

Research Article

Epsilon-Insensitive Primal Loss With Support Vector Machine for Robust Cotton Leaf Disease Classification

Pavana Kumar Eramachetty Dhanajayalu¹, Malathi Sivasamy¹ and Chitveli Siva Balaji Yadav²

¹Department of Computer Science and Engineering, Annamalai University, Chidambaram, Tamil Nadu, India

²Department of Artificial Intelligence, Annamacharaya Institute of Technology, Tirupathi, Andhra Pradesh, India

Article history

Received: 20-09-2025

Revised: 02-06-2026

Accepted: 12-06-2026

Corresponding Author:

Pavana Kumar Eramachetty

Dhanajayalu

Department of Computer

Science and Engineering,

Annamalai University,

Chidambaram, Tamil Nadu,

India

Email:

pavankumar5490@gmail.com

Abstract: Cotton diseases significantly reduce both yield and quality of crops, making early and accurate identification of the disease crucial for effective prevention and control. However, achieving a balance between lightweight models and precise classification of cotton diseases remains challenging. This difficulty arises primarily from the dispersed nature of disease symptoms and interference from noisy backgrounds, which complicate the detection process. To address this issue, this research proposes an Epsilon-Insensitive Primal Loss Function-based Support Vector Machine (EI-PLF-SVM) for cotton leaf disease classification. The EI-PLF improves the SVM's capability to differentiate between the predicted probability and true distribution of classes, resulting in better accuracy. In the Feature extraction process, a Local Binary Pattern (LBP) and Grey-Level Co-Occurrence Matrix (GLCM) are utilized for features extraction to improve the model performance. The proposed EI-PLF-SVM incorporates an ϵ -insensitive primal loss formulation through dual texture descriptors using LBP and GLCM, facilitating robust discrimination under noisy and visually complex leaf patterns. The experimental results illustrate that the proposed EI-PLF-SVM method reaches a higher accuracy of 99.46% and 92.48% on Kaggle Cotton Disease and Plant Village datasets respectively as compared to existing works such as lesion-aware visual transformer and Custom VGG-16. These results confirm that the EI-PLF-SVM provides an effective and lightweight alternative for practical cotton disease diagnosis.

Keywords: Cotton Disease, Grey-Level Co-Occurrence Matrix, Grid Search, Local Binary Pattern, Optimized Support Vector Machine

Introduction

Crop diseases pose a major threat to agriculture, often causing extensive plant damage that results in reduced yield quality and quantity (Moupojou et al., 2023). Cotton diseases are of particular concern due to the crop's economic importance. Cotton serves as a primary source of natural fiber and plays a crucial role in clothing production. Its cultivation supports the growth of the textile industry and contributes significantly to both agricultural and industrial economies (Tiwari et al., 2024; Kang et al., 2023). Cotton production in India is pivotal to the country's agricultural and economic landscape. As one of the globe's significant cotton producers, India becomes an

important part in the global textile industry, catering to both domestic consumption and international trade (Sain et al., 2024; Kaur et al., 2024; Amin et al., 2022). The cotton sector in India not only provides a substantial source of livelihood for millions of farmers but also constitutes a critical component of the nation's industrial and export sectors (Raja and Karthikeyan, 2023). This factor highlights the significance of cotton production in India, tracing its historical importance, examining the challenges it faces, and emphasizing its vital role in the economy of Indigenous populations and rural communities (Lu et al., 2023). Cotton cultivation in India has a rich historical heritage spanning millennia. Indian cotton is used in a wide variety of textile products such as industrial textiles, apparel,

home textiles, and so on (Meshram and Alvi, 2024; Shahid et al., 2024).

The export of cotton and cotton-based products constitutes a significant source of national revenue, positioning India as a prominent player in the global textile market (Jadhav et al., 2024). Recent developments in computer vision and Deep Learning (DL) have created promising opportunities for pests and diseases detection in agriculture leaves (Bappi et al., 2025). These innovations enable automated identification processes, improving both the speed and accuracy of diagnosis. However, in India, traditional practices are still widely employed to detect potential threats to cotton crops (Ahmad, 2024; Susa et al., 2022). Despite the vast areas of land suitable for cotton cultivation, relatively little research has focused on optimizing cotton production. Historically, experts have relied on visual inspection to identify diseases and pest infestations (Rahman et al., 2025). However, this manual approach often lacks precision, leading to misdiagnoses. Owing to rising demand, several systems have been developed using modern technologies that significantly enhance the effectiveness of pest and disease management strategies (Prajapati and Patel 2025; Kalpana et al., 2023). The decline in crop disease incidence has directly contributed to increased yields, positively impacting the economy by more efficient production cycles (Abd Algani et al., 2023; Pandey et al., 2023). Consequently, the integration of information technology into agriculture has played a critical role in advancing India's cotton industry and promoting sustainable economic and environmental development (Dhage and Garg, 2023). Despite recent developments in cotton disease classification, existing DL approaches often suffer from high computational complexity, sensitivity to noisy backgrounds, and limited generalization under class imbalance conditions. This raises an important research question: Can a lightweight machine learning framework achieve classification performance comparable to DL approaches while maintaining computational efficiency and robustness? Therefore, the significant purposes of this research is to develop an efficient and robust cotton leaf disease classification framework using an Epsilon-Insensitive Primal Loss Function-based Support Vector Machine (EI-PLF-SVM) integrated with complementary texture descriptors.

Problem Statement and Objective

Cotton crop diseases significantly affect yield and quality; however, accurate and efficient identification remains a challenge because of dispersed symptoms and noisy image backgrounds. However, traditional manual methods are unreliable and inconsistent. This research aims to develop an efficient and accurate classification method using an Epsilon-Insensitive

Primal Loss Function-based Support Vector Machine (EI-PLF-SVM) for detecting cotton leaf diseases. This method enhances class discrimination while keeping computational efficiency.

This research does not claim to invent the ϵ -insensitive or primal SVM formulations; the proposed contribution is a domain-specialized adaptation:

- (i) Treats cotton leaf disease classification as ϵ -insensitive margin regression to labels $\{-1, +1\}$, yielding a tunable ambiguity band which suppresses loss around noisy decision regions
- (ii) Couples global-local texture statistics (GLCM + LBP) with the ϵ -insensitive primal SVM to stabilize boundaries under highly textured foliar patterns
- (iii) Optimizes ϵ , box constraint (C), and kernel scale via nested cross-validation, resulting in a lightweight model competitive with deep models

The crucial high points of this research are mentioned below:

- This research reformulates cotton leaf disease classification as an ϵ -insensitive margin-based learning problem, thereby adapting the Primal Loss Function of Support Vector Machine (EI-PLF-SVM) to suppress uncertainty near decision boundaries impacted through noisy backgrounds and visually ambiguous disease symptoms
- As compared to traditional SVM-based classification approaches which penalize each margin violations equally, the proposed EI-PLF-SVM produces an ϵ -tolerance band around the separating hyperplane, thereby enabling borderline samples to be ignored during optimization. This crucially enhances robustness under real-world agricultural imaging conditions without increasing the model complexity
- A complementary global-local texture depiction is attained through incorporating the Grey-Level Co-occurrence Matrix (GLCM) (spatial statistics) by Local Binary Pattern (LBP) (micro-pattern encoding), which synergizes through ϵ -insensitive loss to stabilize classification boundaries over heterogeneous leaf textures

Literature Survey

This section surveys diverse approaches for cotton disease classification with a critical discussion of advantages and limitations of these techniques.

Kumar et al. (2024) introduced a complete method to design an ML-assisted system capable of identifying diseases in cotton plants by estimating leaf images. The study involved various phases such as the dataset

acquisition, data preprocessing, model training, development of an ensemble model, estimation of models, and analysis of results. Various ML-assisted approaches were trained and estimated to determine how well they can classify cotton leaves as “Healthy” or “Diseased.” The approaches such as Random Forest (RF), SVM, Multi-Class SVM, and an Ensemble approach, were considered for classification tasks.

Rai and Pahuja (2023) presented DL techniques for classifying and predicting diseases in cotton plants using images that were captured in uncontrolled environments. Their improved Deep Convolutional Neural Network (DCNN) model was tested under three experimental settings, focusing on data split ratios, pooling layer selection, and epoch sizes.

Islam et al. (2023) introduced a DL-assisted approach for cotton leaf disease identification by fine-tuning Transfer Learning (TL) models. Their approach involved adjusting the layers and parameters of frameworks namely Inception-V3, Xception, VGG-16, and VGG-19. The models were tested using a publicly available cotton leaf dataset, and the performance of each model was comparatively validated to determine the most appropriate model for disease prediction.

Taher and Hossain (2023) developed a custom Convolutional Neural Network (CNN) for identification and classification of five types of cotton leaf disease. The architecture comprised five convolutional layers followed by four dense layers designed with a straightforward linear structure. The model was trained and validated using both augmented and original datasets to assess its reliability across various input scenarios.

Pavate et al. (2025) constructed a cotton disease prediction system using EfficientNet and YOLO neural network frameworks. The goal was to build a tool to aid farmers in recognizing cotton plant diseases and recommending appropriate pesticide usage. When real-world inputs originated from diverse sources, the system output was extensively observed and evaluated to ensure practical applicability.

Li et al. (2024) implemented a highly enhanced detection framework called CFNet-VoV GCSP-LSKNet-YOLOv8s, built on the YOLOv8s model. Key innovations include:

- (1) Replacing all backbone C2F modules with CFNet to improve multi-scale feature integration
- (2) Employing VoV-GCSP in the detection head to balance accuracy with computational efficiency
- (3) Embedding the LSKNet attention mechanism into layers that handle small objects
- (4) Integrating the XIoU loss function to improve convergence speed and accuracy

Rai and Pahuja (2024) implemented a robust method for the automated classification and detection of diseased cotton plants and leaves using the DL approach. To address the limitations of the single-model classification, a bagging ensemble approach was performed, combining five well-established transfer-learning models: InceptionV3, InceptionResNetV2, VGG16, MobileNet, and Xception. This ensemble method significantly enhanced the detection accuracy compared with standalone models.

Kukadiya et al. (2024) developed an ensemble deep convolutional neural network (DCNN) architecture combining two pre-trained models, VGG16 and InceptionV3, for early detection of cotton leaf diseases. Their ensemble model demonstrated improved predictive accuracy in both training and testing phases relative to individual base models. Additionally, extensive experiments were conducted to fine-tune hyperparameters such as learning rate, choice of optimizer, and number of epochs. The optimized settings presented in their work could serve as a reference for the future implementation of similar models for cotton disease diagnosis.

Research Gap

Recent DL-assisted cotton disease classification approaches attain greater accuracy by learning large number of parameters from large-scale, balanced datasets under controlled training conditions. Nevertheless, in practical agricultural scenarios, these approaches minimized the effectiveness when exposed to constrained training samples, visually similar disease patterns, and background noise developed through lighting variations, shadows, and occlusions. Additionally, their high training cost, memory utilization, and inference time constraint deployment on resource-constrained agricultural platforms.

Thus, the proposed EI-PLF-SVM explicitly models classification uncertainty by embedding an ϵ -insensitive tolerance band over a decision boundary, while ignoring unstable updates impacted through borderline or noisy samples. When integrated through complementary LBP-GLCM texture descriptors, the classification approach maintains discriminative power without depending on large-scale data or deep architectures. This framework easily solves the limitations identified in DL methods by attaining robust generalization, and reliable effectiveness under class imbalance and domain variation, as estimated by cross-dataset evaluation and ablation analysis.

Methods

This study introduces an EI-PLF-SVM approach for the classification of cotton leaf diseases. The GS supports

the SVM in learning subtle differences between the predicted probability distribution and actual class distributions, thereby improving accuracy. This improves an approach's capability to distinguish among similar diseases in cotton. Fig. 1 specifies the operation of proposed approach for cotton leaf disease classification.

Dataset

This study considered two different datasets, Kaggle cotton disease (<https://www.kaggle.com/datasets/seroshkarim/cotton-leaf-disease-dataset>) and PlantVillage (<https://www.kaggle.com/datasets/emmarex/plantdi-sease>), to compute the performance of the proposed method. The ‘Kaggle Cotton Disease’ dataset is a compilation of images related to cotton plants and the diseases that affect them.

Kaggle is a platform catering to data science and machine learning enthusiasts, which serves as a central hub for sharing datasets for various purposes, including competition, research, and learning. This specific dataset appears to have been curated to accelerate the development of machine learning models designed to identify and categorize diseases in cotton plants with four classes namely healthy, bacterial blight, fusarium wilt and curl virus.

Moreover, this research performs publicly accessible PlantVillage dataset, which is typically introduced by PlantVillage platform.

The comprehensive dataset comprises of 54,306 leaf images over 14 plant species and 38 various categories, while covering both healthy and diseased samples. Particularly, the dataset is imbalanced. Table 1 illustrates the information about sample distribution for each tomato leaf class. Each image 256×256 pixels in size and are in RGB arrangement, involving different color channels. Fig. 2 shows sample images of the collected datasets.

Preprocessing

Preprocessing is important before providing input to the classifier. Therefore, in this study, three preprocessing steps were considered: Image resizing, noise removal using a Gabor filter, and data augmentation. The information about these methods are provided in the following.

Image Resizing

To reduce an image dimensionality, a resizing operation was applied using a computer vision library in Python. All cotton leaf images are uniformly resized to typical resolution of 256×256 pixels (width $\times$ height). Image resizing is shown in Eq. (1) as:

$$R = Re(I, Wo, Ho, Wr, Hr) \quad (1)$$

Where R denotes resized image. Re specifies the resizing operation used in the actual image. I means an actual image. Ho and Wo denote the height and width of an actual image, individually. Hr and Wr denote height and width of resized image individually.

Noise Removal Using Gabor Filter

A Gabor filter is a versatile technique in image processing that excels at detecting edges and textures in images. It combines sine waves and bell-shaped curves to detect distinct patterns and sizes in images. It excels in dealing with variations in lighting and placement of objects in an image.

Gabor filters are utilized in a diverse range of applications, including object recognition in images and medical image analysis. They are extremely useful because they help extract significant details from complex images.

The purpose of Gabor filters is to excel at spotting edges in images. They identified crucial lines that separate different objects or regions by analyzing the texture of an image.

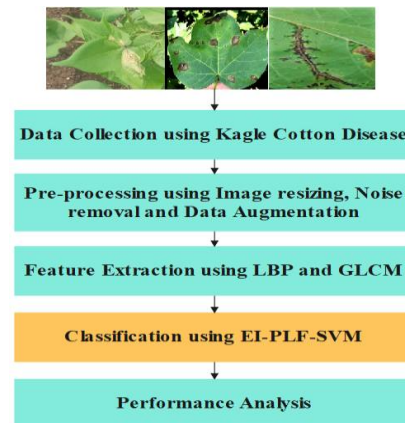


Fig. 1: Block diagram of the proposed method for cotton leaf disease classification



Fig. 2: Sample images of collected datasets

Table 1: Hyperparameters of EI-PLF-SVM

Hyperparameters	Values
Bias	-0.5648
Kernel scale	1
Gama	Default
Alpha	Default
Box constraint	0.0100
Kernel function	Radial Basis Function
Optimizer	GS

This feature is useful for differentiating various textures within an image. Feature extraction is valuable for recognizing objects and patterns. Identifying which elements are closer and farther away also aids the analysis of depth perception in pictures. Gabor filters can improve images by removing unnecessary noise or specks, while retaining vital information. In this study, a Gabor filter was used for image pre-processing. Two-dimensional Gabor filtering closely resembles how basic cells in the human visual system respond to visual stimuli.

Data Augmentation

Data augmentation is a fundamental technique in ML, particularly in DL and computer vision domains. It enriches the training dataset by generating new samples through systematic transformation of the original data. This strategy enhances model generalization, reduces the risk of performance degradation owing to limited training samples, and improves the resilience to variations. In this study, various augmentation techniques were employed such as rescaling, rotation, zooming, height and width shifting, shearing, horizontal flipping, and mode adjustments. Specifically, the images were rotated within a 45-degree range, shifted horizontally and vertically by 30%, sheared and zoomed by a factor of 0.3, also horizontally flipped using a 'reflect' fill mode to maintain edge effects. Additionally, brightness adjustments in the range of 0.5 and 1.5 simulate varying illumination conditions, further increasing dataset diversity and robustness during training. To solve data leakage, data augmentation is applied exclusively to the training subset into every fold of the cross-validation process. The testing subsets remain unaugmented and contains original samples only. Augmentation is employed after data splitting for each fold, thereby ensuring strict separation between training and testing data.

To prevent data leakage during evaluation, data augmentation is significantly applied to the training subset into each fold of the cross-validation process. The testing subsets remained comprehensively unaugmented and involved only actual unseen samples. Particularly, for every fold, the dataset is initially partitioned as training and testing partitions, after which augmentation approaches are used only to training data. This maintained strict separation between training and testing samples and enhanced the reproducibility of the reported results.

Feature Extraction

The preprocessed images obtained from earlier processing stages were used as inputs for feature extraction. These images encapsulate crucial

characteristics such as shape, texture, intensity, and gradient information. To effectively characterize these patterns, current research focuses on extracting shape and texture features, which play a pivotal role in distinguishing between different visual structures in the nodules. The detailed methodologies used for the extraction of shape and texture features are outlined below.

Shape Features

Shape features are generally invariant to changes in the imaging modality and acquisition settings, making them reliable descriptors for classification tasks. They provide complementary information on texture features by capturing the structural outline and geometric properties of the nodule. In this study, contour-based features and Hu moments were used to represent shape characteristics. Contour features extract information from an object's boundaries and convert them into feature vectors that include parameters such as contour count, length, area, and geometric layout. Hu moments are mathematical descriptors that characterize an object's shape in a way that is invariant to image transformations such as scaling, rotation, and reflection. These moments are effective for recognizing shape-based differences among histological image samples and provide robustness in representing the morphology of complex structures (Taher and Hossainm, 2023).

Texture Features

Texture features play a crucial role in differentiating between nodules that have similar shapes but vary in their internal structures or patterns. To capture this, GLCM and LBP were used. GLCM (Pavate et al., 2025) is a statistical approach that estimates the spatial relationship among pixel intensities within an image. It extracts several descriptive features, including energy, contrast, entropy, homogeneity, and variance, based on the frequency of co-occurrence of pixel pairs at defined orientations and distances. The generalized GLCM $C_{\theta,d}$ is expressed by Eq. (2):

$$C_{\theta,d}(x_1, x_2) = \left(\text{Num} \left\{ \left((a_1, b_1), (a_2, b_2) \right) \right\} \right) / K \quad (2)$$

Where $C_{\theta,d}(x_1, x_2)$ specify a measure of the co-occurrence of image patches x_1 and x_2 at distance d and orientation θ , K is a normalized factor, A is the total number of pixel pairs.

LBP (Pavate et al., 2025) is another texture analysis technique that computes local patterns by comparing each pixel with its neighboring values. It encodes results into binary values, which are then used to describe the local textural structure, as numerically formulated in Eq. (3):

$$LBP_{P,R}(x_c) = \sum_{p=0}^{r-1} \mu(x_p - x_c) 2^p, \mu(y) = \begin{cases} 1, y \geq 0 \\ 0, y < 0 \end{cases} \quad (3)$$

Where R specifies radius and species of distance from neighbouring pixels to intermediate pixels, P denotes the number of neighbouring pixels. Through this combined feature extraction process, significant shape and texture information are derived using contour analysis, Hu moments, GLCM, and LBP. These features were then passed on to the classification stage for disease recognition.

Classification Using Epsilon Insensitive With Primal Loss Function Based Support Vector Machine

This section utilizes the input of extracted features for the classification of cotton leaf disease by proposing the EI-PLF-SVM approach. Compared to hybrid and ensemble deep learning approaches, EI-PLF-SVM achieves higher classification performance with lower memory consumption and faster inference time, particularly on both Kaggle and PlantVillage datasets. The use of primal loss and epsilon-insensitive regions helps the model to focus on meaningful deviations, minimizing the impact of mislabeled or ambiguous samples. Additionally, the integration of GLCM and LBP features captures both global texture and local patterns, thereby enabling superior feature representation. This balance of accuracy, robustness, and computational efficiency indicates that the proposed method is a superior choice.

A significant concept of the SVM is to transform a nonlinear input space into a high-dimensional feature space to develop hyperplane with a nonlinear mapping. Function derivation depends on the regression problem; therefore, an Epsilon-insensitive ε -SVM regression is applied in this research. Considering that the new function of input space x as $G(x)$ and $F(x, w)$ is the functions maintained through weight vector w , a target of SVM is to identify an optimal w specified as $\hat{w}$ to reduce the computation of error among $G(x)$ and a particular estimated function of $F(x, w)$. The best value of $\hat{w}$ is reliant on primal loss function, as shown in Eq. (4) as:

$$L = \begin{cases} 0, & \text{if } |y_i - F(x_i, \hat{w})| < \varepsilon \\ |y_i - F(x_i, \hat{w})| < \varepsilon & \text{otherwise} \end{cases} \quad (4)$$

Whereas, x_i denotes an input vector of data point i ; y_i denotes a preserved value consistent with data point i ; $F(x_i, \hat{w})$ specifies a predicted value of data point i . ε is described as the ε -insensitive region (tube) around the predicted value, where no loss is incurred if $F(x_i, \hat{w})$ falls within this region. If the prediction lies outside the ε -tube, then the loss is equal to the amount that exceeds the boundary. To address the optimization issue based on the

PLF, slack variables ξ^* and ξ are implemented to reduce the dimensionality, as formulated in Eq. (5), as follows:

$$U(\sum_{i=1}^N \xi_i^* + \sum_{i=1}^N \xi_i) + \frac{1}{2}(w^t w) \quad (5)$$

Whereas, $U(\cdot)$ denotes a function of the prediction error and w is weight norm; it places greater emphasis on error U being large and more on the weight norm when U is small, therefore promoting greater generalization. N denotes count of training vectors. The limits are provided in Eq. (6), as:

$$\begin{aligned} y_i - (w^t v_i) - \varepsilon &= \xi_i \\ (w^t v_i) + b - y_i &\leq \varepsilon + \xi_i^* \\ \xi_i^* &\geq 0 \\ \xi_i &\geq 0 \end{aligned} \quad (6)$$

Where v_i denotes a training vector, and b is a group of bias. Then, a Lagrangian is formatted with respect to Lagrange multipliers γ_i and α_i are shown in Eq. (7), as follows:

$$Lag = \frac{1}{2}(w^t w) + U(\sum_{i=1}^N \xi_i^* + \sum_{i=1}^N \xi_i) - \sum_{i=1}^N \alpha_i^* [y_i - (w^t v_i) - b + \varepsilon + \xi_i^*] - \sum_{i=1}^N \alpha_i [(w^t v_i) + b - y_i + \varepsilon + \xi_i] - \sum_{i=1}^N (\gamma_i^* \xi_i^* + \gamma_i \xi_i) \quad (7)$$

As shown in Eq. (14), the Lagrangian of the SVM model depends not only on parameters w , b and ξ , but also on Lagrange multipliers. This dependency makes it difficult to solve for w , b and ξ directly by considering partial derivatives of Lagrangian. Nevertheless, by applying the idea of a saddle point and discriminating the Lagrangian to w , b and ξ , we can transform an actual issue into its dual form. This results in an equivalent optimization problem in the dual space, which is easier to solve and is formulated in Eqs. (8) and (9) as follows:

$$W(\alpha, \alpha^*) = -\varepsilon \sum_{i=1}^N (\alpha_i^* + \alpha_i) + \sum_{i=1}^N y_i (\alpha_i^* - \alpha_i) - \frac{1}{2} \sum_{i=1}^N (\alpha_i^* + \alpha_i) (\alpha_j^* + \alpha_j) (v^t v + 1)^p \quad (8)$$

Through the restraints:

$$\begin{aligned} 0 &\leq \alpha_i < U \\ 0 &\leq \alpha_i^* < U \\ \sum_{i=1}^N \alpha_i^* &= \sum_{i=1}^N \alpha_i \end{aligned} \quad (9)$$

As an outcome, the prediction of $y^{(p)}$ respective to the unknown input vector $x^{(p)}$ is illustrated in Eq. (10) as follows:

$$y^{(p)} = \sum_{i=1}^N (\alpha_i^* + \alpha_i) (v_i^t x^{(p)} + 1)^p + b \quad (10)$$

To address both linear and nonlinear regression problems efficiently, this section focuses on the core component of the SVM: The kernel function. The kernel function enables the computation of inner products in a high-dimensional feature space without clearly mapping data, as supported by Hilbert–Schmidt theory. This approach significantly reduces computational complexity and helps mitigate the problem of high-dimensionality. As a result, the significance of the SVM in classification tasks is greatly enhanced. The kernel function operation is described in Eq. (11) as:

$$K(x_1, x_2) = \langle \phi(x_1), \phi(x_2) \rangle \quad (11)$$

Whereas, $K(\cdot)$ demonstrates a kernel function, $\phi(x)$ specifies the transformation which maps x to a high-dimensional space, where $\langle \cdot \rangle$ specifies the product operator. A Radial Basis Function (RBF) attains important superiority over different kernel functions such as linear, polynomial, and sigmoid functions. RBF is formulated in Eq. (12) as follows:

$$K_{RBF}(x_1, x_2) = \exp(-\gamma \|x_1 - x_2\|^2) \gamma > 0 \quad (12)$$

Where γ denotes the hyperparameter of the SVM. It is important to note that data preprocessing is essential to equalize an input data, which helps avoid the geometric unpredictability impacted by varying feature scales and enhances a convergence speed of the model. A standardization of input sample x is defined in Eq. (13) as follows:

$$x^* = \frac{x-u}{\sigma} \quad (13)$$

Where x^* specifies a standardized data and u and σ denotes mean and standard deviation vectors, individually. In standard SVM, the ε -insensitive loss defines a tolerance tube around the target value within neglected prediction errors, thereby enhancing robustness to noise. In the proposed EI-PLF-SVM, the same principle is adapted to classification by embedding the ε -tube around the separating hyperplane. Specifically, training samples whose functional margin values fall within the range are considered and correctly classified without incurring any penalty, as they lie close enough to the decision boundary. Only samples outside this ε -band activate slack variables; hence, contributing to the loss.

The classification problem is formally defined on a labeled dataset $\{x_i, y_i\}_{i=1}^N$, where, $y_i \in \{-1, +1\}$ illustrates the class membership. This notation provides the foundation to express the ε -insensitive primal loss SVM as a constrained optimization problem, while ensuring both mathematical rigor and reproducibility. The ε -insensitive primal loss SVM seeks a separating

hyperplane while ignoring deviations that fall within an ε -tube around the margin. The optimization problem is shown in Eq. (14) as follows:

$$\min_{w, b, \{\xi_i, \xi_i^*\}} \frac{1}{2} \|w\|^2 + U \sum_{i=1}^N (\xi_i + \xi_i^*) \quad (14)$$

Subject to Eq. (15) as:

$$\begin{aligned} y_i(w^T v_i + b) &\geq 1 - \varepsilon - \xi_i, \\ y_i(w^T v_i + b) &\leq 1 + \varepsilon + \xi_i^*, \\ \xi_i, \xi_i^* &\geq 0, i = 1, \dots, N \end{aligned} \quad (15)$$

Where, $\varepsilon \geq 0$ defines the tolerance tube; v_i is a mapped feature vector. Introducing Lagrange multipliers α_i , α_i^* for ε -band constraints, the dual issue is shown in Eq. (16) as follows:

$$\begin{aligned} \max_{\alpha, \alpha^*} &-1/2 \sum_{i=1}^N \sum_{j=1}^N (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) K(v_i, v_j) - \\ &\varepsilon \sum_{i=1}^N (\alpha_i^* + \alpha_i) + \sum_{i=1}^N (\alpha_i - \alpha_i^*) \end{aligned} \quad (16)$$

This formulation explicitly embeds an ε -band around the separating hyperplane: Samples within the band incur no penalty, while those outside activate slack variables. It reduces the sensitivity of the classifier to borderline or mislabeled samples, leading to improved generalization across heterogeneous cotton leaf datasets while retaining the large-margin property of the primal SVM formulation. Algorithm 1 illustrates the pseudocode of proposed method for cotton leaf disease classification.

Algorithm 1: Cotton Leaf Disease Classification using EI-PLF-SVM

Input: Image dataset $D = \{I_1, I_2, \dots, I_n\}$, labels $D = \{y_1, y_2, \dots, y_n\}$

Output: Predicted class labels for test images

1. Dataset Preparation

1.1 Load Kaggle Cotton Disease dataset and PlantVillage dataset.

1.2 Split into training and testing folds (stratified k-fold cross-validation).

1.3 For each fold, keep the testing set un-augmented and apply augmentation only to the training subset

2. Preprocessing

For each image I in D :

2.1 Resize $I \rightarrow 256 \times 256$ pixels.

2.2 Apply Gabor filter to reduce noise and enhance texture.

2.3 Perform data augmentation:

- Rotation, flipping, scaling, shifting, brightness variation.

3. Feature Extraction

For each preprocessed image:

3.1 Compute LBP features:

- Define parameters (P neighbours, radius R).

- Encode each pixel's neighborhood into LBP code.

- 3.2 Compute GLCM features:
 - Extract contrast, homogeneity, energy, entropy, correlation.
- 3.3 Concatenate [LBP + GLCM] $\rightarrow$ feature vector F_i .
4. Feature Normalization
 - 4.1 Apply z-score standardization:
 $F_{i_norm} = (F_i - \mu)\sigma$
5. Classification using EI-PLF-SVM
 - 5.1 Initialize parameters: kernel type = RBF, $\varepsilon = \{0.1\}$, $C = \{0.01\}$, KernelScale = $\{1\}$.
 - 5.2 Optimize hyperparameters (ε , C , KernelScale) using Grid Search (GS)000 with inner k-fold CV.
 - 5.3 Train EI-PLF-SVM:
 - Objective: $\min_{w,b,\xi_i,\xi_i^*} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N (\xi_i + \xi_i^*)$
 - Subject to ε -insensitive constraints with slack variables.
 - 5.4 Store support vectors, dual coefficients, and bias term.
6. Prediction

For each test feature vector F_t :

 - 6.1 Compute kernel similarity $K(F_t, v_i)$ with each support vector.
 - 6.2 Decision function: $\hat{y} = \text{sign}(\sum_{i=1}^N (\alpha_i^* - \alpha_i) K(F_t, v_i) + b)$
7. Performance Evaluation
 - 7.1 Compute metrics: Accuracy, Precision, Recall, F1-score, MCC.
 - 7.2 Perform statistical tests (t-test, ANOVA) across folds.

End

Experimental Results

The proposed method for cotton disease classification was implemented in MATLAB 2020R, using a system with an Intel i5 processor, Windows 10 OS, and 16 GB of RAM. A 5-fold cross-validation was utilized for validating an approach's stability and generalization, as it provided the highest performance across datasets. In each fold, 80% of the data was utilized for training and 20% for testing, while also ensuring class balance across partitions. The reported mean values and variances are computed across five folds. The significance of proposed approach is validated using various performance metrics namely F1-score, recall, precision and accuracy. The mathematical expressions of these performance metrics are formulated in Eqs. (17) – (20) as follows:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (17)$$

$$\text{Precision} = \frac{TP}{TP+FP} + FP \quad (18)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (19)$$

$$F1 - \text{score} = \frac{2TP}{2TP+FP+FN} \quad (20)$$

Where, TP - True Positive; TN – True Negative; FP – False Positive and FN – False Negative.

Table 1 presents the hyperparameter settings for EI-PLF-SVM. The hyperparameters of EI-PLF-SVM model were optimized through GS strategy to ensure balanced performance. While recent studies perform meta-heuristic algorithms such as Mayfly and Moth-Flame optimization for hyperparameter tuning, GS was selected in this research to ensure reproducibility, deterministic convergence, and fair comparison over folds. As compared to stochastic meta-heuristics, GS guaranteed exhaustive exploration into the defined parameter space, thereby designing it appropriate for lightweight models through constrained hyperparameters. An RBF kernel was selected due to its better generalization capability in nonlinear classification. Key hyperparameters included a kernel scale of 1 and a box constraint of 0.01 with default values for alpha and gamma. The Accuracy curves versus C exhibited overfitting beyond $C \geq 1$ (higher variance across folds), while too small CCC underfit. Increased KernelScale beyond 2 oversmoothed RBF, reduced recall on visually subtle lesions. ε in $[0.05, 0.2]$ improved precision on noisy samples; ε yielded sharper but less stable margins. The hyperparameters in Table 1 were optimized through GS with 5-fold cross-validation, not chosen arbitrarily. A box constraint of $C = 0.01$ provided the optimal balance between bias and variance, as higher values caused overfitting while smaller values underfit. The ε parameter was set to 0.1, which improved robustness against noisy and boundary-adjacent samples compared to $\varepsilon = 0$ or $\varepsilon \geq 0.2$. For the RBF kernel, KernelScale = 1 consistently gave the highest accuracy and F1-score, as smaller values led to excessive sensitivity and larger values oversmoothed decision boundaries. These results confirmed that the selected parameters were systematically tuned for stability and generalization.

Performance Evaluation

An important aspect of the proposed approach was assumed to be associated with different previous methods. This research performed multiple experiments that had been conducted to verify the proposed method for cotton disease classification. Table 2 presents a performance analysis of the feature extraction results against baselines. Existing methods such as Gabor Filter, HOG, LBP, and GLCM were tested individually and then combined. Among these, the combined LBP-GLCM method outperformed others, achieving 99.46% accuracy on the Kaggle dataset and 92.48% accuracy on PlantVillage dataset. This clearly shows that integrating both local texture (LBP) and spatial relationships (GLCM) leads to

richer feature representations. The classifier's ability is enhanced to categorize among diseased and healthy cotton leaves. These findings justify the selection of dual-feature extraction for proposed model.

Table 3 compares the classification results of various ML approaches namely Decision Tree (DT), Random Forest (RF), K-Nearest Neighbor (KNN), traditional SVM, and the proposed EI-PLF-SVM. The EI-PLF-SVM model achieved best results across both datasets, with 99.46% accuracy on the Kaggle dataset and 92.48% accuracy on Plant Village. These improvements can be attributed to the use of a refined loss function and optimized parameter tuning. The results confirm that EI-PLF-SVM is more accurate, also more consistent and robust than the conventional classifiers tested.

Table 4 presents the performance estimation of the comprehensive k-fold analysis of the proposed EI-PLF-SVM approach using different datasets. The values of $K = 2, 4, 5, 7$ and 9 were chosen for k-fold cross-validation to analyze the balance between training stability and generalization capability. These values demonstrates typically utilized folds in ML research, where $K = 2$ and $K = 5$ offers low-bias, high-variance split appropriate for small datasets, whereas $K = 7$ and $K = 9$ produce a slightly greater granularity for most consistent estimation.

Between these, $K = 5$ reliably attained the improved accuracy across all datasets, demonstrating that it offered the optimal balance between the training and testing robustness.

Table 2: Performance analysis of feature extraction results against baselines

Dataset	Method	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Kaggle Cotton Leaf Disease	Gabour Filter	92.48	94.47	92.57	92.57
	HOG	94.37	95.37	93.57	95.32
	LBP	96.37	96.22	94.37	96.36
	GLCM	97.47	97.47	95.47	97.46
	LBP-GLCM	99.46	99.56	99.58	99.62
Plant Village	Gabour Filter	84.38	85.39	84.47	85.39
	HOG	87.64	86.39	86.32	87.30
	LBP	89.57	88.57	87.49	88.58
	GLCM	91.46	91.47	91.48	91.49
	LBP-GLCM	92.48	93.48	92.68	93.87

Table 3: Performance analysis of EI-PLF-SVM method with existing methods for cotton leaf disease classification using different datasets

Dataset	Method	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Kaggle Cotton Leaf Disease	DT	92.59	95.69	91.79	92.57
	RF	93.68	96.48	93.68	94.58
	KNN	95.79	97.50	95.78	95.58
	SVM	97.92	98.69	96.69	97.57
	EI-PLF-SVM	99.46	99.56	99.58	99.62
Plant Village	DT	85.33	86.49	84.30	82.48
	RF	86.39	89.56	87.48	86.39
	KNN	87.48	90.57	90.47	87.48
	SVM	90.48	91.56	91.44	91.48
	EI-PLF-SVM	92.48	93.48	92.68	93.87

Table 4: Cross-fold analysis of the proposed EI-PLF-SVM method for cotton leaf disease classification using different datasets

Dataset	Method	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Kaggle Cotton Leaf Disease	K=2	92.53	92.58	90.43	91.47
	K=4	94.38	97.38	93.42	93.58
	K=5	99.46	99.56	99.58	99.62
	K=7	96.48	96.42	96.39	94.39
	K=9	97.48	98.38	97.32	96.32
Plant Village	K=2	84.28	83.68	85.89	85.69
	K=4	85.38	86.48	87.85	87.49
	K=5	92.48	93.48	92.68	93.87
	K=7	87.47	88.47	91.57	91.48
	K=9	91.37	90.47	89.68	89.58

Thus, these values were chosen to maintain significant and generalizable evaluations without overfitting. This analysis describes the selection of $K = 5$ for steady performance over the datasets.

Table 5 lists the computational complexity and statistical analysis of the proposed method using different datasets. The computational complexity measures include training time, inference speed, memory usage, model size and FLOPS. In addition to classification accuracy, computational efficiency is an important consideration for practical agricultural deployment. Hence, a lightweight benchmarking analysis is performed through comparing the proposed EI-PLF-SVM against both traditional ML and DL methods, including VGG-16 and Transformer models. The comparison includes training time, inference speed, runtime memory usage, model size, and FLOPS to evaluate computational complexity and deployment suitability in resource-constrained environments. These results demonstrate that there is a significant variation among at least one model group with respect to performance. Although the EI-PLF-SVM model has a higher training time, it achieves fastest inference speed and lowest memory usage during execution. As compared to existing methods, the proposed EI-PLF-SVM handles a compact model size (<1 MB),

minimum FLOPs, and sub-millisecond inference speed while attaining statistically significant effectiveness enhancements ($p < 0.05$). These findings demonstrate that EI-PLF-SVM is a lightweight alternative appropriate for real-time and resource-constrained agricultural applications. These attributes make it suitable for real-time deployment. Statistical tests (P-Test, T-Test, ANOVA) further validate the model's superiority with the lowest p-values, thereby indicating statistically significant improvements. A paired t-test is applied to compare EI-PLF-SVM against each baseline, with the null hypothesis stating no significant difference in mean accuracy. The degrees of freedom ($df = 4$) correspond to the number of folds minus one, and p-values < 0.05 were considered statistically significant. This design ensures that the improvements are not isolated results but are consistently observed across different partitions of the data. Overall, the table illustrates that the proposed model is both high-performing and computationally efficient.

Table 6 specifies the cross-dataset analysis of proposed approach using various performance metrics.

Cross-dataset analysis is complex because models trained only on an individual dataset often overfit to dataset-specific properties such as background uniformity or image resolution.

Table 5: Computational efficiency and Statistical analysis of the proposed method based

Dataset	Methods	Training time (s)	Inference speed (ms)	Run time Memory (MB)	Model size (MB)	FLOPS (G)	P-test	T-Test	ANOVA
Kaggle	DT	0.256	8.42	18.64	0.48	0.046	0.05	0.04	0.06
Cotton	RF	1.089	9.37	24.58	0.42	1.024	0.04	0.03	0.05
Leaf	KNN	0.773	12.84	31.47	0.60	1.019	0.03	0.02	0.04
Disease	SVM	2.080	10.26	27.36	0.61	0.026	0.02	0.02	0.03
	VGG-16	5.699	14.82	19.49	0.91	2.483	0.07	0.06	0.08
	Transformer	7.693	16.94	20.83	0.95	3.609	0.08	0.07	0.09
	EI-PLF-SVM	4.155	11.42	17.59	0.32	0.021	0.01	0.01	0.02
Plant	DT	0.256	11.16	21.46	0.56	0.002	0.05	0.04	0.06
Village	RF	1.089	10.44	20.96	0.96	0.011	0.04	0.03	0.05
	KNN	0.773	14.26	17.49	0.49	2.009	0.03	0.02	0.04
	SVM	2.080	11.83	19.46	0.46	1.025	0.02	0.02	0.03
	VGG-16	6.593	15.63	19.56	0.78	3.685	0.07	0.06	0.08
	Transformer	7.598	18.41	20.78	0.56	4.696	0.08	0.07	0.09
	EI-PLF-SVM	4.155	10.01	18.48	0.35	0.038	0.01	0.01	0.02

Table 6: Cross-dataset analysis of the proposed EI-PLF-SVM method for cotton leaf disease classification using different datasets

Type	Method	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Trained on Kaggle dataset and tested on PlantVillage dataset	DT	86.05	85.62	86.04	84.06
	RF	87.01	88.89	86.89	85.95
	KNN	88.90	90.71	89.01	87.85
	SVM	90.72	91.75	90.70	88.88
	EI-PLF-SVM	91.55	92.40	91.48	91.03
Trained on PlantVillage dataset and tested on Kaggle dataset	DT	72.09	72.09	72.06	71.09
	RF	81.02	79.31	78.45	78.54
	KNN	83.02	81.03	81.12	80.46
	SVM	84.04	83.10	83.12	82.17
	EI-PLF-SVM	85.98	85.00	83.93	83.93

When trained on the Kaggle dataset and tested on Plant Village, all classifiers demonstrated a decline in effectiveness because of its higher variability; however, EI-PLF-SVM achieved 91.55% accuracy, outperforming baselines by maintaining stability over domain shift. Likely, when trained on Plant Village and tested on Kaggle, accuracies were generally lower, thereby reflecting the challenge of transferring knowledge from a noisier dataset to a more curated one. Even in this complex setting, EI-PLF-SVM achieved 85.98% accuracy, demonstrating stronger adaptability than traditional ML baselines.

Table 7 shows the performance estimation of the ablation study in this research. The contribution of each component to the complete effectiveness of the model. By comparing versions of the model with and without components such as LBP-GLCM and the Epsilon-Insensitive Primal Loss Function, it is evident that performance improves as more components are integrated. The complete version of the proposed EI-PLF-SVM with both LBP and GLCM achieved highest accuracy and F1-score. This step-by-step analysis provided strong evidence that each design choice in the proposed approach was impactful and justified.

Table 8 presents a comparative analysis of the proposed EI-PLF-SVM using different hyperparameter tuning strategies. Although meta-heuristic optimizers such as Bayesian optimizer (BO), Mayfly Algorithm (MFA), Moth-Flame Optimization (MFO) and Honey Badger Algorithm (HBA), provide competitive performance and their stochastic search behaviour introduces performance variability across runs. The

recent meta-heuristic optimization approaches BO, MFA, MFO, and HBA have been widely applied for hyperparameter tuning because of their strong global search capability in nonlinear optimization spaces. However, these methods often introduce stochastic variability and higher computational overhead. On the other hand, the proposed EI-PLF-SVM possesses a compact and low-dimensional hyperparameter space, making GS more appropriate for exhaustive and reproducible optimization. HBA and MFO achieve marginally higher upper-bound accuracies (99.52% for HBA) compared to an exhaustive GS (99.46%). However, these stochastic methods introduce performance variability across multiple runs and require significantly higher computational overhead to compute population updates. Because the proposed EI-PLF-SVM possesses a highly compact, low-dimensional hyperparameter space (comprising only the box constraint C, the tolerance band (epsilon), and the Kernel Scale), deploying complex global heuristics is computationally redundant. GS was ultimately selected as the optimal tuning strategy. While it yields a fractionally lower raw accuracy than HBA, GS guarantees 100% deterministic reproducibility, stable convergence, and a balanced F1-score (99.62%), perfectly aligning with the objective of deploying a reliable, lightweight framework for agricultural environments. The GS reliably achieves higher accuracy and stability by exhaustively exploring a well-defined and low-dimensional hyperparameter space.

Table 7: Ablation study of this research

Dataset	Method	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Kaggle Cotton	SVM	91.57	92.59	94.39	91.47
Leaf Disease	PLF-SVM without LBP-GLCM	92.48	94.38	95.37	94.23
	PLF-SVM + LBP-GLCM	95.38	95.38	96.37	96.22
	EI-PLF-SVM + LBP without LBP-GLCM	98.47	97.73	97.56	97.23
	EI-PLF-SVM + LBP-GLCM	99.46	99.56	99.58	99.62
Plant Village	SVM	85.39	84.24	83.04	80.12
	PLF-SVM without LBP-GLCM	88.57	86.92	85.20	83.48
	PLF-SVM + LBP-GLCM	90.48	89.38	87.30	86.20
	EI-PLF-SVM + LBP without LBP-GLCM	91.48	92.48	90.21	91.45
	EI-PLF-SVM + LBP-GLCM	92.48	93.48	92.68	93.87

Table 8: Performance analysis of proposed EI-PLF-SVM with different hyperparameter tuning optimizers

Dataset	Method	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Kaggle Cotton	EI-PLF-SVM with BO	99.35	99.40	99.38	99.39
Leaf Disease	EI-PLF-SVM with MFA	99.42	99.48	99.45	99.46
	EI-PLF-SVM with MFO	99.48	99.50	99.52	99.51
	EI-PLF-SVM with HBA	99.52	99.60	99.59	99.59
	EI-PLF-SVM with GS	99.46	99.56	99.58	99.62
Plant Village	EI-PLF-SVM with BO	92.15	92.30	99.58	92.20
	EI-PLF-SVM with MFA	92.35	92.50	92.10	92.45
	EI-PLF-SVM with MFO	92.50	93.10	92.40	92.90
	EI-PLF-SVM with HBA	92.65	93.60	92.70	93.20
	EI-PLF-SVM with GS	92.48	93.48	92.68	93.87

This deterministic tuning strategy ensures reproducible and optimal parameter selection for the proposed approach. The results indicate that, for compact and lightweight classifiers such as EI-PLF-SVM, Grid Search provides an efficient balance between performance, reliability, and computational efficiency.

Fig. 3 specifies the confusion matrix of proposed approach on Kaggle Cotton Disease and PlantVillage datasets. The diagonal dominance of matrices demonstrates that the classifier attains near-perfect discrimination between diseased and healthy samples with some misclassifications. Specifically, the Kaggle dataset confusion matrix demonstrates almost all test samples are mapped correctly. On more challenging Plant Village dataset, although background noise and

inter-class similarity are higher, the confusion matrix still reflects balanced recall and precision across classes.

Fig. 4 represents the ROC curve of proposed method. The ROC curves of EI-PLF-SVM on both Kaggle and PlantVillage datasets achieve AUC values above 0.98, thereby indicating excellent discrimination among diseased and healthy leaves. A near-vertical ascent on Kaggle reflects very low false positives, while the PlantVillage results confirm robustness under greater variability.

Comparative Analysis

Table 9 specifies a comparative estimation of the proposed with existing approaches using different datasets.

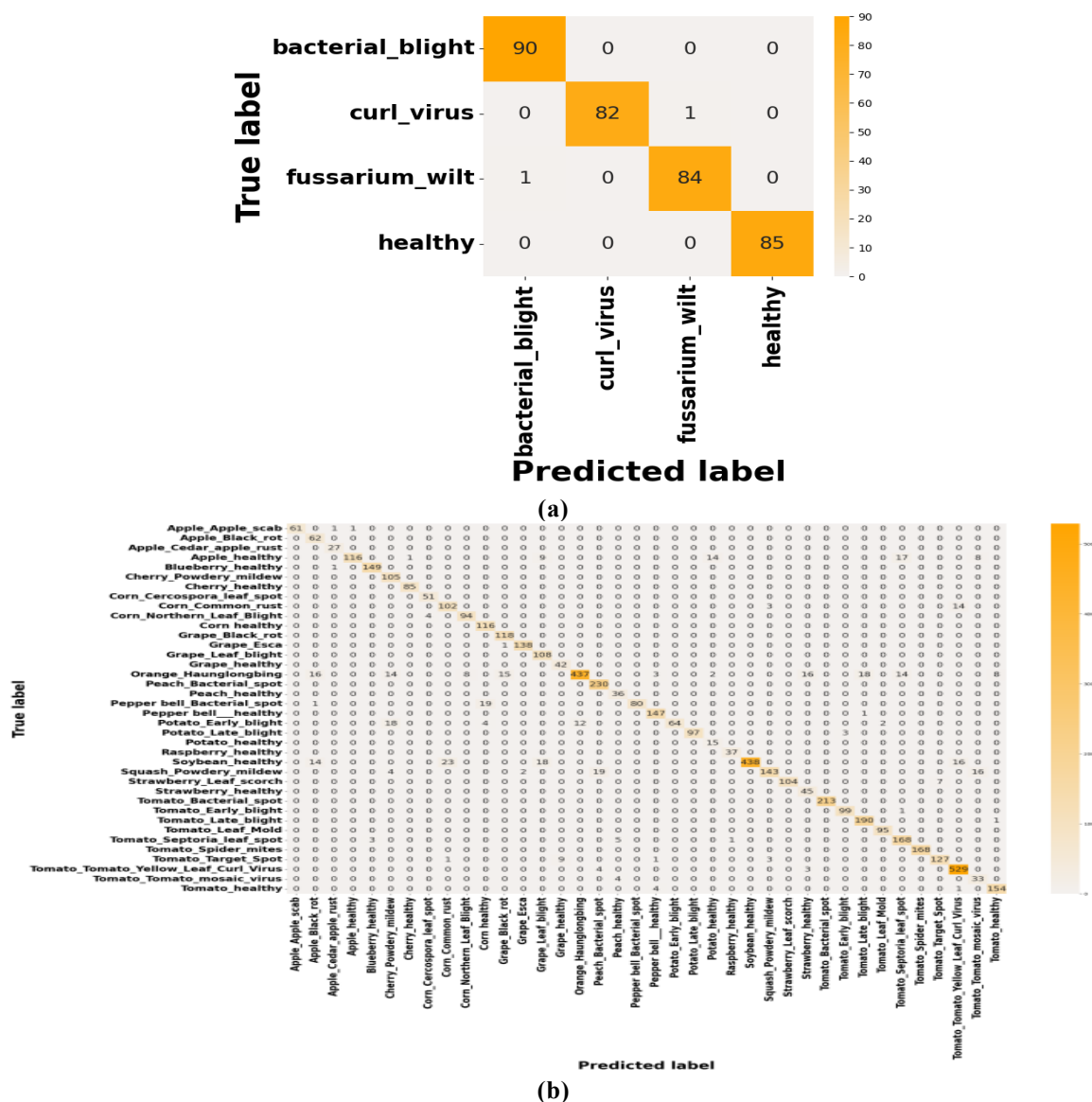


Fig. 3: Confusion matrix of proposed method, (a). Kaggle cotton leaf disease dataset, (b). Plant Village dataset

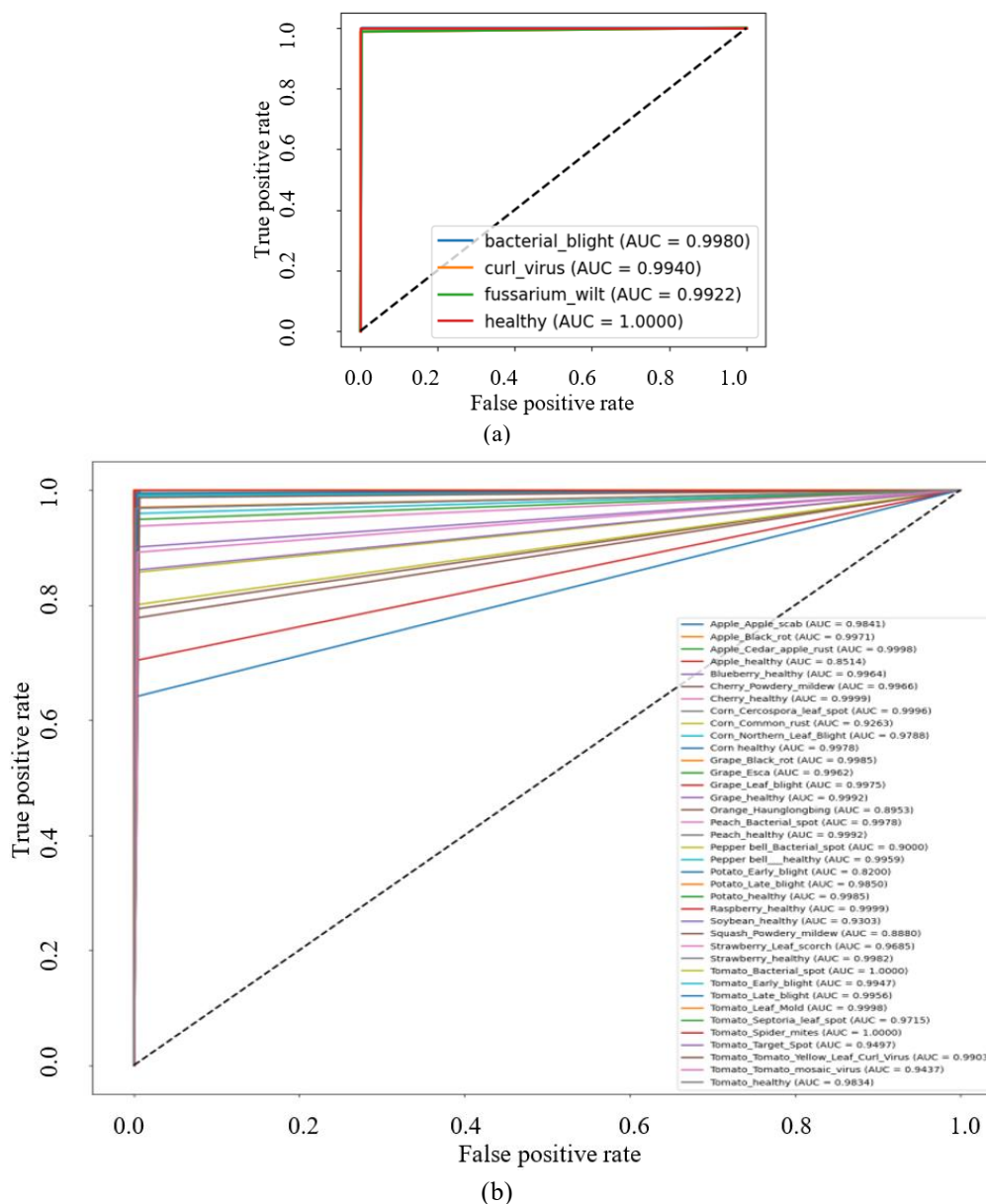


Fig. 4: ROC curve of proposed method, (a). Kaggle cotton leaf disease dataset, (b). Plant Village dataset

Table 9: Comparative analysis of the proposed method based on different datasets

Dataset	Method	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Kaggle dataset	Hybrid approach (Kumar et al., 2024)	94.5	94.4	94.4	94.4
	DCNN (Rai and Pahuja, 2023)	97.98	-	-	-
	Xception (Islam et al., 2023)	98.7	98.8	98.9	98.85
	ETL-NET-Adam (Rai and Pahuja, 2024)	98.52	99	99	99
	Proposed EI-PLF-SVM	99.46	99.56	99.58	99.62

Existing methods such as the hybrid approach (Kumar et al., 2024), DCNN (Rai and Pahuja, 2023), Xception (Islam et al., 2023), ETL-NET-Adam (Rai and Pahuja 2024), and VGG16+InceptionV3 (Kukadiya et al., 2024)

were validated and compared with proposed EI-PLF-SVM approach. In this analysis, the value of the existing VGG16+InceptionV3 (Kukadiya et al., 2024) method was converted into a percentage based on the value of the

proposed approach.

Discussion

This portion describes the drawbacks of previous studies and explains how the proposed EI-PLF-SVM approach addresses them. Key limitations in prior research include difficulties in differentiating disease-related lesions from similar visual patterns, the absence of robust feature extraction methods, suboptimal model performance, complexity in adapting to diverse conditions, and high sensitivity to environmental variations. To overcome these challenges, this research proposes an EI-PLF-SVM approach for the classification of cotton leaf diseases. The EI-PLF-SVM approach offers a significant improvement over existing methods by integrating an enhanced feature extraction process with a robust classification framework. By effectively distinguishing disease-specific lesions from visually similar patterns, it reduces misclassification errors and improves reliability. The meta-heuristic optimization approaches such as Mayfly Algorithm (MFA), Moth-Flame Optimization (MFO), Honey Badger Algorithm (HBA) and Bayesian optimizer are widely utilized for tuning the hyperparameter because of their effective exploration of large and non-convex search spaces, especially in DL approaches. While these approaches attain enhanced performance, they often introduce more computational overhead and stochastic variability. In this research, the Grid Search is performed because of their constrained and well-defined hyperparameter space of the proposed EI-PLF-SVM, ensuring deterministic, reproducible, and computationally efficient tuning appropriate for lightweight agricultural applications. The effectiveness of the proposed EI-PLF-SVM approach is tested using two standard datasets, the Kaggle cotton disease and PlantVillage datasets. For the Cotton leaf dataset, the proposed EI-PLF-SVM approach attained higher accuracy of 99.46% and 92.48% on the Kaggle Cotton Disease and Plant Village datasets respectively; thus, the lesion-aware visual transformer approach attained accuracy. However, the limitation of the proposed approach is its reliance on handcrafted features, which may restrict scalability to highly complex disease patterns. Additionally, the performance may vary under extreme illumination or unseen crop types. Future work will focus on hybrid deep shallow models and meta-heuristic-based parameter optimization.

Conclusion

DL is a recent and advanced approach in the field of image classification, and is much more efficient in the classification of diseases in cotton leaf plants. This research proposes an EI-PLF-SVM for robust cotton leaf disease classification. The key contribution is presented in adapting

the ε -insensitive primal loss to a classification model and integrating it through complementary texture descriptors (LBP and GLCM), facilitating efficient suppression of noisy boundary samples while preserving discriminative margins. This minimizes the impact of mislabeled data on the learning process, leading to more consistent performance. The utilization of data augmentation techniques enhances the amount of data from the raw input and leads to better generalizability. This research focuses on benchmark datasets that are pre-cleaned and publicly available rather than on real-time or in-the-wild face recognition environments. This decision is made because of practical limitations such as real-time data acquisition, privacy, and resource availability. Future work will concentrate on extending the proposed EI-PLF-SVM to multi-class disease identification and incorporate adaptive meta-heuristic optimization approaches for automated parameter tuning. Furthermore, deployment on embedded and edge-based agricultural monitoring systems will be explored for real-time disease diagnosis.

Notation List

Notations	Descriptions
R	Resized image
Re	Resizing operation used in the actual image
I	Actual image
Wo and Ho	Width and height of an actual image, individually
Wr and Hr	Width and height of the resized image individually
$C_{\theta,d}(x_1, x_2)$	Measure of the co-occurrence of image patches x_1 and x_2
d	Distance
θ	Orientation
K	Normalized factor
A	Total number of pixel pairs
R	Radius and species of the distance from neighbouring pixels to intermediate pixels
P	Number of neighbouring pixels
x_c	Intensity value of the centre pixel under consideration
x_p	Intensity value of p th neighbouring pixel
$\mu(\cdot)$	Thresholding function that converts the intensity difference between a neighbour pixel and the centre pixel into a binary code
x_i	Input vector of data point i
y_i	Preserved value consistent with data point i
$F(x_i, \hat{w})$	Predicted value of data point i
ε	ε -insensitive region (tube) around the predicted value
ξ^* and ξ	Slack variables
$U(\cdot)$	Function of the prediction error
w	Weight norm
w^T	Squared L2-norm of the weight vector
N	Count of training vectors
v_i	Training vector
b	Group of bias

γ_i	Lagrange multipliers
α_i^*	Lagrange multipliers enforcing the ε -insensitive constraints
$x^{(p)}$	Test input vector
$y^{(p)}$	Predicted output
y	Bias
$K(\cdot)$	Kernel function
$\phi(x)$	Transformation that maps x to a high-dimensional space
$\langle \cdot \rangle$	Product operator
γ	Hyperparameter of the SVM
x^*	Standardized data
μ and σ	Mean and standard deviation vectors individually

Acknowledgment

The authors would like to express their sincere gratitude to the Department of Computer Science and Engineering, Annamalai University, Chidambaram, Tamil Nadu, India, and the Department of Artificial Intelligence, Annamacharaya Institute of Technology, Tirupathi, Andhra Pradesh, India, for their continuous support and encouragement throughout this research work. The authors also thank all the faculty members, peers, and reviewers for their valuable suggestions that helped improve the quality of this paper.

Funding Information

The authors declare that no specific funding was received to support this research work.

Author's Contributions

Pavana Kumar Eramachetty Dhanajayalu and Malathi Sivasamy: Methodology, validation, Data curation, formal analysis, investigation, resources, Write original draft preparation, visualization, supervision, project administration, funding acquisition.

Chitveli Siva Balaji: Yadav: Software, Writing review and editing, visualization, supervision, project administration, funding acquisition.

Ethics

This article does not contain any studies with human participants or animals performed by any of the authors. All authors have read and agreed to the ethical guidelines of the journal.

References

Abd Algani, Y. M., Caro, O. J. M., Bravo, L. M. R., Kaur, C., Ansari, Mohammed Saleh Al, & Bala, B. Kiran. (2023). Leaf disease identification and classification using optimized deep learning. *Measurement: Sensors*, 25, 100643.
<https://doi.org/10.1016/j.measen.2022.100643>

Ahmad, M. (2024). Cotton Leaf Disease Detection Using Vision Transformers: A Deep Learning Approach. *African Journal of Biomedical Research*, 27(3), 5760–5769.
<https://doi.org/10.53555/ajbr.v27i3s.3421>

Amin, J., Anjum, M. A., Sharif, M., Kadry, S., & Kim, J. (2022). Explainable Neural Network for Classification of Cotton Leaf Diseases. *Agriculture*, 12(12), 2029.
<https://doi.org/10.3390/agriculture12122029>

Bappi, Md. B. R., Swapno, S. M. M. R., & Rabbi, M. M. F. (2025). Deploying DenseNet for Cotton Leaf Disease Detection on Deep Learning. *Proceedings of Trends in Electronics and Health Informatics*, 1034, 485–498.
https://doi.org/10.1007/978-981-97-3937-0_33

Dhage, S., & Garg, V. K. (2023). Cotton Plant Fungal Disease Classification using Deep Learning Models. *Proceedings of the 2023 11th International Conference on Emerging Trends in Engineering & Technology - Signal and Information Processing (ICETET - SIP)*, 1–5. <https://doi.org/10.1109/icetetsip58143.2023.10151608>

Islam, Md. M., Talukder, Md. A., Sarker, Md. R. A., Uddin, M. A., Akhter, A., Sharmin, S., Mamun, Md. S. A., & Debnath, S. K. (2023). A deep learning model for cotton disease prediction using fine-tuning with smart web application in agriculture. *Intelligent Systems with Applications*, 20, 200278.
<https://doi.org/10.1016/j.iswa.2023.200278>

Jadhav, B. D., Ghate, P. M., Chopade, P. B., Kota, P. N., & Kulkarni, S. S. (2024). An Improved Model for Plant Disease Analysis Using Attention Based Capsule Network. *International Journal of Intelligent Engineering and Systems*, 17(5), 12–21.
<https://doi.org/10.22266/ijies2024.1031.02>

Kalpna, T., Thamilselvan, R., Saravanan, T. M., Pyngkodi, M., Nandhakumar, S., Priyanka, V., & Sowndharya, M. (2023). An image-based classification and prediction of diseases on cotton leaves using deep learning techniques. *Proceedings of the 2023 International Conference on Computer Communication and Informatics (ICCCI)*, 1–6.
<https://doi.org/10.1109/iccci56745.2023.10128306>

Kang, X., Huang, C., Zhang, L., Yang, M., Zhang, Z., & Lyu, X. (2023). Assessing the severity of cotton Verticillium wilt disease from in situ canopy images and spectra using convolutional neural networks. *The Crop Journal*, 11(3), 933–940.
<https://doi.org/10.1016/j.cj.2022.12.002>

Kumar, R., Kumar, A., Bhatia, K., Nisar, K. S., Chouhan, S. S., Maratha, P., & Tiwari, A. K. (2024). Hybrid Approach of Cotton Disease Detection for Enhanced Crop Health and Yield. *IEEE Access*, 12, 132495–132507.
<https://doi.org/10.1109/access.2024.3419906>

- Kaur, A., Kukreja, V., Kumar, M., Choudhary, A., & Sharma, R. (2024). A Fine-tuned Deep Learning-based VGG16 Model for Cotton Leaf Disease Classification. *Proceedings of the 2024 IEEE 9th International Conference for Convergence in Technology (I2CT)*, 1–6.
<https://doi.org/10.1109/i2ct61223.2024.10544051>
- Kukadiya, H., Arora, N., Meva, D., & Srivastava, S. (2024). An ensemble deep learning model for automatic classification of cotton leaves diseases. *Indonesian Journal of Electrical Engineering and Computer Science*, 33(3), 1942–1949.
- Li, R., He, Y., Li, Y., Qin, W., Abbas, A., Ji, R., Li, S., Wu, Y., Sun, X., & Yang, J. (2024). Identification of cotton pest and disease based on CFNet- VoV-GCSP -LSKNet-YOLOv8s: a new era of precision agriculture. *Frontiers in Plant Science*, 15, 1348402. <https://doi.org/10.3389/fpls.2024.1348402>
- Lu, Z., Huang, S., Zhang, X., Shi, Y., Yang, W., Zhu, L., & Huang, C. (2023). Intelligent identification on cotton verticillium wilt based on spectral and image feature fusion. *Plant Methods*, 19(1), 75. <https://doi.org/10.1186/s13007-023-01056-4>
- Meshram, R. A., & Alvi, A. S. (2024). Design of an Iterative Method for Crop Disease Analysis Incorporating Graph Attention with Spatial-Temporal Learning and Deep Q-Networks. *International Journal of Intelligent Engineering and Systems*, 17(3), 706–719.
<https://doi.org/10.22266/ijies2024.0630.55>
- Moupojou, E., Tagne, A., Retraint, F., Tadonkemwa, A., Wilfried, D., Tapamo, H., & Nkenlifack, M. (2023). FieldPlant: A Dataset of Field Plant Images for Plant Disease Detection and Classification With Deep Learning. *IEEE Access*, 11, 35398–35410.
<https://doi.org/10.1109/access.2023.3263042>
- Pandey, B. N., Singh, R. P., Pandey, M. S., & Jain, S. (2023). Cotton Leaf Disease Classification Using Deep Learning based Novel Approach. *Proceedings of the 2023 International Conference on Disruptive Technologies*, 559–561.
<https://doi.org/10.1109/icdt57929.2023.10150884>
- Pavate, A., Kukreja, S., Janrao, S., Bankar, S., Patil, R., & Bidve, V. (2025). Efficient model for cotton plant health monitoring via YOLO-based disease prediction. *Indonesian Journal of Electrical Engineering and Computer Science*, 37(1), 164–178.
- Prajapati, B. S., & Patel, J. M. (2025). Vgg-Resnet Hybrid (VRH) Model for Automated Classification of Cotton Leaf Diseases. *Proceedings of the 2025 3rd International Conference on Intelligent Systems, Advanced Computing and Communication (ISACC)*, 739–798.
<https://doi.org/10.1109/isacc65211.2025.10969234>
- Rahman, M. A., Ullah, Md. S., Devnath, R. K., Chowdhury, T. H., Rahman, G., & Rahman, M. A. (2025). Cotton Leaf Disease Detection: An Integration of CBAM with Deep Learning Approaches. *International Journal of Computer Applications*, 186(67), 1–8.
<https://doi.org/10.5120/ijca2025924487>
- Rai, C. K., & Pahuja, R. (2023). Classification of Diseased Cotton Leaves and Plants Using Improved Deep Convolutional Neural Network. *Multimedia Tools and Applications*, 82(16), 25307–25325.
<https://doi.org/10.1007/s11042-023-14933-w>
- Rai, C. K., & Pahuja, R. (2024). An ensemble transfer learning-based deep convolution neural network for the detection and classification of diseased cotton leaves and plants. *Multimedia Tools and Applications*, 83(36), 83991–84024.
<https://doi.org/10.1007/s11042-024-18963-w>
- Raja, D., & Karthikeyan, M. (2023). Red Deer Optimization with Deep Learning Enabled Agricultural Plant Disease Detection and Classification Model. *International Journal of Intelligent Engineering and Systems*, 16(5), 21–30.
<https://doi.org/10.22266/ijies2023.1031.03>
- Sain, S. K., Paul, D., Kumar, P., Kumar, A., Mohan, M., Monga, D., Prakash, A. H., & Prasad, Y. G. (2024). Cotton leaf curl disease (CLCuD) prediction modeling in upland cotton under different ecological conditions using machine learning tools. *Ecological Informatics*, 81, 102648.
<https://doi.org/10.1016/j.ecoinf.2024.102648>
- Shahid, M. F., Khanzada, T. J. S., Aslam, M. A., Hussain, S., Baowidan, S. A., & Ashari, R. B. (2024). An ensemble deep learning models approach using image analysis for cotton crop classification in AI-enabled smart agriculture. *Plant Methods*, 20(1), 104. <https://doi.org/10.1186/s13007-024-01228-w>
- Susa, J. A. B., Nombrefia, W. C., Abustan, A. S., Macalisang, J., & Maaliw, R. R. (2022). Deep Learning Technique Detection for Cotton and Leaf Classification Using the YOLO Algorithm. *Proceedings of the 2022 International Conference on Smart Information Systems and Technologies (SIST)*, 1–6.
<https://doi.org/10.1109/sist54437.2022.9945757>
- Taher, A., & Hossain, Md. F. (2023). Cotton Leaves Disease Classification Using Custom Convolutional Neural Network (CNN) Model. *Proceedings of the 2023 International Conference on Next-Generation Computing, IoT and Machine Learning (NCIM)*, 1–6.
<https://doi.org/10.1109/ncim59001.2023.10212758>
- Tiwari, M., Kumar, H., Prakash, N., Kumar, S., Neware, R., Tripathi, S., & Agarwal, R. (2024). Tomato Disease Detection Using Vision Transformer with Residual L1-Norm Attention and Deep Neural Networks. *International Journal of Intelligent Engineering and Systems*, 17(1), 679–688.
<https://doi.org/10.22266/ijies2024.0229.57>