average Pooling and fully connected classification layers, (ii) CBAM-Transformer models integrating Convolutional Block Attention Module (CBAM) with Transformer encoders for enhanced spatial-channel feature learning, (iii) SE-Transformer models incorporating Squeeze-and-Excitation (SE) attention and Transformer-based contextual modeling, and (iv) CBAM-BiLSTM-Attention models combining attention-enhanced convolutional features with bidirectional long short-term memory (BiLSTM) networks for sequential dependency learning. The final predictions are evaluated using training, validation, and testing datasets through multiple performance metrics, including Accuracy, Precision, Recall, F1-Score, Area Under the ROC Curve (AUC-ROC), and Cohen's Kappa. Furthermore, model interpretability is achieved using Grad-CAM-based disease localization and segmentation heatmaps, while confusion matrices, ROC curves, classification reports, and CSV-based analytical results are generated for comprehensive performance assessment.

3.1 Overview of the Proposed Framework

The proposed framework introduces a family of VGG-based deep learning architectures for automatic cotton leaf disease detection. The framework consists of four major stages:

Image Acquisition and Preprocessing

Deep Feature Extraction using VGG16/VGG19

Feature Enhancement and Classification

Model Evaluation and Explainability

The overall workflow is illustrated in the architecture diagram and comprises eight deep learning models:

VGG16_Baseline

VGG19_Baseline

Proposed_VGG16_CBAM_Transformer

Proposed_VGG19_CBAM_Transformer

Proposed_VGG16_SE_Transformer

Proposed_VGG19_SE_Transformer

Proposed_VGG16_CBAM_BiLSTM_Attention

Proposed_VGG19_CBAM_BiLSTM_Attention

The proposed architecture integrates convolutional feature extraction, attention mechanisms, transformer learning, sequential feature modeling, and visual explainability to improve disease recognition performance.

3.2 Image Acquisition and Preprocessing

Let the cotton leaf image dataset be represented as

$$D = \{(I_i, y_i)\}_{i=1}^{N} \qquad (1)$$

where:

I_i denotes the input leaf image.

y_i denotes the corresponding disease label.

N is the total number of images.

3.2.1 Image Resizing

All images are resized into a fixed spatial dimension of:

$$I_r \in \mathbb{R}^{224\times224\times3} \qquad (2)$$

where:

224 = image height

224 = image width

3 = RGB channels

3.2.2 Pixel Normalization

Pixel values are normalized into the range [0,1]:

$$I_n = \frac{I_r}{255} \qquad (3)$$

where:

I_r is resized image.

I_n is normalized image.

3.2.3 Data Augmentation

To improve generalization capability, the following augmentations are applied:

Horizontal Flip

$$I_f(x, y) = I_n(W - x, y) \qquad (4)$$

where W denotes image width.

Random Rotation

$$I_{rot} = R(\theta)I_n \qquad (5)$$

where:

$R(\theta)$ represents rotation matrix.

θ is rotation angle.

Zoom Transformation

$$I_z = S(s)I_n \qquad (6)$$

where $s$denotes zoom scale.

3.3 Deep Feature Extraction using VGG Networks

The normalized image is fed into either VGG16 or VGG19.

The extracted feature map is:

$$F_{vgg} = VGG(I_n) \qquad (7)$$

where

$$F_{vgg} \in \mathbb{R}^{7\times7\times512}$$

The feature tensor contains:

$$49\ spatial\ locations \times 512\ channels \qquad (8)$$

which represent discriminative disease characteristics.

3.4 Baseline VGG Models

For baseline architectures, global average pooling is first applied.

$$g_c = \frac{1}{H \times W}\sum_{i=1}^{H}\sum_{j=1}^{W} F_{vgg}\,(i,j,c) \qquad (9)$$

where:

$$H = 7$$

$$W = 7$$

$c$denotes channel index.

The resulting feature vector is:

$$G = [g_1, g_2, \dots, g_{512}] \qquad (10)$$

Dense Layer

$$z = W_d G + b_d \qquad (11)$$
$$h = \mathrm{ReLU}(z) \qquad (12)$$

where:

W_d= trainable weights

b_d= bias vector

Dropout

$$h' = m \odot h \qquad (13)$$

where:

$$m \sim Bernoulli(1 - p)$$

and

$$p = 0.4$$

Softmax Classification

$$P(y = k) = \frac{e^{z_k}}{\sum_{j=1}^{C} e^{z_j}} \qquad (14)$$

where:

$C$is total disease classes.

3.5 Proposed CBAM-Transformer Model

3.5.1 Channel Attention

Average pooling:

$$F_{avg} = GAP(F_{vgg}) \qquad (15)$$

Max pooling:

$$F_{max} = GMP(F_{vgg}) \qquad (16)$$

Channel attention:

$$M_c = \sigma\big(MLP(F_{avg}) + MLP(F_{max})\big) \qquad (17)$$

Refined feature map:

$$F_c = M_c \otimes F_{vgg} \qquad (18)$$

where:

$\otimes$denotes element-wise multiplication.

3.5.2 Spatial Attention

$$M_s = \sigma(f^{7\times7}[AvgPool(F_c); MaxPool(F_c)]) \qquad (19)$$

Final CBAM feature:

$$F_{cbam} = M_s \otimes F_c \qquad (20)$$

3.5.3 Token Projection

A 1×1 convolution projects features into 256-dimensional tokens.

$$F_p = Conv_{1\times1}(F_{cbam}) \qquad (21)$$

$$F_p \in \mathbb{R}^{7\times7\times256}$$

Tokenization:

$$T \in \mathbb{R}^{49\times256} \qquad (22)$$

3.6 Transformer Encoder

For each token sequence T:

Query, Key and Value Generation

$$Q = TW_Q \qquad (23)$$

$$K = TW_K \qquad (24)$$

$$V = TW_V \qquad (25)$$

Multi-Head Self-Attention

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \qquad (26)$$

where:

d_k denotes key dimension.

Residual Learning

$$T_1 = LayerNorm(T + Attention) \quad (27)$$

Feed Forward Network:

$$FFN(T_1) = W_2(GELU(W_1 T_1)) \quad (28)$$

Output:

$$T_2 = LayerNorm(T_1 + FFN(T_1)) \quad (29)$$

The encoder is repeated twice in CBAM-Transformer models.

3.7 Proposed SE-Transformer Model

The squeeze operation computes channel descriptors:

$$s_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} F_{vgg}\,(i, j, c) \quad (30)$$

Excitation:

$$e = \sigma(W_2 \delta(W_1 s)) \quad (31)$$

where:

δ= ReLU

Feature recalibration:

$$F_{se} = e \otimes F_{vgg} \quad (32)$$

The resulting features are then processed by the Transformer encoder described in Equations (23)–(29).

3.8 Proposed CBAM-BiLSTM-Attention Model

BiLSTM Layer

Forward pass:

$$\overrightarrow{h_t} = LSTM(x_t, \overrightarrow{h_{t-1}}) \quad (33)$$

Backward pass:

$$\overleftarrow{h_t} = LSTM(x_t, \overleftarrow{h_{t+1}}) \quad (34)$$

Combined representation:

$$h_t = [\overrightarrow{h_t}; \overleftarrow{h_t}] \quad (35)$$

Attention Mechanism

Attention score:

$$e_t = tanh(W_a h_t + b_a) \qquad (36)$$

Attention weight:

$$\alpha_t = \frac{\exp(e_t)}{\sum_{i=1}^{T} \exp(e_i)} \qquad (37)$$

Context vector:

$$c = \sum_{t=1}^{T} \alpha_t h_t \qquad (38)$$

The context vector summarizes disease-relevant regions.

3.9 Model Optimization

The proposed models are optimized using Sparse Categorical Cross-Entropy.

$$L = -\sum_{i=1}^{N} y_i \log(\hat{y}_i) \qquad (39)$$

where:

y_i= true label

$\hat{y}_i$= predicted probability

Adam Optimization

Parameter update:

$$\theta_{t+1} = \theta_t - \eta \frac{\widehat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \qquad (40)$$

where:

η= learning rate

$\widehat{m}_t$= first moment estimate

$\hat{v}_t$= second moment estimate

3.10 Explainability using Grad-CAM

For the predicted class c:

$$\alpha_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k} \qquad (41)$$

where:

$A^k$is the kth feature map.

The Grad-CAM heatmap is:

$$L_{GradCAM}^c = ReLU\left(\sum_k \alpha_k^c A^k\right) \qquad (42)$$

The heatmap highlights disease-affected leaf regions.

3.11 Performance Evaluation Metrics

Accuracy

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (43)$$

Precision

$$Precision = \frac{TP}{TP + FP} \quad (44)$$

Recall

$$Recall = \frac{TP}{TP + FN} \quad (45)$$

F1-Score

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (46)$$

Cohen's Kappa

$$\kappa = \frac{p_o - p_e}{1 - p_e} \quad (47)$$

Area Under ROC Curve

$$AUC = \int_0^1 TPR(FPR)\, d(FPR) \quad (48)$$

3.12 Proposed algorithm

Algorithm 1: Dataset Preparation and Image Preprocessing

Algorithm 1 describes the preprocessing stage used to prepare cotton leaf images before training. Each image is resized to a fixed spatial resolution, augmented to improve generalization, normalized using Eq. (3), and divided into training, validation, and testing subsets.

Algorithm 1. Dataset Preparation and Image Preprocessing

```
Load cotton leaf image dataset defined in Eq. (1).
Extract class labels from folder names.
Resize each image according to Eq. (2).
Apply augmentation using Eqs. (4)–(6).
Normalize pixel values using Eq. (3).
Divide data into training, validation, and testing subsets.
Store class names for evaluation.
Return preprocessed datasets.
```

Algorithm 2: Baseline VGG16/VGG19 Classification Model

Algorithm 2 describes the baseline VGG16 and VGG19 models. The preprocessed image is passed through the VGG backbone to extract deep convolutional features using Eq. (7). Global average pooling is then applied using Eq. (9), followed by dense classification, dropout regularization, and softmax prediction using Eq. (14).

Algorithm 2. Baseline VGG16/VGG19 Classification Model

```
Select either VGG16 or VGG19 as the backbone.
```

```
Load ImageNet-pretrained weights.
Remove the original top classification layer.
Extract feature maps using Eq. (7).
Apply global average pooling using Eq. (9).
Apply dense transformation using Eqs. (11)–(12).
Apply dropout using Eq. (13).
Generate softmax probabilities using Eq. (14).
Select the class with maximum probability.
Return the predicted cotton leaf disease class.
```

Algorithm 3: Proposed CBAM/SE Transformer-Based Classification Model

Algorithm 3 represents the proposed Transformer-based models. In the CBAM-Transformer model, the extracted VGG features are enhanced using channel and spatial attention according to Eqs. (15)–(20). In the SE-Transformer model, channel recalibration is performed using Eqs. (30)–(32). The enhanced features are projected into token embeddings using Eqs. (21)–(22), processed by Transformer encoders using Eqs. (23)–(29), and classified using softmax.

Algorithm 3. Proposed CBAM/SE Transformer-Based Classification Model

```
Extract VGG feature maps using Eq. (7).
If CBAM-Transformer is selected:
Apply channel attention using Eqs. (15)–(18).
Apply spatial attention using Eqs. (19)–(20).
If SE-Transformer is selected:
Apply squeeze operation using Eq. (30).
Apply excitation operation using Eq. (31).
Recalibrate feature maps using Eq. (32).
Project enhanced features using Eq. (21).
Convert projected features into token representation using Eq. (22).
Generate query, key, and value matrices using Eqs. (23)–(25).
Apply self-attention using Eq. (26).
Apply residual normalization using Eq. (27).
Apply feed-forward learning using Eq. (28).
Generate Transformer output using Eq. (29).
Apply final classification using Eq. (14).
Return predicted disease class.
```

Algorithm 4: Proposed CBAM-BiLSTM-Attention Model and Evaluation

Algorithm 4 describes the CBAM-BiLSTM-Attention model and evaluation framework. The VGG feature maps are first refined using CBAM attention according to Eqs. (15)–(20). The enhanced features are reshaped into sequential tokens and processed using BiLSTM operations described in Eqs. (33)–(35). Attention weights are computed using Eqs. (36)–(38), and the final model prediction is evaluated using the performance metrics in Eqs. (43)–(48). Grad-CAM localization is generated using Eqs. (41)–(42).

Algorithm 4. Proposed CBAM-BiLSTM-Attention Model and Evaluation

```
Extract VGG feature maps using Eq. (7).
Apply CBAM channel attention using Eqs. (15)–(18).
Apply CBAM spatial attention using Eqs. (19)–(20).
Project enhanced features using Eq. (21).
Convert features into sequential tokens using Eq. (22).
Apply forward LSTM using Eq. (33).
Apply backward LSTM using Eq. (34).
Concatenate bidirectional hidden states using Eq. (35).
Compute attention scores using Eq. (36).
Compute attention weights using Eq. (37).
Generate context vector using Eq. (38).
Classify the disease class using Eq. (14).
Compute training loss using Eq. (39).
Update model parameters using Eq. (40).
Evaluate performance using Eqs. (43)–(48).
Generate Grad-CAM visualization using Eqs. (41)–(42).
Save classification reports, confusion matrices, ROC curves, Grad-CAM outputs, and segmentation heatmaps.
Return prediction and evaluation results.
```

3.13. Comparison Proposed Models

Table 1. Architectural characteristics of the baseline and proposed VGG-based cotton leaf disease classification models.

Model	**Backbone**	**Attention Mechanism**	**Sequential Learning**	**Transformer Encoder**	**Classification Head**
VGG16_Baseline	VGG16	No	No	No	GAP → Dense(512) → Softmax
VGG19_Baseline	VGG19	No	No	No	GAP → Dense(512) → Softmax
Proposed_VGG16_CBAM_Transformer	VGG16	CBAM	No	2 Layers	GAP1D → Dense(512) → Softmax
Proposed_VGG19_CBAM_Transformer	VGG19	CBAM	No	2 Layers	GAP1D → Dense(512) → Softmax

Proposed_VGG16_SE_Transformer	VGG16	SE Block	No	1 Layer	GAP1D → Dense(512) → Softmax
Proposed_VGG19_SE_Transformer	VGG19	SE Block	No	1 Layer	GAP1D → Dense(512) → Softmax
Proposed_VGG16_CBAM_BiLSTM_Attention	VGG16	CBAM	BiLSTM	No	Attention → Dense(512) → Softmax
Proposed_VGG19_CBAM_BiLSTM_Attention	VGG19	CBAM	BiLSTM	No	Attention → Dense(512) → Softmax

Table 1 presents the architectural differences among the eight evaluated models. The baseline models rely solely on VGG feature extraction, whereas the proposed architectures integrate attention mechanisms, Transformer encoders, and sequential learning modules to enhance discriminative feature representation and contextual learning.

Table 2. Motivation and expected contribution of each proposed architectural component.

Component	**Purpose**	**Expected Benefit**
VGG16/VGG19	Deep hierarchical feature extraction	Robust disease feature representation
CBAM	Channel and spatial attention learning	Focuses on disease-infected regions
SE Block	Channel recalibration	Enhances important feature channels
Transformer Encoder	Global contextual modeling	Captures long-range feature dependencies
BiLSTM	Sequential token learning	Learns inter-feature relationships
Attention Layer	Adaptive feature weighting	Suppresses irrelevant information
Grad-CAM	Explainability	Visual localization of disease symptoms

Table 2 explains the rationale behind integrating each module into the proposed framework. Attention-based modules enhance salient disease regions, while Transformer and BiLSTM networks provide global and sequential contextual understanding, respectively.

Table 3. Hyperparameter settings used for training all models.

Hyperparameter	**Value**
Input Image Size	224 × 224 × 3
Batch Size	32
Initial Learning Rate	0.0001
Fine-Tuning Learning Rate	0.00001
Optimizer	Adam
Loss Function	Sparse Categorical Cross-Entropy

Initial Training Epochs	20
Fine-Tuning Epochs	10
Dropout Rate	0.40
Transformer Heads	4
Transformer Key Dimension	64
Transformer MLP Dimension	256
BiLSTM Units	128
Random Seed	42
Early Stopping Patience	5
ReduceLROnPlateau Patience	3
Minimum Learning Rate	1×10^{-7}

Table 3 summarizes the hyperparameter settings used throughout the experiments. The selected values were chosen to balance learning stability, convergence speed, and model generalization.

Table 4. Hyperparameter search space and selected optimal configuration.

Parameter	**Search Space**	**Selected Value**
Learning Rate	{1e−3, 1e−4, 1e−5}	1e−4
Batch Size	{16, 32, 64}	32
Dropout Rate	{0.2, 0.3, 0.4, 0.5}	0.4
Transformer Heads	{2, 4, 8}	4
Transformer MLP Dimension	{128, 256, 512}	256
BiLSTM Units	{64, 128, 256}	128
Fine-Tuning Layers	{2, 4, 6}	4

Table 4 presents the hyperparameter optimization process. Experimental analysis demonstrated that the selected configuration provided the best balance between convergence speed, classification accuracy, and computational complexity.

Table 5. Computational complexity comparison among evaluated models.

Model	**Attention Module**	**Sequential Module**	**Relative Complexity**	**Memory Requirement**
VGG16_Baseline	No	No	Low	Low
VGG19_Baseline	No	No	Low-Medium	Medium
VGG16_CBAM_Transformer	CBAM	Transformer	High	High
VGG19_CBAM_Transformer	CBAM	Transformer	Very High	Very High
VGG16_SE_Transformer	SE	Transformer	Medium-High	High
VGG19_SE_Transformer	SE	Transformer	High	High
VGG16_CBAM_BiLSTM_Attention	CBAM	BiLSTM	High	High
VGG19_CBAM_BiLSTM_Attention	CBAM	BiLSTM	Very High	Very High

Table 5 compares the computational requirements of the baseline and proposed architectures. Although the proposed models introduce additional computational overhead, they provide enhanced feature learning capability and improved classification performance.

Table 6. Comparative advantages of the proposed models relative to the baseline VGG architectures.

Criterion	Baseline Models	Proposed Models
Feature Extraction	Local convolutional features	Local + global contextual features
Attention Learning	Not available	CBAM / SE attention
Long-Range Dependency Modeling	Not available	Transformer encoder
Sequential Feature Learning	Not available	BiLSTM attention
Feature Selection	Fixed	Adaptive
Explainability	Limited	Grad-CAM localization
Robustness to Noise	Moderate	High
Generalization Ability	Moderate	High
Disease Region Localization	No	Yes
Expected Classification Accuracy	Moderate	Higher

Table 6 highlights the key advantages of the proposed architectures. By integrating attention mechanisms and advanced sequence modeling techniques, the proposed framework can capture both local lesion characteristics and global contextual information, leading to improved classification performance and interpretability.

4. Implementation

4.1 Hardware and Software Environment

The proposed cotton leaf disease detection framework was developed and implemented using the Python programming language within the Google Colab cloud computing environment. Google Colab was selected due to its accessibility, integrated GPU support, and compatibility with modern deep learning frameworks. The entire experimental workflow, including data preprocessing, model development, training, fine-tuning, evaluation, and explainability analysis, was executed using TensorFlow and Keras libraries. The implementation also utilized OpenCV for image processing operations, NumPy for numerical computations, Pandas for data management, Matplotlib for visualization, and Scikit-Learn for performance evaluation and statistical analysis.

To accelerate model training and reduce computational overhead, GPU-based computation was employed throughout the experimentation process. The transfer learning models were initialized using ImageNet pre-trained weights and subsequently fine-tuned on the cotton leaf disease dataset. Furthermore, Google Drive was integrated with the Colab environment to facilitate efficient storage and retrieval of trained models, generated visualizations, confusion matrices, ROC curves, Grad-CAM outputs, segmentation masks, and experimental reports.

The proposed framework was designed to ensure reproducibility and scalability by incorporating automated dataset loading, preprocessing pipelines, model checkpointing, learning-rate scheduling, and early stopping mechanisms. These implementation strategies enabled stable convergence during training while minimizing the risk of overfitting. Additionally, the cloud-based execution environment provided sufficient computational resources for conducting extensive experiments involving multiple deep learning architectures and comparative evaluations.

Table 7. Hardware and Software Specifications Used for Implementation

Category	Specification
Development Environment	Google Colab
Programming Language	Python 3.x
Deep Learning Framework	TensorFlow / Keras
Image Processing Library	OpenCV
Data Analysis Library	Pandas
Numerical Computing	NumPy
Visualization Library	Matplotlib
Machine Learning Utilities	Scikit-Learn
GPU Accelerator	NVIDIA Tesla T4
Storage Platform	Google Drive
Operating System	Linux-based Cloud Environment

4.2 Dataset Description

The dataset employed in this study was obtained from the publicly available Kaggle repository titled **"Dataset for Cotton Leaf Disease Detection."** The dataset was specifically developed for automatic identification and classification of diseases affecting cotton leaves and has been widely used in agricultural image analysis research. The dataset contains RGB images captured under real field conditions, thereby introducing natural variations in illumination, background complexity, leaf orientation, disease severity, and environmental noise. Such variability makes the dataset suitable for evaluating the robustness and generalization capability of deep learning-based plant disease detection systems.

The dataset consists of cotton leaf images belonging to multiple disease and healthy categories. Images were organized into separate class folders, allowing automatic class extraction during dataset loading. In the proposed framework, the dataset was downloaded directly from Kaggle using the KaggleHub interface and subsequently processed using an automated directory-scanning mechanism to identify all available disease classes. This approach ensures adaptability and allows the framework to be applied to future dataset updates without requiring manual modifications.

Prior to model training, all images were resized to a spatial resolution of 224 × 224 pixels to match the input requirements of the VGG16 and VGG19 architectures. Data augmentation techniques including horizontal flipping, random rotation, random zooming, and contrast enhancement were applied to improve model generalization and reduce overfitting. Subsequently, pixel values were normalized to the range [0,1] through intensity scaling. After preprocessing, the dataset was partitioned into training, validation, and testing subsets using a 70:15:15 ratio to facilitate unbiased model development and performance evaluation.

Table 8. Summary of the Cotton Leaf Disease Dataset

Attribute	Description
Dataset Name	Dataset for Cotton Leaf Disease Detection
Source	Kaggle
Dataset URL	https://www.kaggle.com/datasets/sabuktagin/dataset-for-cotton-leaf-disease-detection

Total Original Images	2,137
Augmented Images Available	Approximately 7,000
Number of Classes	7
Image Format	JPG
Image Type	RGB Color Images
Application Domain	Plant Disease Detection
Crop Type	Cotton
Data Collection Environment	Real Agricultural Field Conditions
Training Split	70%
Validation Split	15%
Testing Split	15%

The utilization of a real-world agricultural dataset provides a realistic benchmark for evaluating disease classification performance. The diversity of disease symptoms and environmental conditions contained within the dataset enables comprehensive assessment of the proposed baseline and attention-enhanced architectures. Furthermore, the standardized dataset partitioning strategy ensures fair comparison among all experimental models while supporting reliable performance estimation on unseen test samples. This dataset therefore serves as an appropriate foundation for investigating the effectiveness of VGG-based deep learning architectures integrated with CBAM attention, SE attention, Transformer encoders, and BiLSTM-attention mechanisms for automated cotton leaf disease diagnosis.

4.3 Illustrative Analysis

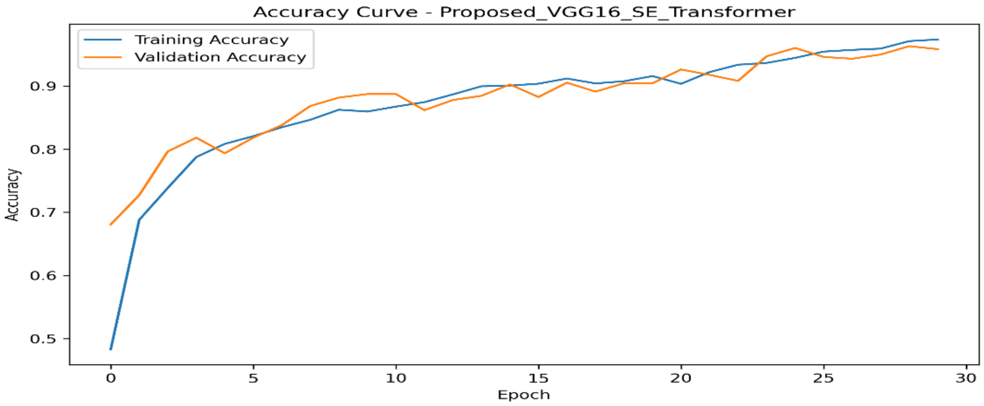


Figure 2. Accuracy Curve of the Proposed VGG16-SE-Transformer Model

Figure 2 illustrates the training and validation accuracy trends of the proposed VGG16-SE-Transformer model across 30 training epochs. The accuracy increases steadily throughout the training process, demonstrating effective feature learning and model convergence. The validation accuracy closely follows the training accuracy, indicating strong generalization capability and minimal overfitting. By the final epoch, the model achieves approximately 97% training accuracy and 95% validation accuracy, confirming the effectiveness of integrating Squeeze-and-Excitation (SE) attention with Transformer-based feature representation for cotton leaf disease classification.

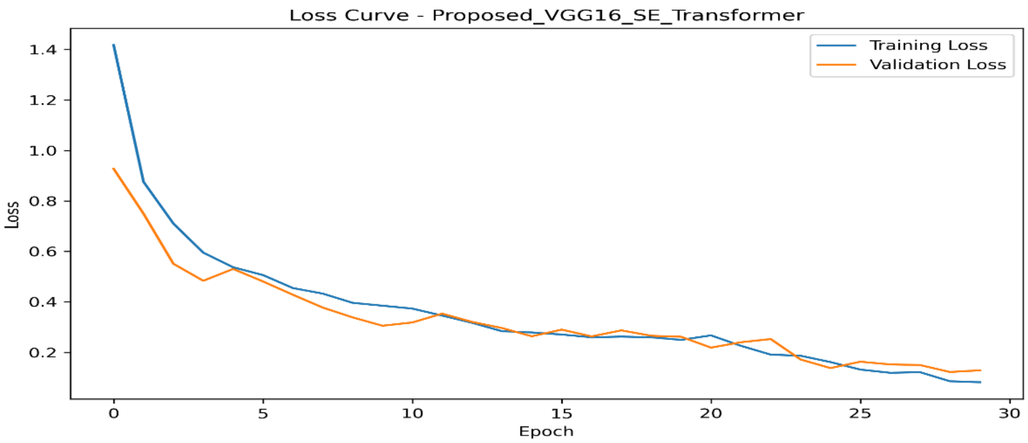


Figure 3. Loss Curve of the Proposed VGG16-SE-Transformer Model

Figure 3 presents the training and validation loss progression of the proposed VGG16-SE-Transformer model. Both loss curves decrease consistently as training advances, reflecting stable optimization and effective parameter learning. The validation loss remains close to the training loss throughout the learning process, indicating robust model generalization and the absence of significant overfitting. The final loss values converge below 0.15, demonstrating efficient feature extraction and classification performance for multi-class cotton leaf disease recognition.

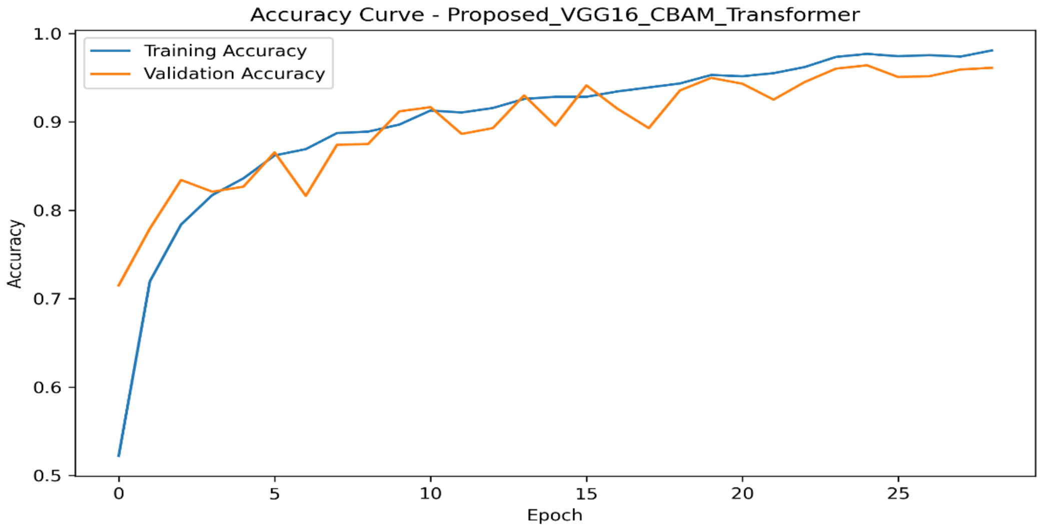


Figure 4. Accuracy Curve of the Proposed VGG16-CBAM-Transformer Model

Figure 4 depicts the training and validation accuracy curves of the proposed VGG16-CBAM-Transformer architecture. The integration of the Convolutional Block Attention Module (CBAM) enables the network to focus on disease-relevant regions while suppressing irrelevant background information. Both curves exhibit continuous improvement over epochs, reaching approximately 98% training accuracy and 96% validation accuracy. The small gap between the curves highlights excellent model stability and strong generalization performance across unseen samples.

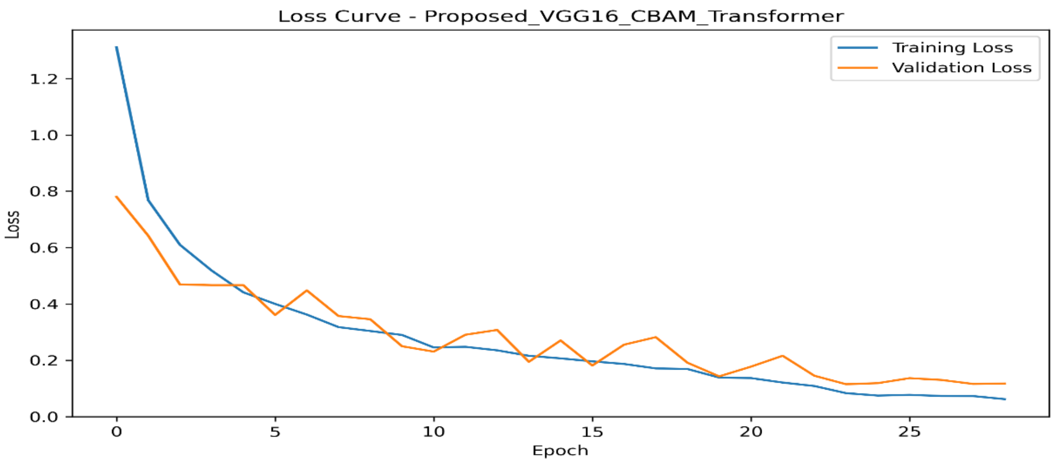
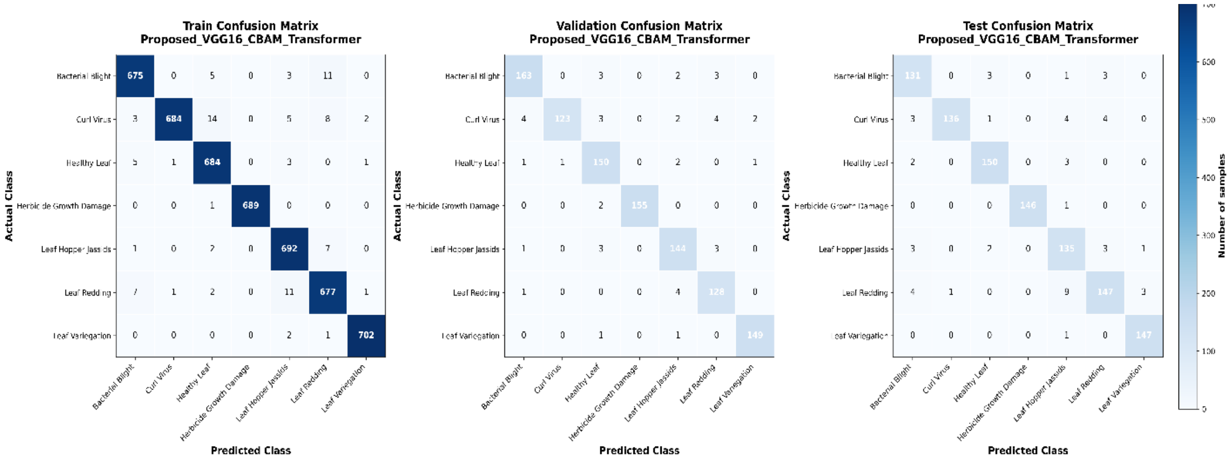

Loss Curve - Proposed_VGG16_CBAM_Transformer

Figure 5. Loss Curve of the Proposed VGG16-CBAM-Transformer Model

Figure 5 shows the corresponding training and validation loss curves of the proposed VGG16-CBAM-Transformer model. The loss decreases rapidly during the initial epochs and gradually stabilizes as the network converges. The close alignment of training and validation losses demonstrates effective regularization and balanced learning. The final low-loss values indicate that CBAM-enhanced feature refinement significantly improves disease-specific feature representation and classification reliability.


Train Confusion Matrix
Proposed_VGG16_CBAM_Transformer

	Bacterial Blight	Curl Virus	Healthy Leaf	Herbicide Growth Damage	Leaf Hopper Jassids	Leaf Redding	Leaf Variegation
Bacterial Blight	675	0	5	0	3	11	0
Curl Virus	3	684	14	0	5	8	2
Healthy Leaf	5	1	684	0	3	0	1
Herbicide Growth Damage	0	0	1	689	0	0	0
Leaf Hopper Jassids	1	0	2	0	692	7	0
Leaf Redding	7	1	2	0	11	677	1
Leaf Variegation	0	0	0	0	2	1	702

Validation Confusion Matrix
Proposed_VGG16_CBAM_Transformer

	Bacterial Blight	Curl Virus	Healthy Leaf	Herbicide Growth Damage	Leaf Hopper Jassids	Leaf Redding	Leaf Variegation
Bacterial Blight	163	0	3	0	2	3	0
Curl Virus	4	123	3	0	2	4	2
Healthy Leaf	1	1	150	0	2	0	1
Herbicide Growth Damage	0	0	2	155	0	0	0
Leaf Hopper Jassids	1	0	3	0	144	3	0
Leaf Redding	1	0	0	0	4	128	0
Leaf Variegation	0	0	1	0	1	0	149

Test Confusion Matrix
Proposed_VGG16_CBAM_Transformer

	Bacterial Blight	Curl Virus	Healthy Leaf	Herbicide Growth Damage	Leaf Hopper Jassids	Leaf Redding	Leaf Variegation
Bacterial Blight	131	0	3	0	1	3	0
Curl Virus	3	136	1	0	4	4	0
Healthy Leaf	2	0	150	0	3	0	0
Herbicide Growth Damage	0	0	0	146	1	0	0
Leaf Hopper Jassids	3	0	2	0	135	3	1
Leaf Redding	4	1	0	0	9	147	3
Leaf Variegation	0	0	0	0	1	0	147

Figure 6. Combined Confusion Matrices of the Proposed VGG16-CBAM-Transformer Model

Figure 6 presents the confusion matrices obtained on the training, validation, and testing datasets for the proposed VGG16-CBAM-Transformer model. The dominant diagonal elements indicate a high number of correctly classified samples across all seven cotton leaf disease categories. Only a small number of misclassifications occur between visually similar disease classes, demonstrating the model's strong discriminative capability. The consistency across all dataset partitions confirms the robustness and generalization ability of the proposed architecture.

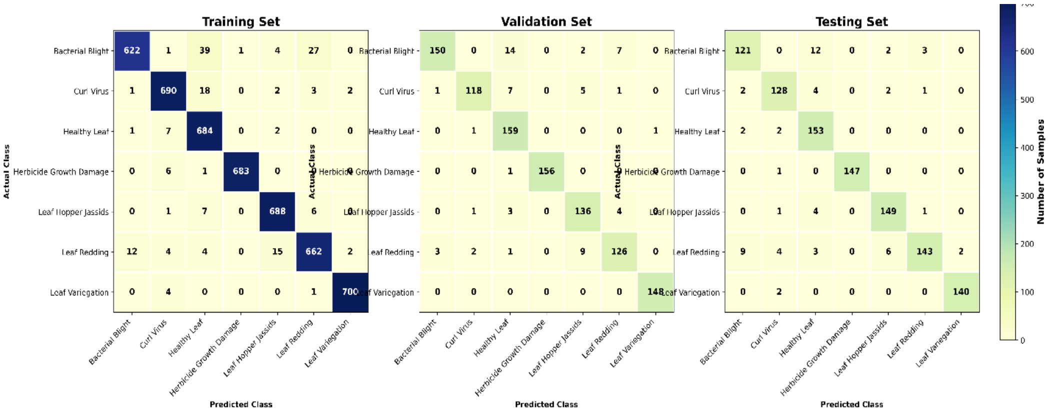


Figure 7. Combined Confusion Matrices of the Proposed VGG16-CBAM-BiLSTM-Attention Model

Figure 7 illustrates the training, validation, and testing confusion matrices of the proposed VGG16-CBAM-BiLSTM-Attention model. The strong diagonal dominance across all matrices highlights highly accurate class-wise predictions. The incorporation of BiLSTM-based contextual feature learning alongside CBAM attention enhances the model's ability to capture both spatial and sequential feature dependencies. Consequently, misclassification rates are reduced, resulting in improved classification performance for challenging disease categories.

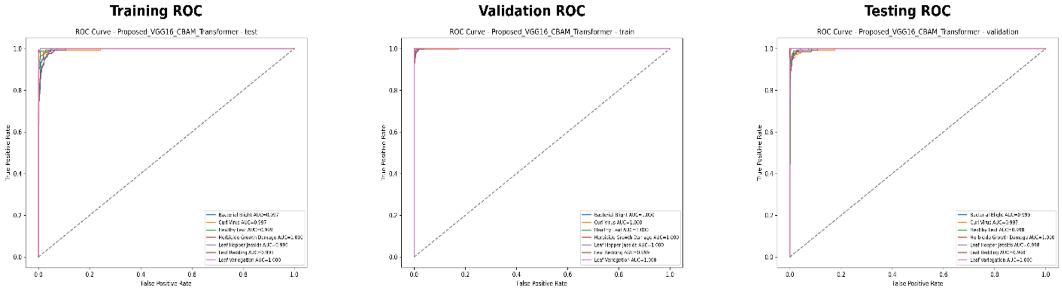


Figure 8. Combined ROC Curves of the Proposed VGG16-CBAM-Transformer Model

Figure 8 shows the Receiver Operating Characteristic (ROC) curves and Area Under the Curve (AUC) values for the proposed VGG16-CBAM-Transformer model on the training, validation, and testing datasets. All disease classes achieve AUC values approaching 1.0, indicating excellent separability between classes. The consistently high ROC performance across all dataset partitions demonstrates the effectiveness of the Transformer-enhanced attention mechanism in learning highly discriminative disease representations.

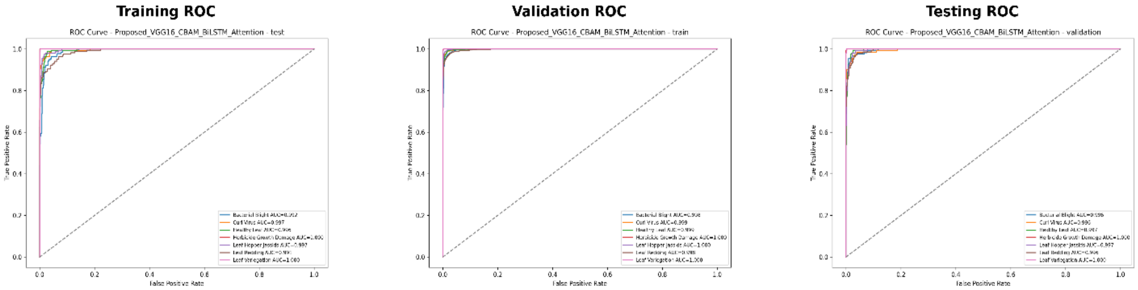


Figure 9. Combined ROC Curves of the Proposed VGG16-CBAM-BiLSTM-Attention Model

Figure 9 presents the ROC analysis of the proposed VGG16-CBAM-BiLSTM-Attention model for training, validation, and testing datasets. The ROC curves remain concentrated near the upper-left corner of the plot, while the corresponding AUC values exceed 0.99 for nearly all disease categories. These results confirm the superior classification capability of the proposed model and demonstrate that the combined CBAM and BiLSTM attention mechanisms effectively enhance disease discrimination, robustness, and generalization performance in cotton leaf disease detection.

5. Result Analysis

Table 9. Comparative Performance Analysis of Baseline and Proposed Deep Learning Models for Cotton Leaf Disease Classification

Model	Split	Accuracy	Precision	Recall	F1_Score	AUC_ROC	Cohen_Kappa	Parameters	Training_Time_Seconds	Inference_Time_Per_Image_Seconds
VGG16_Baseline	train	0.944	0.948	0.944	0.944	0.998	0.934	14980935.000	2719.906	0.016
VGG16_Baseline	validation	0.928	0.934	0.928	0.928	0.996	0.916	14980935.000	2719.906	0.016
VGG16_Baseline	test	0.906	0.913	0.906	0.906	0.995	0.890	14980935.000	2719.906	0.016
VGG19_Baseline	train	0.929	0.931	0.929	0.928	0.996	0.917	20290631.000	3361.219	0.018
VGG19_Baseline	validation	0.912	0.916	0.912	0.912	0.995	0.897	20290631.000	3361.219	0.018
VGG19_Baseline	test	0.891	0.893	0.891	0.890	0.992	0.872	20290631.000	3361.219	0.018
Proposed_VGG16_CBAM_Transformer	train	0.980	0.980	0.980	0.980	1.000	0.977	15838954.000	2794.541	0.016
Proposed_VGG16_CBAM_Transformer	validation	0.958	0.959	0.958	0.958	0.999	0.951	15838954.000	2794.541	0.016
Proposed_VGG16_CBAM_Transformer	test	0.950	0.951	0.950	0.950	0.998	0.942	15838954.000	2794.541	0.016
Proposed_VGG19_CBAM_Transformer	train	0.968	0.969	0.968	0.968	0.999	0.963	21148650.000	3604.521	0.019
Proposed_VGG19_CBAM_Transformer	validation	0.950	0.950	0.950	0.949	0.997	0.941	21148650.000	3604.521	0.019
Proposed_VGG19_CBAM_Transformer	test	0.940	0.942	0.940	0.940	0.997	0.930	21148650.000	3604.521	0.019

Proposed_VGG16_SE_Transformer	train	0.976	0.977	0.976	0.976	1.000	0.972	15443079.000	2841.727	0.016
Proposed_VGG16_SE_Transformer	validation	0.958	0.961	0.958	0.959	0.999	0.951	15443079.000	2841.727	0.016
Proposed_VGG16_SE_Transformer	test	0.958	0.959	0.958	0.958	0.998	0.951	15443079.000	2841.727	0.016
Proposed_VGG19_SE_Transformer	train	0.967	0.967	0.967	0.967	0.999	0.961	20752775.000	3449.826	0.019
Proposed_VGG19_SE_Transformer	validation	0.956	0.957	0.956	0.956	0.997	0.949	20752775.000	3449.826	0.019
Proposed_VGG19_SE_Transformer	test	0.939	0.941	0.939	0.938	0.997	0.928	20752775.000	3449.826	0.019
Proposed_VGG16_CBAM_BiLSTM_Attention	train	0.965	0.966	0.965	0.965	0.999	0.959	15441899.000	2845.962	0.016
Proposed_VGG16_CBAM_BiLSTM_Attention	validation	0.940	0.943	0.940	0.940	0.997	0.930	15441899.000	2845.962	0.016
Proposed_VGG16_CBAM_BiLSTM_Attention	test	0.940	0.941	0.940	0.939	0.996	0.930	15441899.000	2845.962	0.016
Proposed_VGG19_CBAM_BiLSTM_Attention	train	0.942	0.944	0.942	0.942	0.997	0.933	20751595.000	3498.283	0.020
Proposed_VGG19_CBAM_BiLSTM_Attention	validation	0.924	0.927	0.924	0.924	0.996	0.912	20751595.000	3498.283	0.020
Proposed_VGG19_CBAM_BiLSTM_Attention	test	0.904	0.907	0.904	0.904	0.993	0.888	20751595.000	3498.283	0.020

Model	**Split**	**Accuracy**	**Precision**	**Recall**	**F1_Score**	**AUC_ROC**	**Cohen_Kappa**	**Parameters**	**Training_Time_Seconds**	**Inference_Time_Per_Image_Seconds**
Proposed_VGG16_SE_Transformer	test	0.958	0.959	0.958	0.958	0.998	0.951	15443079.000	2841.727	0.016
Proposed_VGG16_CBAM_Transformer	test	0.950	0.951	0.950	0.950	0.998	0.942	15838954.000	2794.541	0.016

Proposed_VGG19_CBAM_Transformer	test	0.940	0.942	0.940	0.940	0.997	0.930	21148650.000	3604.521	0.019
Proposed_VGG16_CBAM_BiLSTM_Attention	test	0.940	0.941	0.940	0.939	0.996	0.930	15441899.000	2845.962	0.016
Proposed_VGG19_SE_Transformer	test	0.939	0.941	0.939	0.938	0.997	0.928	20752775.000	3449.826	0.019
VGG16_Baseline	test	0.906	0.913	0.906	0.906	0.995	0.890	14980935.000	2719.906	0.016
Proposed_VGG19_CBAM_BiLSTM_Attention	test	0.904	0.907	0.904	0.904	0.993	0.888	20751595.000	3498.283	0.020
VGG19_Baseline	test	0.891	0.893	0.891	0.890	0.992	0.872	20290631.000	3361.219	

Table 9 presents a comprehensive performance comparison of the baseline models and the proposed hybrid deep learning architectures across the training, validation, and testing datasets. The evaluation includes multiple performance indicators such as Accuracy, Precision, Recall, F1-Score, AUC-ROC, Cohen's Kappa coefficient, number of trainable parameters, training time, and inference time per image. These metrics collectively provide insights into classification effectiveness, model robustness, computational complexity, and deployment feasibility.

The results demonstrate that all proposed architectures significantly outperform the conventional VGG16 and VGG19 baseline networks. Among the baseline approaches, VGG16_Baseline achieved a test accuracy of 90.6% with an AUC-ROC of 0.995, whereas VGG19_Baseline attained a lower test accuracy of 89.1% and an AUC-ROC of 0.992. Although both baseline models produced satisfactory results, their performance was inferior to the attention-enhanced and transformer-based architectures, indicating the limitations of conventional CNN feature extraction for complex cotton leaf disease patterns.

Among the proposed methods, the Proposed_VGG16_SE_Transformer model achieved the best overall performance with a test accuracy of 95.8%, precision of 95.9%, recall of 95.8%, F1-score of 95.8%, AUC-ROC of 0.998, and Cohen's Kappa value of 0.951. These results indicate excellent classification capability and strong agreement between predicted and actual class labels. Furthermore, the model required only 15.44 million parameters, which is substantially fewer than VGG19-based variants, demonstrating an efficient balance between accuracy and computational complexity.

The Proposed_VGG16_CBAM_Transformer model ranked second, achieving a test accuracy of 95.0% and an AUC-ROC of 0.998. The integration of the Convolutional Block Attention Module (CBAM) with the Transformer encoder enabled the model to focus on disease-relevant regions while simultaneously capturing global contextual relationships. This combination contributed to superior classification performance compared with the baseline models while maintaining relatively low inference time.

The Proposed_VGG19_CBAM_Transformer and Proposed_VGG16_CBAM_BiLSTM_Attention models produced comparable test accuracies of 94.0%, demonstrating the effectiveness of both Transformer-based and recurrent attention-based feature learning strategies. However, the VGG19-based architecture required significantly more parameters (21.15 million) and longer training time (3604.52 seconds) than its VGG16 counterpart, indicating higher computational requirements without substantial performance gains.

The Proposed_VGG19_SE_Transformer achieved a test accuracy of 93.9% and an AUC-ROC of 0.997, further validating the effectiveness of SE attention mechanisms combined with Transformer learning. Although its performance remained competitive, the additional computational cost associated with the deeper VGG19 backbone resulted in reduced efficiency compared to the VGG16-based SE Transformer model.

The Proposed_VGG19_CBAM_BiLSTM_Attention model achieved a test accuracy of 90.4%, slightly below the VGG16 baseline. Despite incorporating advanced attention and sequential learning mechanisms, the larger VGG19 feature extractor increased model complexity without delivering proportional improvements in classification performance. This observation suggests that deeper architectures do not necessarily guarantee better results for the considered cotton leaf disease dataset.

A comparison of computational efficiency reveals that VGG16-based proposed models consistently required fewer parameters and lower training times than their VGG19-based counterparts while achieving superior or comparable performance. Inference times remained relatively stable across all models, ranging between 0.016 and 0.020 seconds per image, indicating that all proposed architectures are suitable for near real-time disease diagnosis applications.

The results confirm that incorporating attention mechanisms (SE and CBAM) and Transformer-based feature learning significantly enhances classification performance compared to traditional CNN architectures. The Proposed_VGG16_SE_Transformer emerged as the most effective model, offering the highest predictive accuracy, strongest class agreement, excellent AUC-ROC performance, and efficient computational requirements, making it the most suitable candidate for automated cotton leaf disease and pest detection systems.

Figure 10. Representative Classification Results of the Proposed Deep Learning Framework for Cotton Leaf Disease Identification.

This figure 10 presents representative test samples correctly classified by the proposed model. The first image corresponds to the Curl Virus class, the second image represents a Healthy Leaf sample, and the third image depicts Herbicide Growth Damage. For each sample, the ground-truth label, predicted class, and prediction confidence score are displayed. The results demonstrate the capability of the proposed architecture to accurately distinguish between healthy and diseased cotton leaves across multiple disease categories, highlighting its effectiveness for automated plant disease diagnosis

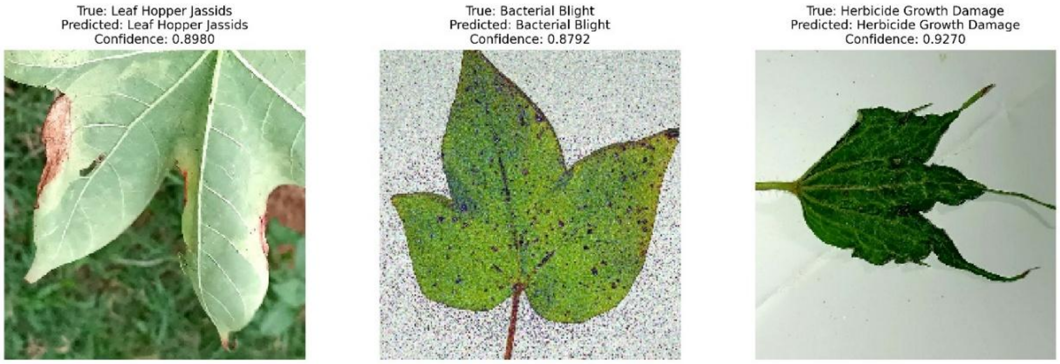


Figure 11. Representative Correct Classification Results for Herbicide Growth Damage, Bacterial Blight, and Leaf Hopper Jassids.

This figure 11 illustrates representative test samples correctly classified by the proposed deep learning framework. The displayed samples correspond to Herbicide Growth Damage, Bacterial Blight, and Leaf Hopper Jassids, respectively. For each sample, the ground-truth class, predicted class, and confidence score are presented. The results demonstrate the model's capability to accurately identify diverse cotton leaf diseases and pest infestations with high confidence, highlighting its robustness for automated disease diagnosis and crop health monitoring.

6. Conclusion

This study presented a comprehensive VGG-based deep learning framework for automated cotton leaf disease and pest classification using attention-enhanced and sequence-learning architectures. The proposed framework integrated VGG16 and VGG19 feature extractors with CBAM attention, SE attention, Transformer encoders, and BiLSTM-attention mechanisms to improve disease recognition accuracy and feature representation. Extensive experiments were conducted on a publicly available cotton leaf disease dataset containing multiple disease and healthy leaf categories. The results demonstrated that all proposed models outperformed conventional VGG16 and VGG19 baseline architectures across training, validation, and testing datasets. Among all evaluated models, the Proposed_VGG16_SE_Transformer achieved the best overall performance, attaining 95.8% test accuracy, 95.9% precision, 95.8% recall, 95.8% F1-score, and an AUC-ROC value of 0.998. The integration of channel attention and Transformer-based contextual learning enabled superior discrimination of disease symptoms while maintaining computational efficiency. Furthermore, Grad-CAM visualizations provided interpretable localization of infected regions, enhancing model transparency and reliability. The experimental findings confirm that attention-driven hybrid architectures significantly improve cotton leaf disease diagnosis and support real-time agricultural monitoring. Future work will focus on lightweight edge-deployable models, larger field-scale datasets, multimodal sensing integration, and real-time smart farming applications for precision crop health management.

References:

1. Rahman, M., M. Ullah, R. Devnath, T. Chowdhury, Gulapur Rahman, and M. Rahman. "Cotton leaf disease detection: an integration of CBAM with deep learning approaches." Int J Comput Appl 975 (2025): 8887.
2. Wang, Zhengle, Heng-Wei Zhang, Ying-Qiang Dai, Kangning Cui, Haihua Wang, Peng W. Chee, and Rui-Feng Wang. "Resource-Efficient Cotton Network: A Lightweight Deep Learning Framework for Cotton Disease and Pest Classification." Plants 14, no. 13 (2025): 2082.
3. Faisal, Hafiz Muhammad, Muhammad Aqib, Saif Ur Rehman, Khalid Mahmood, Silvia Aparicio Obregon, Rubén Calderón Iglesias, and Imran Ashraf. "Detection of cotton crops diseases using customized deep learning model." Scientific Reports 15, no. 1 (2025): 10766.
4. Pavate, Aruna, Swetta Kukreja, Surekha Janrao, Sandip Bankar, Rohini Patil, and Vijaykumar Bidve. "Efficient model for cotton plant health monitoring via YOLO-based disease prediction." Indonesian Journal of Electrical Engineering and Computer Science 37, no. 1 (2025): 164-178.
5. Azfar, Saeed, Adnan Nadeem, Kamran Ahsan, Amir Mehmood, Abdul Ali, Saad S. Al Qahtany, and Hani Almoamari. "Automated System for Detecting, Identifying, and Preventing Cotton Leaf and Boll Diseases Using Deep Learning." International Journal of Advances in Soft Computing & Its Applications 17, no. 1 (2025).
6. Rehman, Abdul, Nadeem Akhtar, and Omar H. Alhazmi. "Monitoring and predicting cotton leaf diseases using deep learning approaches and mathematical models." Scientific Reports 15, no. 1 (2025): 22570.
7. Tanwar, Priya, Rashi Shah, Jaini Shah, and Unik Lokhande. "Cotton Price Prediction and Cotton Disease Detection Using Machine Learning." In Intelligent Data Communication Technologies and Internet of Things: Proceedings of ICICI 2021, pp. 115-128. Singapore: Springer Nature Singapore, 2022.
8. Khalid, Benish, Khushbu Khalid Butt, Hamza Shahab Awan, and Irshad Ahmed Sumra. "An Image Processing System for the Detection of Cotton Crop Diseases." Journal of Computing & Biomedical Informatics 9, no. 01 (2025).
9. Damalas, Christos A., and Spyridon D. Koutroubas. "Current status of herbicide resistance in global cotton production: trends and perspectives." Crop Protection (2025): 107320.
10. Toshpulatova, Gulnoza K., Sayfulla G. Boboyev, Muhammad Zafar, Salman Majeed, Ahmed Noureldeen, Motirh Al-Mutairi, Salman Aloufi et al. "Interspecific Hybridization for Enhancing Wilt Resistance in Cotton: A Genetic Approach Against Verticillium and Fusarium Wilts." Journal of Phytopathology 173, no. 4 (2025): e70109.
11. Kumar, Rishi, Debashis Paul, S. K. Sain, Amarpreet Singh, Subhash Chandra, Satpal Singh, Rupinder Singh Cheema, and Y. G. Prasad. "Pink Bollworm, Pectinophora gossypiella (Saunders) Infestation Escalates from Incipient to Serious Proportions in North Cotton Zone of India." Journal of Crop Health 77, no. 1 (2025): 1.
12. Kakade, Kanchan J., V. A. More, Monica Shinde, Kavita Suryawanshi, and Gopal U. Shinde. "Design of Precision Agriculture System using Automating Pink Bollworm Detection in Cotton Crops: AI based Digital Approach for Sustainable Pest Management." In 2025 1st International Conference on AIML-Applications for Engineering & Technology (ICAET), pp. 1-6. IEEE, 2025.
13. Tian, Liwen, Feng Shi, Jianping Cui, and Honghai Luo. "Analysis of Cotton Production Status and Key Supporting Technologies in Xinjiang." (2025).

14. Faisal, Hafiz Muhammad, Muhammad Aqib, Khalid Mahmood, Mejdl Safran, Sultan Alfarhood, and Imran Ashraf. "A customized convolutional neural network-based approach for weeds identification in cotton crops." Frontiers in Plant Science 15 (2025): 1435301.
15. Memon, Muhammad Suleman, Pardeep Kumar, and Rizwan Iqbal. "Meta deep learn leaf disease identification model for cotton crop." Computers 11, no. 7 (2022): 102.
16. Paul Joshua, K., Sini Anna Alex, M. Mageswari, and R. Jothilakshmi. "Enhanced conditional self-attention generative adversarial network for detecting cotton plant disease in IoT-enabled crop management." Wireless Networks 31, no. 1 (2025): 299-313.
17. Patra, Aswini Kumar, and Tejashwini Gajurel. "Improved cotton leaf disease classification using parameter-efficient deep learning framework." In 2025 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON), pp. 737-742. IEEE, 2025.
18. Ozbuldu, Mustafa, and Ahmet Irvem. "Evaluation of Climate Change Impacts on Maize and Cotton Yields Under Different Climate Scenarios in Kırıkhan District, Türkiye." International Journal of Plant Production 19, no. 1 (2025): 65-81.
19. Yang, Zhi-Yu, Wan-Ke Xia, Hao-Qi Chu, Wen-Hao Su, Rui-Feng Wang, and Haihua Wang. "A comprehensive review of deep learning applications in cotton industry: From field monitoring to smart processing." Plants 14, no. 10 (2025): 1481.
20. Nagrare, V. S., Bhausaheb Naikwadi, Babasaheb B. Fand, V. Chinna Babu Naik, Prabhulinga Tenguri, Nandini Gokte-Narkhedkar, and V. N. Waghmare. "Seasonal population trend and relative occurrence of pests and their natural enemies among cotton species and cultivars in India." Journal of Cotton Research 8, no. 1 (2025): 12.
21. Xu, Ying, Benxue Ma, Guowei Yu, Ruoyu Zhang, Haibo Tan, Fujia Dong, and Huitao Bian. "Accurate cotton verticillium wilt segmentation in field background based on the two-stage lightweight DeepLabV3+ model." Computers and Electronics in Agriculture 229 (2025): 109814.
22. Wang, Yajuan, Jun Qin, Mengmeng Wei, Xiwen Liao, Wenjing Shang, Jieyin Chen, Krishna V. Subbarao, and Xiaoping Hu. "Verticillium dahliae elicitor VdSP8 enhances disease resistance through increasing lignin biosynthesis in cotton." Plant, Cell & Environment 48, no. 1 (2025): 728-745.
23. Aslam, Afira, Syed Muhammad Usman, Muhammad Zubair, Amanullah Yasin, Muhammad Owais, and Irfan Hussain. "Multi-convolutional neural networks for cotton disease detection using synergistic deep learning paradigm." PLoS One 20, no. 5 (2025): e0324293.
24. Rajendran, Sivakumar. "Enhancing Cotton Leaf Disease Detection using Deep Learning Approach: A Study and Comparison." In 2025 International Conference on Electronics and Renewable Systems (ICEARS), pp. 1450-1454. IEEE, 2025.
25. Salot, Pinal, Prachi Pancholi, Hiral Rathod, Swati Rathod, Miral Thakkar, and Jainam Shah. "Cotton leaf analysis based early plant disease detection using Machine Learning." Journal of Integrated Science and Technology 13, no. 1 (2025): 1015-1015.
26. Badar, Ayesha, Rhea Aqueel, Ali Nawaz, Umer Zeeshan Ijaz, and Kauser Abdulla Malik. "Microbiota transplantation for cotton leaf curl disease suppression—core microbiome and transcriptome dynamics." Communications Biology 8, no. 1 (2025): 380.
27. Bhujade, Vaishali, Vijay Sambhe, and Biplab Banerjee. "Cotton and soybean plant leaf dataset generation for multiclass disease classification." Journal of Phytopathology 173, no. 2 (2025): e70051.
28. Pandiyaraju, V., B. Anusha, A. M. Senthil Kumar, K. Jaspin, Shravan Venkatraman, and A. Kannan. "Spatial attention-based hybrid VGG-SVM and VGG-RF frameworks for improved cotton leaf disease detection." Neural Computing and Applications 37, no. 14 (2025): 8309-8329.
29. Zhang, Xiaojun, Shiming Liu, Peng Wu, Wanying Xu, Dingyi Yang, Yuqing Ming, Shenghua Xiao et al. "A panoramic view of cotton resistance to Verticillium dahliae: From genetic architectures to precision genomic selection." iMeta 4, no. 3 (2025): e70029.
30. Zhu, Lingxiao, Mingyu Hou, Hongchun Sun, Yongjiang Zhang, Ke Zhang, Zhiying Bai, Anchang Li, Hezhong Dong, Liantao Liu, and Cundong Li. "Zinc oxide nanoparticles conferring resistance to Verticillium wilt in cotton." Industrial Crops and Products 225 (2025): 120523.
31. Tan, Fei, Hao Cang, Xiuwen Gao, Nianyi Wu, Ruoyu Di, Yuan Zhang, Pan Gao, Xin Lv, and Chu Zhang. "Early detection of cotton Verticillium wilt based on generative adversarial networks and hyperspectral imaging technology." Industrial Crops and Products 231 (2025): 121167.
32. Mo, Henghui, and Linjing Wei. "Lightweight detection of cotton leaf diseases using StyleGAN2-ADA and decoupled focused self-attention." Journal of King Saud University Computer and Information Sciences 37, no. 4 (2025): 41.
33. Li, Weinan, Lisen Liu, Jianing Li, Weiguang Yang, Yang Guo, Longyu Huang, Zhaoen Yang, Jun Peng, Xiuliang Jin, and Yubin Lan. "Spectroscopic detection of cotton Verticillium wilt by spectral feature selection and machine learning methods." Frontiers in Plant Science 16 (2025): 1519001.