



Detection and Classification of Night Jasmine Leaf Diseases Using a Novel Deep Learning - Based Optimization Model

Grace Thabitha Jayaraj^{1*}, Ponnusamy Ramasamy²

¹ Department of Computer Science, Annamalai University, Chidambaram 608002, India

² Department of Computer and Information Science, Annamalai University, Chidambaram 608002, India

Corresponding Author Email: gracephd87@gmail.com

Copyright: ©2025 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/ts.420627>

ABSTRACT

Received: 21 July 2025

Revised: 12 November 2025

Accepted: 2 December 2025

Available online: 31 December 2025

Keywords:

night jasmine leaf disease detection and classification, Adaptive Threshold-based Segmentation, VGG-16, ISSAE, PPBO

Nyctanthes arbor-tristis, sometimes known as night jasmine, describes a beloved ornamental as well as medicinal plant valued for its fragrant blooms and significant cultural value. However, like many other plants, it is susceptible to many illnesses caused by fungus as well as bacteria, which can significantly impair its development and aesthetic appeal. Therefore, it is essential to detect diseases in night jasmine as soon as feasible in order to maintain its quality as well as prevent potential losses. In order to protect the health as well as welfare of night jasmine plants, it might be helpful to establish a precise, real-time disease identification framework. Thus, this research detects and classifies the night jasmine leaf disease using novel intelligent deep learning-oriented optimization concept. The data is initially collected from the Kaggle repository named Night Jasmine Leaf Database. The pre-processing of this gathered data is done by the bilateral filtering, median filtering, wiener filtering and Contrast Limited Adaptive Histogram Equalization (CLAHE) approaches. From this pre-processed data, the segmentation is accomplished using the adaptive threshold-based segmentation method. Now, the Visual Geometry Group-16 (VGG-16) model performs the feature extraction of this segmented data. Finally, the novel Improved Stacked Sparse Auto Encoder (ISSAE) model does the detection and classification of night jasmine leaf disease. The parameter tweaking of traditional SSAE is performed by the optimization algorithm called Paper Publishing Based Optimization (PPBO) with the consideration of returning accuracy maximization as the fitness function. According to the experimental findings, the recommended approach provides a better way to detect and classify night jasmine leaf diseases. The proposed ISSAE-PPBO model is 10.80% and 12.61% better than the other existing methods in terms of accuracy and sensitivity for the proposed night jasmine leaf disease detection and classification model, respectively.

1. INTRODUCTION

Night jasmine (*Nyctanthes arbor-tristis*), a widely grown decorative and medicinal plant, is important in the fields of horticulture, conventional medicine, and scent due to its fragrant blossoms as well as therapeutic uses [1]. However, like many other plants, night jasmine is extremely susceptible to a number of leaf diseases, including viral, bacterial, as well as fungal infections, which can seriously impair plant health, reduce flower yield, and affect entire aesthetic and commercial value [2]. Although early infections might be subtle as well as difficult to detect without close inspection, foliage diseases often exhibit obvious symptoms such as blights, spots, or color variations [3]. Hence, maintaining healthy crops as well as ensuring sustainable agricultural methods depend on the timely, accurate, and efficient detection and classification of night jasmine leaf diseases [4]. Advances in deep learning as well as AI nowadays have opened up novel possibilities in precision agriculture, particularly in the identification of plant diseases [5]. Automated methods for picture analysis that make use of deep learning methods have emerged as reliable

choices, capable of managing large datasets, identifying complex patterns, as well as correctly distinguishing among instances of healthy and diseased leaves [6].

A balance between accuracy, sensitivity, as well as specificity is also critical to ensure that both diseased and healthy leaves are accurately identified, reducing the need for needless treatments and avoiding infections that are overlooked [7]. If left unchecked, these diseases may spread quickly throughout plantations, leading to large infestations that harm night jasmine cultivation as well as increase the need for chemical treatments, increasing costs and environmental problems [8].

Conventional methods that rely on human scouting as well as expert assessments are labor-intensive, time-consuming, and prone to human error, making them unsuitable for large-scale agricultural operations [9]. With a focus on malaria, the present study aimed to do ethnobotanical research, extract phytochemical components, as well as investigate the mechanisms of pharmacological impacts and therapeutic uses [10]. The plant's leaves (NAT) were identified, and upon confirmation, the sample specimens were kept in an

herbarium. The phytoconstituents were identified by the column chromatography technique. The anti-malarial actions were reviewed. NAT leaves' in vitro efficacy was assessed in comparison to chloroquine (CQ). The ethnobotanical usage of night jasmine to treat malaria was confirmed by the present study. Iridoid glycosides that induced oxidative stress might be connected to the chemical elements that provided pharmacological actions. In many agricultural environments, particularly in small-scale and resource-limited settings, visual symptom assessment is the primary diagnostic tool, making automated image-based detection highly relevant.

Numerous innovative designs have been used in plant disease identification tasks with promising results, including SE-VRNet, customized Convolutional Neural Networks (CNNs), MULTINET, as well as LF-Mamba [11]. Methods can be confused by variations in lighting conditions, color, leaf texture, as well as overlapping symptoms within different illnesses, which can lead to missed detections or false positives [12]. Although many existing methods show good effectiveness, they may still be improved, especially in terms of improving crucial metrics that directly impact the framework's practical dependability, such as F1 score, Matthew's Correlation Coefficient (MCC), as well as False Negative Rate (FNR). Therefore, even if the field has advanced thanks to a number of well-known methods, the continuous search for more robust, effective, as well as broadly applicable solutions is necessary to fully meet the demands of real-world agricultural applications focused on night jasmine leaf disease detection.

The paper contribution is as below.

- To detect and classify the night jasmine leaf disease using novel intelligent deep learning-oriented optimization concept.
- To pre-process the gathered data by the bilateral filtering, median filtering, wiener filtering and CLAHE approaches and to accomplish the segmentation by the adaptive threshold-based segmentation method.
- To do the detection and classification of night jasmine leaf disease by the novel ISSAE, where the parameter tweaking of traditional SSAE is performed by the PPBO with the consideration of returning accuracy maximization as the fitness function.

The paper organization is as follows. Section 1 is the introduction of the night jasmine leaf disease model. Section 2 is literature survey. Section 3 is proposed methodology with proposed model, data collection, pre-processing, segmentation by adaptive threshold-based segmentation, feature extraction by VGG-16, detection and classification by novel ISSAE and PPBO algorithm. Section 4 is results and analysis. Section 5 is the conclusion.

1.1 Motivation

The urgent need to protect the vital decorative as well as therapeutic plant from yield reduction, quality degradation, and monetary losses brought on by disease outbreaks is what motivates the development of an effective methodology for identifying and categorizing night jasmine leaf diseases. Traditional manual inspection methods are time-consuming, biased, as well as often fail to detect diseases in their early stages, which cause delays and significant crop damage. While recent deep learning methods such as SE-VRNet, customized CNNs, MULTINET, as well as LF-Mamba have shown promise in addressing plant disease detection, challenges remain in achieving consistently high precision, sensitivity,

accuracy, and low FNRs, especially under changing environmental conditions. By providing farmers as well as horticulturists with timely, data-driven insights, an effective automated framework will reduce the need for overuse of pesticides, lower operating costs, and promote healthier crop management techniques. Thus, improving automated night jasmine disease detection is essential for both technical development as well as environmentally friendly farming methods.

2. RELATED WORK

Isolating the phytochemical components as well as comprehending the pharmacological action processes, especially with regard to pyretic circumstances, were the goals of this work [13]. The present study highlighted the importance of night jasmine in Ayurveda by providing comprehensive insights into its chemical constituents, biological roles of important chemicals, pharmacological impacts, therapeutic applications, as well as micro propagation. It was believed that *Nyctanthes arbor-tristis* Linn was a legendary plant with significant therapeutic value. The purpose of this study was to determine the quality criteria for the leaves in accordance with WHO recommendations in order to confirm their authenticity as well as purity. Leaf juice was employed to treat several forms of persistent fevers and was a safe laxative for babies. The right pediatric emulsion for lowering fever was presented in this article.

Using Multi-agent DRL as well as EfficientNet, a system called MULTINET was created for 3D plant leaf disease diagnosis and severity estimate [14]. Accurately identifying plant leaf diseases as well as determining their severity was the main goal of this study. The four consecutive steps listed below made up the proposed task. In order to enhance picture quality as well as balance the classes, image pre-processing was first performed for data cleaning using the Adaptable LoW Pass Weiner (AWW) filter. Next, the EMbellished Manta-Ray Optimization Algorithm (EMMARO) was used for data augmentation [15, 16]. In order to explore information from several viewpoints as well as provide improved views from diverse angles, the Block Divider Model (BDM) was used to turn 2D photos into 3D.

These methods are excellent options for detecting night jasmine leaf diseases because they make use of attention strategies, multi-layered feature extraction, as well as spatial or temporal modelling to improve classification efficacy [17]. Despite these advancements, there are still unique challenges in recognizing as well as classifying leaf diseases in night jasmine [18]. To increase the efficiency of identifying as well as evaluating plant disease instances, a novel approach that combined Neural Networks (NNs) and clustering approaches was proposed [19]. This method aimed to accurately quantify the disease's damage as well as expedite diagnosis. It provided a method for assessing damage, calculating the percentage associated with the total leaf area that was infected. This advancement highlighted the framework's evolution and was a major improvement over previous methods. This innovative approach sought to revolutionize the detection as well as monitoring of plant diseases in agriculture using the capabilities of NNs and clustering techniques, offering a more precise and practical remedy.

The present study was concerned with the preliminary phytochemical analysis as well as measurement using

established methods of bioactive substances, including terpenoids, flavonoids, phenolics, glycosides, alkaloids, tannins, cardiac glycosides, and proteins [20]. These were the elements that provided the human body with certain physiological impacts. The presence of these chemical substances, which were often known as secondary metabolites, was thought to be responsible for the plant's medicinal value. The physiological condition related to the plant, in addition to

a number of external elements such as precipitation and temperature, determined the generation of these active chemicals, which were particular to the stage or organ. The Soxhlet extraction technique was used to generate crude ethanolic extracts from the *Nyctanthes arbor-tristis* plant's leaves, flowers, stems, as well as fruits. Table 1 lists the features and challenges of some of the existing works.

Table 1. Features and challenges of some existing works

Citation	Methodology Used	Features	Challenges
Hosny et al. [15]	CNN, Local Binary Pattern (LBP)	Works across multiple datasets	Dependent on pre-processing quality
Xiao et al. [16]	Deep Residual Network combined with a Squeeze-and-Excitation (SE-VRNet) module	Enhances feature extraction	Computational load is increased.
Parekh et al. [21]	YOLOv8	Involves severity and treatment recommendations	Combining entire stages in real-time is challenging
Rahman et al. [22]	CNN	User-friendly deployment through web and mobile apps	Combining hybrid methods enhances system complexity
Wang et al. [23]	LeafMamba	Scalable for vast farms	Field testing is needed to confirm lab findings

2.1 Problem statement

The difficulty in identifying as well as categorizing illnesses of night jasmine leaves arises from the inadequacies of existing techniques in efficiently and precisely identifying various disease stages in practical situations. Traditional hand examination is labor-intensive, subjective, prone to human error, as well as often overlooks subtle or early signs of infection. Even though plant disease applications employ sophisticated deep learning methods such as SE-VRNet, custom CNNs, MULTINET, as well as LF-Mamba, these methods still face problems with noisy or imbalanced information, poor generalization across datasets, and insufficient sensitivity and precision. Furthermore, many existing methods struggle to strike a compromise among low FNRs as well as high detection precision, which is essential for preventing disease transmission and minimizing missed infections. In order to overcome these obstacles as well as provide a scalable, reliable, and automated technique for detecting and classifying night jasmine leaf diseases in actual agricultural settings, there exists a pressing need for advanced computational techniques.

While the above studies provide valuable insights into image-based plant disease detection, recent advances highlight several trends that remain underexplored for night jasmine leaf analysis. Modern transformer-based architectures, attention-guided CNNs, hyperspectral models, and biologically inspired optimization algorithms have demonstrated strong performance in broader agricultural domains, yet their applicability to night jasmine remains limited due to the absence of large, annotated datasets and the computational demands of such models. Additionally, the literature indicates that many existing frameworks struggle with noise resilience, class imbalance, and stability under varying illumination or occlusion, which are the challenges that are particularly relevant for night jasmine leaves. Furthermore, current optimization strategies used in related works often lack robustness in hyperparameter tuning, leading to inconsistent performance across datasets. Motivated by these gaps, the present study focuses on developing a lightweight yet effective ISSAE-PPBO framework that improves feature abstraction, stabilizes hyperparameter selection, and enhances

classification reliability, particularly in resource-constrained agricultural environments. This positions the proposed method as a practical alternative to more complex architectures while addressing key limitations identified in prior research.

3. PROPOSED METHODOLOGY

3.1 Proposed model

The proposed night jasmine leaf disease detection and classification model is consisted of numerous phases like data collection, pre-processing, segmentation, feature extraction, detection, and classification. The first source of the data describes the Night Jasmine Leaf Database, a Kaggle resource. The bilateral filter, median filter, Wiener filter, as well as CLAHE techniques are used to process the gathered information. The adaptive threshold-based segmentation approach is used to segment this pre-processed information. This segmented data is now subjected to feature extraction using the VGG-16 method. Finally, night jasmine leaf disease is identified and categorized using the new ISSAE method. The optimization technique called PPBO does the usual SSAE parameter modification with the objective function of maximizing return accuracy. The proposed night jasmine leaf disease detection and classification model is diagrammatically depicted in Figure 1.

3.2 Data collection

The purpose of the comprehensive collection known as the Night Jasmine Leaf Diseases Dataset is to support the study as well as management of leaf diseases that affect the night jasmine plant. The dataset is gathered from the link, “<https://www.kaggle.com/datasets/shuvokumarbasak4004/night-jasmine-leaf-diseases-dataset>.” High-resolution photos related to various leaf disease stages as well as types are included in this collection, which describes a great resource for plant pathology and agricultural technology researchers and experts. In order to facilitate the development as well as assessment of deep learning methods for illness detection and classification, the photos are arranged to depict different

disease situations. The dataset contributes to the advancement of automated plant disease diagnosis by offering a diverse array of visual data, which results in more accurate as well as efficient agricultural practices. It serves as a fundamental tool for developing early disease detection computer vision applications, which helps to lower crop loss as well as promote sustainable agriculture. The dataset is openly available on Kaggle, encouraging collaboration as well as innovation in the field of plant disease management. The dataset comprises symptom-based image labels, as pathogen-level annotations were not available for this study. The proposed framework is thus designed to operate on visual disease cues. The description of the dataset is listed in Table 2.

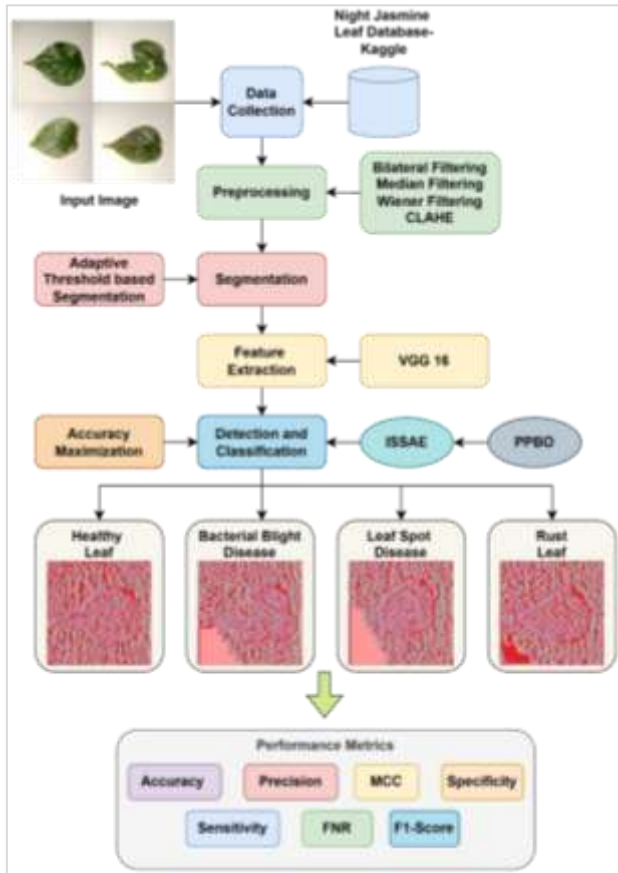


Figure 1. Proposed night jasmine leaf disease detection and classification model

Table 2. Dataset description

Class	Count of samples
Healthy	4000
Bacterial blight	2000
Leaf spot	2000
Rust	4000

3.3 Pre-processing

Pre-processing enhances the quality as well as consistency of input photos before they are used in deep learning algorithms, which is crucial for the night jasmine leaf disease detection and classification model. By ensuring that the method highlights important patterns related to leaf texture, color variations, as well as disease signs, the first processing stages improve the accuracy of feature extraction and categorization. Effective pre-processing improves the

resilience as well as dependability of the detection framework by reducing computing complexity and the chance of overfitting. Here, the pre-processing of the collected night jasmine leaf disease images is done by the bilateral filtering, median filtering, wiener filtering and CLAHE approaches.

Create a two-dimensional matrix representation of the original greyscale picture.

$$J(j, k), 1 \leq j \leq N, 1 \leq k \leq O \quad (1)$$

Here, $J(j, k)$ is the pixel intensity in an image of size $N \times O$ at the coordinates (j, k) .

Bilateral filtering: One non-linear smoothing method that maintains edges describes the bilateral filter, which is described as follows:

$$J_c(j, k) = \frac{1}{X(j, k)} \sum_{l=-s}^s \sum_{m=-s}^s H_t(l, m) \cdot H_s(J(j+l, k+m) - J(j, k)) \cdot J(j+l, k+m) \quad (2)$$

Here, the output after bilateral filtering is shown by $J_c(j, k)$, window radius is shown by s , spatial Gaussian kernel is shown by $H_t(l, m) = \exp\left(-\frac{l^2+m^2}{2\sigma_t^2}\right)$, range Gaussian kernel is shown by $H_s(\Delta) = \exp\left(-\frac{\Delta^2}{2\sigma_s^2}\right)$, $\Delta = J(j+l, k+m) - J(j, k)$, and the normalization factor is shown by $X(j, k)$ respectively.

$$X(j, k) = \sum_{l=-s}^s \sum_{m=-s}^s H_t(l, m) \cdot H_s(J(j+l, k+m) - J(j, k)) \quad (3)$$

This filter combines pixel similarity as well as spatial closeness to reduce noise and preserve edges.

Median filtering: Utilizing a sliding window, the median filter replaces every pixel with the median value of nearby pixels:

$$J_N(j, k) = \{\text{Median}\}\{J(j+l, k+m) \mid -s \leq l, m \leq s\} \quad (4)$$

The window's dimensions are $(2s+1) \times (2s+1)$, and it is very effective in removing impulse noise, sometimes known as salt-and-pepper noise, where $J_N(j, k)$ describes the filtered output at the pixel at (j, k) . The median operator avoids intensity blurring, which protects edges compared to averaging filters.

Wiener filtering: Reducing the mean square error among the estimated as well as real images is the goal of the Wiener filter. In the frequency domain, the formulation is:

$$J(v, w) = \left[\frac{|I(v, w)|^2}{|I(v, w)|^2 + \frac{T_o(v, w)}{T_g(v, w)}} \right] \cdot \frac{G(v, w)}{I(v, w)} \quad (5)$$

Here, the Fourier transform related to the degraded image is shown by $G(v, w)$, Wiener-filtered image in frequency domain is shown by $J(v, w)$, Power Spectral Densities (PSDs) associated with the noise as well as the original image is shown by $T_o(v, w)$ and $T_g(v, w)$, and the degradation function like

blur kernel is shown by $I(v, w)$ respectively. In the spatial domain (with local window estimates considered):

$$J_X(j, k) = \mu + \frac{\sigma^2 - \eta^2}{\sigma^2} \cdot (J(j, k) - \mu) \quad (6)$$

Here, the local variance is shown by σ^2 , local mean in the window is shown by μ , noisy image intensity is shown by $J(j, k)$ and noise variance is shown by η^2 respectively. When the noise power is known or estimated, this filter is perfect for eliminating Gaussian noise. The image may undergo sequential processing for efficient noise reduction.

$$J_{Final} = X(N(C(J))) \quad (7)$$

Here, bilateral filtering operation is shown by $C(J)$, median filtering operation is shown by $N(\cdot)$ and wiener filtering operation is shown by $X(\cdot)$ respectively.

CLAHE: An adaptation of the AHE method is CLAHE. CLAHE solves the over amplification issue with normal AHE by making use of the clip limit as well as count of tiles options. The CLAHE method is used to segment the image into $N \times O$ local tiles. Every tile's histogram is calculated separately. In order to compute the histogram, the mean pixel count per region may first be found using Eq. (8).

$$O_b = (O_y \times O_z) / O_h \quad (8)$$

Here, O_h is the number of grey levels, O_b is the mean number of pixels, O_y is the number of pixels in the y dimension, and O_z is the number of pixels in the z dimension. The clip limit can then be defined as shown by Eq. (9) to restrict the histogram.

$$O_{CL} = O_b \times O_{OCL} \quad (9)$$

Here, O_{OCL} stands for the normalized clip limit, which ranges from 0 to 1, and O_{CL} stands for the clip limit. The clip limit for the height associated with every tile's histogram is then established using Eq. (10).

$$I_j = \begin{cases} O_{CL} & \text{if } O_j \geq O_{CL} \\ O_j & \text{otherwise} \end{cases} \quad (10)$$

Here, $j = 1, 2, \dots, M-1, M$ indicates the count of grey levels, I_j describes the height associated with the j^{th} tile's histogram, and O_j describes the j^{th} tile's histogram. The total number of clipped pixels may be determined using Eq. (11):

$$O_d = (O_y \times O_z) - \sum_{j=0}^{M-1} I_j \quad (11)$$

Here, the count of pixels that are clipped is denoted by O_d . Once O_d has been established, the clipped pixels must be redistributed. Redistribution of pixels can be either uniform or nonuniform. Eq. (12) may be used to calculate the amount of pixels that need to be reallocated.

$$O_s = O_d / M \quad (12)$$

Here, the amount of pixels that need to be reallocated is

indicated by O_s . The reduced histogram is then normalized using Eq. (13).

$$I_j = \begin{cases} O_{CL} & \text{if } O_j + O_s \geq O_{CL} \\ O_j + O_s & \text{otherwise} \end{cases} \quad (13)$$

Here, $j = 1, 2, \dots, M-1$. To get the number of undistributed pixels, Eqs. (11) and (12) are used. Go over the Eq. (13) again until all of the pixels have been redistributed. In the end, the contextual region's cumulative histogram might be represented using Eq. (14):

$$D_j = \frac{1}{O_y \times O_z} \sum_{k=0}^1 I_k \quad (14)$$

After completing all among the previously described computations, the contextual region's histogram is aligned with Rayleigh, uniform, or exponential probability distributions to attain a desired brightness as well as visual quality. The enhanced image will be the end outcome when the CLAHE processes have been completed.

The use of multiple filters in the preprocessing stage is motivated by the heterogeneous noise characteristics observed in night jasmine leaf images. Bilateral filtering preserves edges while removing Gaussian noise, median filtering effectively mitigates impulse noise, Wiener filtering suppresses blur-induced distortions, and CLAHE enhances contrast under uneven illumination. Since no single method is capable of simultaneously addressing all these distortions, the combined pipeline provides a balanced enhancement strategy that improves the quality of downstream feature extraction.

3.4 Segmentation by adaptive threshold-based segmentation

By isolating the afflicted leaf regions from the background, segmentation is essential to the night jasmine leaf disease detection and classification model because it enables the method to concentrate only on the relevant portions of the picture. By separating the leaf region from dirt, shadows, or various adjacent objects, segmentation reduces noise as well as enhances the delineation of disease-associated characteristics, such as spots, discoloration, or texture changes. This focused isolation improves classification accuracy by enabling the algorithm to pinpoint particular characteristics linked to illness severity as well as trends. Furthermore, segmentation improves resource effectiveness by handling less irrelevant information, which lowers computing needs. By ensuring that forecasts only consider the affected leaf areas, it further improves the interpretability of the method as well as makes the disease detection framework more robust, reliable, and effective in real-world applications. Here, the segmentation process for the pre-processed image of the night jasmine leaf disease model is accomplished by the adaptive threshold-based segmentation approach.

In order to effectively distinguish among foreground as well as background, adaptive threshold-based segmentation describes a dynamic method for picture segmentation that adjusts the threshold value locally depending on the pixel neighborhood, even in the presence of irregular lighting or contrast variations. Instead of relying on a single overall threshold, this method separates a picture mathematically by computing a locally varying threshold for each pixel. Assume $J(j, k)$ be the representation of the greyscale picture input,

where (j, k) are the spatial coordinates associated with the $N \times O$ image. For every pixel $J(j, k)$, a square window with dimensions $x \times x$ is considered. The local standard deviation $\sigma_x(j, k)$ as well as local average $\mu_x(j, k)$ are calculated at this time. Next, the adaptive threshold $U(j, k)$ is computed as follows:

$$U(j, k) = \mu_x(j, k) + l \cdot \sigma_x(j, k) \quad (15)$$

Here, l describes a constant that affects how sensitive the threshold is to changes in local intensity. Segmentation is used to construct the binary image $T(j, k)$ utilizing

$$T(j, k) = \begin{cases} 1 & \text{if } J(j, k) \geq U(j, k) \\ 0 & \text{otherwise} \end{cases} \quad (16)$$

This technique adapts to local picture properties as well as works particularly well for textured sceneries, medical photos, and documents when uniform thresholding is insufficient. To increase resilience in some variations, the arithmetic mean may be swapped out for the local median or a weighted average using a Gaussian kernel. Additionally, the window size w is crucial; a bigger window produces smoother transitions, while a smaller window provides higher sensitivity to local variations. Adaptive thresholding uses contextual intensity distributions to provide a mathematically efficient method for achieving precise segmentation in complex visual environments.

3.5 Feature extraction by VGG-16

The night jasmine leaf disease detection and classification model relies heavily on feature extraction, which transforms unprocessed picture data into meaningful descriptions that highlight the unique characteristics of both healthy as well as diseased leaves. By focusing only on the major relevant data, effective feature extraction reduces dataset complexity, increases method generalization, as well as enhances classification accuracy. It ensures that the classification framework can more accurately distinguish between distinct disease kinds or healthy leaves, which will lead to a more reliable as well as efficient disease detection procedure. Here, the features are extracted from the segmented images of the proposed night jasmine leaf disease model by the VGG-16 approach.

A deep CNN (Convolutional Neural Network) represents the VGG-16. It has three fully connected layers, five max-pooling layers, thirteen convolutional layers, as well as a Softmax output layer. The Conv Layer's main tasks include making local connections, employing neurons as filters, as well as applying convolution operations on local input via applying filters and a sliding window technique. The generalized formula associated with the operating procedure is:

$$B(j, k) = (Y * G)(j, k) + c \quad (17)$$

Here, $*$ describes the convolution operator, Y describes the input matrix, c describes the bias, and $B(j, k)$ describes the position value related to filter matrix G with respect to output matrix (j, k) .

The order y associated with the final output matrix is as

follows if the input matrix has order o , the filter matrix is of order g , and border padding as well as stride are involved:

$$y = \frac{o + 2q - g}{t} + 1 \quad (18)$$

The activation function in the ReLU Layer enhances the nonlinearity of the method by applying a nonlinear mapping to the feature matrix produced from the convolutional Layer. Rectified Linear Units, or ReLUs, are widely used and are among the major significant unsaturated activation functions.

The main purpose of the pooling layer, also known as down sampling, is to reduce the number of parameters as well as their dimensionality. The idea is to divide the feature graph into many non-overlapping regions using different Windows, then apply average pooling or maximum pooling on these regions.

The data properties are obtained in the Fully Connected Layer after the convolution as well as pooling processes of the convolution and pooling layers, respectively. The correlation among the fully connected layer as well as the outputs related to linear neurons is shown as follows:

$$g(y) = \sum (x_{jk}y_j) + c_j \quad (19)$$

Here, b is the bias component in the fully connected layer's output.

3.6 Detection and classification by novel ISSAE

The core components related to the night jasmine leaf disease model are detection and classification, which enable the framework to automatically identify as well as differentiate infected leaves from healthy ones. The detection stage focusses on locating the diseased leaf sections, sometimes using segmentation findings or bounding box approaches to pinpoint the exact places displaying symptoms like blights, spots, or color changes. Following the identification of the afflicted regions, the classification stage assigns the leaf to a specific category, such as healthy, mildly infected, or severely infected, or, in the event that many disease types are found, identifies the exact kind of illness. Together, detection as well as classification forms a robust procedure that not only identifies diseased leaves yet also informs farmers about the kind and severity associated with the disease, enabling them to make more accurate and timely agricultural decisions. The detection and classification of the extracted features of the proposed night jasmine leaf disease model is done here by the novel ISSAE approach, where the parameters of SSAE are tweaked by PPBO with the intention of maximizing accuracy as the fitness function.

There are several advantages of using the SSAE to detect as well as classify illnesses in night jasmine leaves. First of all, SSAE excels in learning compressed as well as hierarchical descriptions associated with input pictures, spotting subtleties that conventional techniques would overlook, such as texture variations, color patterns, and disease signs. By ensuring that only the important as well as instructive neurons are triggered, the sparsity requirement reduces noise and improves the method's capacity for generalization. Even in situations where labelled data is few, the framework may improve classification accuracy by SSAE to obtain more abstract as well as

differentiating characteristics. Additionally, SSAE can efficiently handle high-dimensional picture information, reducing processing burden while maintaining important details. As a result, the method can withstand variations in lighting, backdrop as well as leaf placement. Using SSAE improves the detection as well as classification procedure, enabling quick and precise identification of night jasmine leaf diseases and assisting farmers in taking prompt action to protect their crops.

The SSAE has several limitations even if it offers strong feature learning for identifying as well as categorizing illnesses of night jasmine leaves. Its high computational complexity describes a major drawback, especially when working with large picture datasets or more complicated network topologies, which leads to lengthy training times as well as a substantial hardware need. Furthermore, without extensive testing, SSAE tuning becomes more difficult due to the influence of hyper parameter settings, such as the learning rate, number of hidden units, as well as sparsity penalties. When the complexity related to the method surpasses the amount of labelled information that is present, overfitting may occur, especially in agricultural datasets, in which annotated pictures are often limited. Furthermore, SSAE training typically entails unsupervised pre-training and fine-tuning, adding more steps than end-to-end methods like CNNs. Unless carefully optimized, these limitations may hinder real-world implementation and make SSAEs less suitable for real-time or resource-constrained applications.

Compared to the traditional SSAE, the ISSAE provides a number of advantages for identifying as well as categorizing illnesses of night jasmine leaves. By using attention mechanisms, optimization strategies or advanced regularization approaches, ISSAE enhances feature learning and is able to recognize more unique as well as disease-specific patterns from leaf pictures. This results in increased resistance to noise, background fluctuations as well as illumination variations. Using optimized architectures or adaptive learning approaches, ISSAE often improves convergence speed as well as reduces training time, making it more efficient on large datasets. Additionally, its improved sparsity management ensures that only the major important characteristics are emphasized, supporting the method's ability to generalize even with little labelled information. Additionally, the improved architecture helps to lessen overfitting, which is a common issue with deep methods, especially when it comes to agricultural datasets. ISSAE is highly successful for early as well as precise disease detection in night jasmine leaves due to its greater accuracy, improved stability, and enhanced computing effectiveness.

Many sparse auto-encoders are set up to form SSAE, a deep network format. The output associated with the hidden layer z_m for each hidden layer m in the group $\{1, \dots, M-1\}$ related to the SSAE method is described as below:

$$z_m = g(c_i i_{M-1} + c_m) \quad (20)$$

Here, M is the total number of layers, g describes the dimensionality related to latent factor, c describes the bias, X describes the weight and the hidden description is shown by i respectively. The first $M/2$ layers associated with the method are the encoder, and the latter $M/2$ layers are the decoder. It is suggested that the $M/2$ layer in the SSAE method should

generate the latent component, and that there should only be one hidden layer close to it. To replicate the input as well as lower the squared loss among the produced outputs, the SSAE uses a deep framework. The below formula is used to determine the loss function for SSAE.

$$K_{Spare}(X_m, c_m) = K(X_m, c_m) + \beta \sum_{j=1}^n LM(\rho || \hat{\rho}_j) \quad (21)$$

Here, the weight matrix is shown by X_m , bias vector associated with every layer is shown by c_m , divergence is shown by LM , loss function of SAE is shown by K_{Spare} , weight employed to control sparsity penalty factor is shown by β and $\sum_{j=1}^n LM(\rho || \hat{\rho}_j)$ is shown as below.

$$\sum_{j=1}^n LM(\rho || \hat{\rho}_j) = \sum_{j=1}^n \rho \log \frac{\rho}{\hat{\rho}_j} + (1 - \rho) \log \frac{1 - \rho}{1 - \hat{\rho}_j} \quad (22)$$

Here, $\rho = \frac{1}{n} \sum_{j=1}^n (b_k(y_j))$ describes the average activation of entire training instances in hidden layer neuron k , activation value in hidden layer neurons is shown by b_k and the count of hidden units per layer is shown by n respectively. The proposed novel ISSAE for the detection and classification of night jasmine leaf disease is diagrammatically shown in Figure 2.

The ISSAE architecture differs from the traditional SSAE through three key enhancements. First, an adaptive sparsity regulation mechanism dynamically adjusts the sparsity penalty based on reconstruction behavior, preventing neuron under-utilization or collapse. Second, ISSAE employs progressive layer-wise fine-tuning, which stabilizes convergence by refining each hidden layer sequentially instead of training the entire stack at once. Third, PPBO-assisted hyperparameter stabilization ensures optimal selection of sparsity coefficients, hidden units, and learning rates, reducing sensitivity to initialization. Together, these enhancements enable ISSAE to extract more discriminative and stable representations from leaf images compared to the conventional SSAE.

3.7 PPBO algorithm

By fine-tuning the method parameters, optimization is crucial to increasing the efficacy related to the night jasmine leaf disease detection and classification model. In order to improve stability, model convergence, as well as generalization, it helps select the best hyperparameters (like batch size, learning rate, and count of hidden layers). The resilience, effectiveness, as well as capacity of the pipeline to produce precise illness forecasts under various circumstances are all improved via optimization. Here, the PPBO algorithm tunes the parameters of the SSAE model in the proposed night jasmine leaf disease detection and classification model in order to return the accuracy maximization as the objective function.

The process of publishing papers is replicated by PPBO. The main driving force behind PPBO is writers' efforts to improve their work in response to reviewers' as well as editors' comments in order to get acceptance. After outlining the PPBO format, its mathematical modelling is developed by simulating the publication procedure of papers.

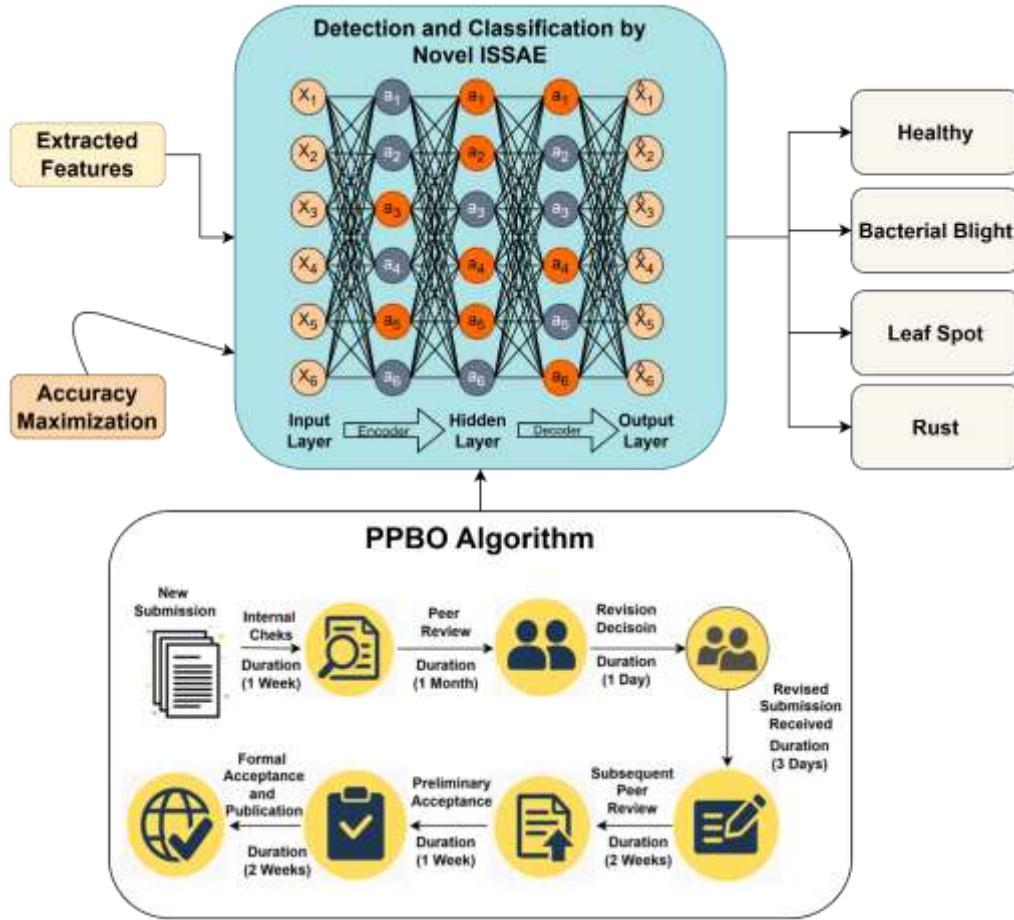


Figure 2. Proposed ISSAE for the detection and classification of night jasmine leaf disease model

By utilizing the search skills of its individuals inside the problem-handling domain, the PPBO approach describes a population-based optimization scheme that can provide effective solutions for optimization difficulties. Every population individual determines values for the decision variables in the PPBO model based on where they are in the problem-handling domain. As a result, each person in the population is a solution to the issue that a vector may mathematically depict. This means that a matrix based on Eq. (23) may be used to describe the PPBO population composed of these vectors. Utilizing Eq. (24), every PPBO individual's initial location in the problem-handling space is chosen at random.

$$Q = \begin{bmatrix} q_{1,1} & \dots & q_{1,k} & \dots & q_{1,n} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ q_{j,1} & \dots & q_{j,k} & \dots & q_{j,n} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ q_{O,1} & \dots & q_{O,k} & \dots & q_{O,n} \end{bmatrix}_{O \times n} \quad (23)$$

$$Q_j: q_{j,k} = LB_k + s_{jk} \cdot (UB_k - LB_k) \quad (24)$$

Here, Q stands for the PPBO's population matrix, O for the total count of population members, and n for the count of decision factors. The j^{th} candidate solution is denoted as Q_j , $q_{j,k}$ is its k^{th} variable, s_{jk} represents uniform random integers in the interval $[0,1]$, and LB_k and UB_k serve as the lower as well as upper bounds related to the k^{th} decision variable, respectively. In the fitness function, each individual of the

PPBO can be evaluated as a solution. As a result, a fitness function value is established for every PPBO individual, and these values may be described by a vector in accordance with the Eq. (25).

$$G = \begin{bmatrix} G_1 \\ \vdots \\ G_j \\ \vdots \\ G_O \end{bmatrix}_{O \times 1} = \begin{bmatrix} G(Y_1) \\ \vdots \\ G(Y_j) \\ \vdots \\ G(Y_O) \end{bmatrix}_{O \times 1} \quad (25)$$

In this case, G_j denotes the fitness function value obtained from the j^{th} POA individual, and G stands for the vector associated with fitness function values. The simulation of responding to reviewer input describes the basis of the first population update step in PPBO. The population's location is significantly altered as a result of this process, which also improves PPBO's exploration capabilities for worldwide research. Eq. (26) is used in PPBO model to establish the reviewer's position in the problem-handling space for each individual related to the population.

$$S_j = Q_l + Rand \cdot (Q_{Best} - Q_{Worst}) \quad (26)$$

In this case, S_j describes the reviewer associated with the j^{th} PPBO individual (i.e., the paper); Q_l ($l \in \{1,2,\dots,O\}, l \neq j$) describes the l^{th} PPBO individual; Q_{Best} describes the best PPBO individual; Q_{Worst} describes the worst PPBO

individual; and *Rand* stands for uniform random counts selected from the interval [0,1]. Utilizing Eq. (27), a novel location has been calculated for every individual related to the population in the model of PPBO on the basis of the simulation associated with the article revision procedure to answer reviewer comments. The novel location will next replace the previous location related to the corresponding individual in accordance with Eq. (28) if the fitness function value indicates enhancement.

$$Q_j^1 = Q_j + Rand \cdot (S_j - J_j \cdot Q_j) \quad (27)$$

$$Q_j = \begin{cases} Q_j^1, & G_j^1 \leq G_j \\ Q_j, & \text{otherwise} \end{cases} \quad (28)$$

In this case, the newly calculated location for the j^{th} PPBO individual based on the initial stage related to the PPBO is denoted by Q_j^1 , the fitness function value is indicated by G_j^1 , and the counts J_j are uniformly selected at random to be either 1 or 2. The count of editor comments is considered a variable in the PPBO model, and its maximum is equal to the count of decision variables according to Eq. (29). As shown by Eq. (30), the amount of editor comments determines which decision variables require adjustment.

$$O_d = \left[1 + \frac{n}{u}\right] \leq n \quad (29)$$

$$n_d = \{n_l | n_l \in \{1, 2, \dots, n\}, \quad l = 1, 2, \dots, O_d\} \quad (30)$$

The variables in this case are denoted by n , the total count of editor comments during the u^{th} iteration by O_d , the group of variables selected for the updating procedure at the u^{th} iteration by n_d , and the l^{th} variable selected by n_l . A novel location has been calculated for every individual associated with the population using Eq. (31) in order to create PPBO that is obtained by modelling the article review procedure to answer the editor's input. The new location then replaces the previous location related to the appropriate individual employing Eq. (32) if the value associated with the fitness function has increased.

$$Q_{j,n_{dl}}^2 = Q_{j,n_{dl}} + Rand \cdot (Q_{Best,n_{dl}} - J_j \cdot Q_{j,n_{dl}}), \quad l = 1, 2, \dots, O_d \quad (31)$$

$$Q_j = \begin{cases} Q_j^2, & G_j^2 \leq G_j \\ Q_j, & \text{otherwise} \end{cases} \quad (32)$$

In this case, Q_j^2 describes the newly calculated location associated with the j^{th} individual related to the population on the basis of the second stage of the PPBO, $Q_{j,n_{dl}}^2$ indicates its n_{dl}^{th} dimension, G_j^2 describes the value associated with its fitness function, *Rand* describes a random value selected from the interval [0,1], and J_j describes a uniformly selected count that can be either 1 or 2.

The PPBO algorithm operates through iterative exploration–exploitation updates. At iteration t , each candidate solution x_i^t is evaluated using the reconstruction-

based fitness function $F(x_i^t)$. During exploration, new solutions are sampled using perturbation strategies that promote diversity, while exploitation refines top-ranked candidates using weighted corrections. The update rule follows:

$$x_i^{t+1} = x_i^t + \alpha(x_{best}^t - x_i^t) + \beta \cdot \epsilon_t, \quad (33)$$

where, α denotes exploitation strength, β controls random perturbation, and ϵ_t is a stochastic disturbance. Although PPBO does not offer theoretical convergence guarantees, it functions effectively as a practical meta-heuristic optimizer for stabilizing ISSAE hyperparameters.

The initial iteration associated with the algorithm is completed after entire PPBO individuals have been modified in accordance with the first as well as second phases. The best solution discovered throughout every cycle is updated as well as saved. The algorithm then proceeds to the next iteration using the updated values, and the PPBO update process continues till the algorithm's last iteration, as described by Eqs. (26) to (32). After PPBO has finished running, the best outcome from all of the algorithm's iterations is displayed as a solution to the given problem. Algorithm 1 provides pseudocode outlining the procedures for PPBO implementation.

Algorithm 1: PPBO

Start

Input data of optimization problem [extracted features of the proposed night jasmine leaf disease detection and classification model]

Set parameters of O and U and place $j = 1$ and $u = 1$

Generate and evaluate the initial population

While $j < O$

 If $u < U$

$$S_j = Q_l + Rand \cdot (Q_{Best} - Q_{Worst})$$

$$Q_j^1 = Q_j + Rand \cdot (S_j - J_j \cdot Q_j)$$

$$Q_j = \begin{cases} Q_j^1, & G_j^1 \leq G_j \\ Q_j, & \text{otherwise} \end{cases}$$

$$O_d = \left[1 + \frac{n}{u}\right] \leq n$$

$$Q_{j,n_{dl}}^2 = Q_{j,n_{dl}} + Rand \cdot (Q_{Best,n_{dl}} - J_j \cdot Q_{j,n_{dl}}), \quad l = 1, 2, \dots, O_d$$

$$Q_j = \begin{cases} Q_j^2, & G_j^2 \leq G_j \\ Q_j, & \text{otherwise} \end{cases}$$

 else

$$u = u + 1$$

$$j = j + 1$$

End if

$$j = j + 1$$

end

Output the best quasi-optimal solution [maximized accuracy of the proposed night jasmine leaf disease detection and classification model]

Stop

The design of ISSAE is motivated by the need for controlled sparsity and stable hierarchical feature extraction. By integrating adaptive sparsity adjustment with progressive depth-wise fine-tuning, ISSAE prevents neuron inactivity and

reduces parameter sensitivity—two limitations commonly observed in traditional SSAE implementations. These modifications ensure that the autoencoder captures salient textural and morphological patterns from leaf images while maintaining efficient representation. Thus, ISSAE constitutes an enhanced variant of SSAE designed specifically for structured plant-disease imagery.

The present study focuses exclusively on the visual symptom patterns of night jasmine leaves, which represent the earliest and most accessible indicators for farmers and field workers. The objective of the proposed pipeline is early-stage screening rather than pathogen-level or infection-stage diagnosis, which requires laboratory assays and specialist input beyond the scope of an image-driven computational framework. For practical agricultural settings where laboratory facilities are limited, visual inspection remains the predominant method of disease identification. Therefore, the classifier is trained to distinguish characteristic symptom expressions such as spotting, color variation, tissue distortion, and margin decay—features that farmers traditionally rely on for early disease assessment.

4. EXPERIMENTAL ANALYSIS

4.1 Experimental setup

The proposed ISSAE-PPBO for the night jasmine leaf disease detection and classification model was implemented in MATLAB and the findings were discussed. The population size was placed to be 10. The iteration count was taken as 200. The proposed ISSAE-PPBO was compared with numerous traditional methods like MULTINET, custom CNN, SE-VRNet and LF-Mamba with consideration of analysis such as accuracy, sensitivity, precision, MCC, F1 score, specificity, and FNR to prove the effectiveness of the developed night jasmine leaf disease detection and classification model. The simulation parameters used for the experimentation are listed in Table 3.

Table 3. Simulation parameters

Parameters	Description
Platform	MATLAB
Dataset used	Night Jasmine Leaf Database
Iteration count	200
Population size	10
Objective function	Accuracy maximization
Optimization	PPBO

4.2 Segmentation and Output Image results

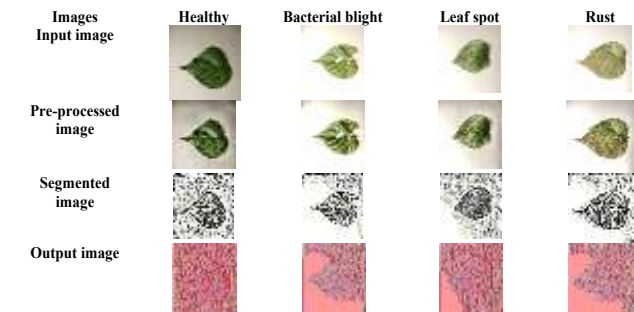


Figure 3. Sample output images of the proposed night jasmine leaf disease detection and classification model

Some of the sample input images pre-processed images, segmented images and output images of the proposed night jasmine leaf disease detection and classification model are shown in Figure 3.

4.3 Accuracy analysis

The accuracy analysis in Figure 4 for night jasmine leaf disease detection and classification demonstrates a progressive evaluation throughout ten iterations (from 20 to 200) for several methods: LF-Mamba, SE-VRNet, custom CNN, MULTINET, as well as the proposed ISSAE-PPBO. With additional rounds, a consistent improvement in accuracy across entire techniques is observed, suggesting improved learning as well as generalization with time. While the custom CNN advances from 0.8250 to 0.8695, MULTINET starts at 0.8120 and gradually rises to 0.8571 after 200 iterations. The effectiveness of SE-VRNet is better, starting at 0.8340 and reaching 0.8784. After starting at 0.8412, LF-Mamba improves even further, reaching 0.8876. Surprisingly, the proposed ISSAE-PPBO significantly outperforms the existing methods in every iteration, starting at 0.8945 and ending at a strong 0.9497. This implies that the proposed method offers improved accuracy as well as learning efficacy, making it a very promising technique for the successful identification and categorization of illnesses affecting night jasmine leaves. The proposed ISSAE-PPBO for the night jasmine leaf disease detection and classification model in terms of accuracy is 10.80%, 9.22%, 8.12% and 6.99% better than MULTINET, custom CNN, SE-VRNet and LF-Mamba, respectively.

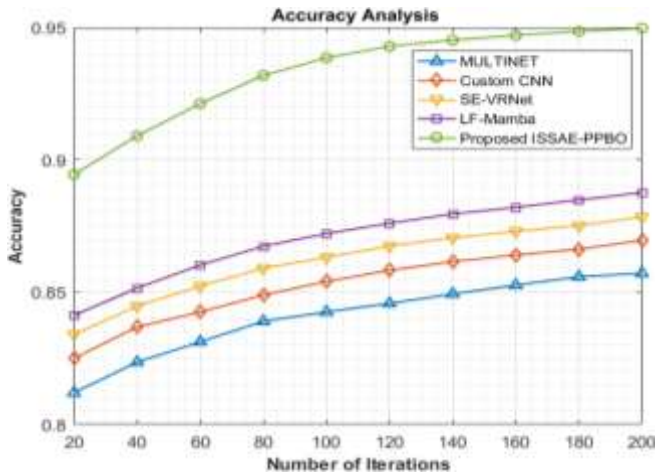


Figure 4. Accuracy analysis

4.4 Sensitivity analysis

The sensitivity analysis in Figure 5 for the night jasmine leaf disease detection and classification model demonstrates the progressive development of many methods over 10 iterations (varying from 20 to 200): MULTINET, custom CNN, SE-VRNet, LF-Mamba, as well as the proposed ISSAE-PPBO. While the custom CNN improves from 0.7582 to 0.8015, MULTINET shows a gradual rise in sensitivity from 0.7450 to 0.7880. While LF-Mamba outperforms SE-VRNet, increasing from 0.7755 to 0.8190 at the final iteration, SE-VRNet exhibits a slight enhancement, increasing from 0.7680 to 0.8115. However, the proposed ISSAE-PPBO method clearly outperforms in sensitivity at every stage, starting at 0.8320 and reaching a remarkable 0.8874 after 200 runs. This steady

benefit suggests that, in comparison to existing techniques, ISSAE-PPBO is more effective at precisely detecting sick occurrences, reducing false negatives, as well as enhancing early detection abilities. Because of its high sensitivity, it is perfect for practical applications in precision farming as well as night jasmine disease management. The sensitivity associated with the proposed ISSAE-PPBO model for night jasmine leaf disease detection and classification is 12.61%, 10.72%, 9.35% and 8.35% higher than that of MULTINET, custom CNN, SE-VRNet, and LF-Mamba, respectively.

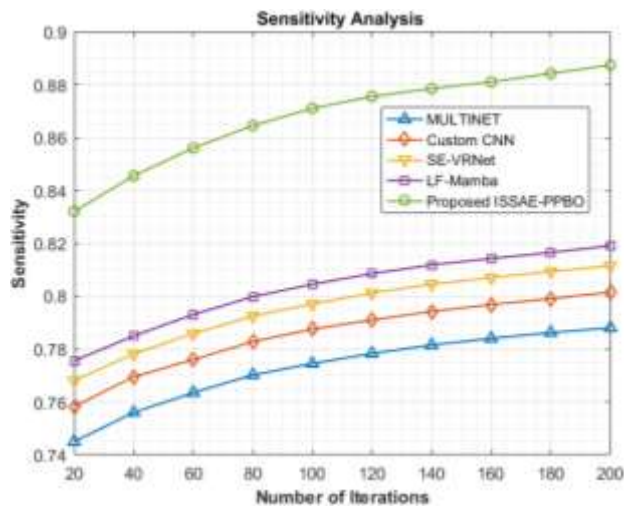


Figure 5. Sensitivity analysis

4.5 Precision analysis

The development of several methods across 10 iterations (range from 20 to 200) is shown in the precision analysis in Figure 6 for identifying as well as categorizing illnesses of night jasmine leaves. While the custom CNN increases from 0.7820 to 0.8280, showing better precision, MULTINET shows steady development, increasing from 0.7705 to 0.8147. While LF-Mamba also improves, reaching 0.8465 by the final iteration, SE-VRNet does better, going from 0.7922 to 0.8370. However, the proposed ISSAE-PPBO method continually outperforms entire existing approaches, starting at a strong 0.8602 and reaching an impressive 0.9203 by the 200th iteration. This noteworthy benefit suggests that the proposed approach is more effective at lowering false positives as well as producing extremely reliable forecasts. Its improved precision implies that ISSAE-PPBO is more successful in ensuring that only actual disease cases are marked as positive, which is important in disease detection jobs, in which a wrong classification might lead to needless treatments or missed infections. For the detection and classification of night jasmine leaf disease, the proposed ISSAE-PPBO model has a precision that is 12.96%, 11.15%, 9.95% and 8.72% higher than that of MULTINET, custom CNN, SE-VRNet, and LF-Mamba, respectively.

4.6 MCC analysis

The enhanced effectiveness associated with several methods across ten iterations (range from 20 to 200) is displayed in the MCC analysis in Figure 7 for the night jasmine leaf disease detection and classification model. From 0.6850 to 0.7320, MULTINET shows consistent improvement, while the custom CNN slightly outperforms it,

going from 0.6982 to 0.7440. While LF-Mamba continues on its upward trend, rising from 0.7158 to 0.7610 at the final iteration, SE-VRNet exhibits improved effectiveness, rising from 0.7085 to 0.7540. However, starting with a significantly higher 0.7820 and reaching an amazing 0.8540 by iteration 200, the proposed ISSAE-PPBO method clearly performs exceptionally well in MCC. This continuously high MCC demonstrates ISSAE-PPBO's robust as well as reliable classification capacity by achieving a more efficient balance between true positives, true negatives, false positives, as well as false negatives. High MCC values demonstrate the method's overall prediction accuracy and make it particularly helpful for precisely identifying diseases in night jasmine leaves. The ISSAE-PPBO model outperforms MULTINET, custom CNN, SE-VRNet, and LF-Mamba in terms of MCC for the detection and classification of night jasmine leaf disease by 16.67%, 14.78%, 13.26% and 12.22%, respectively.

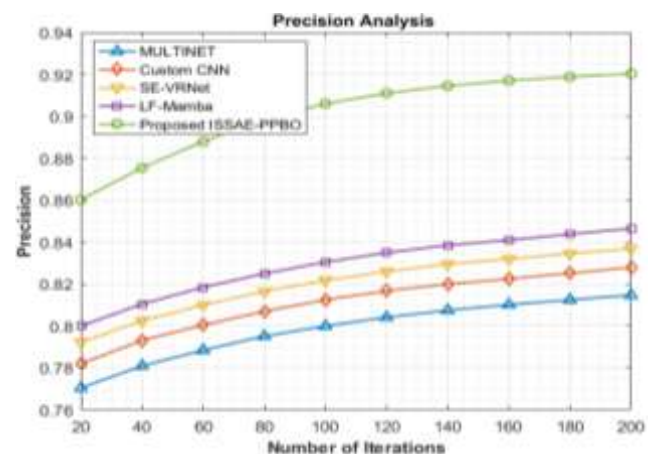


Figure 6. Precision analysis

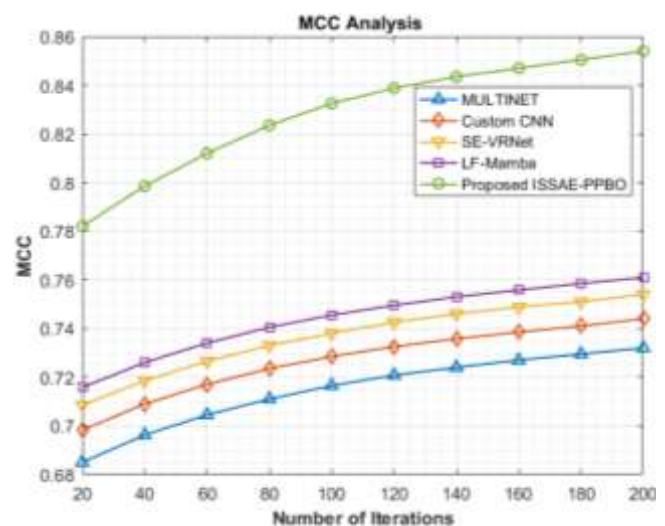


Figure 7. MCC analysis

4.7 F1 score analysis

The night jasmine leaf disease detection and classification method's F1 score assessment in Figure 8 shows how every method improved across ten iterations (from 20 to 200), demonstrating how well it balanced recall as well as precision. While custom CNN accomplishes better, moving from 0.7645 to 0.8095, MULTINET shows steady improvement, increasing from 0.7520 to 0.7958. SE-VRNet achieves better

results, increasing from 0.7740 to 0.8185, while LF-Mamba improves even more, reaching 0.8270 at the final iteration. However, the proposed ISSAE-PPBO method, which starts at 0.8455 and reaches an impressive 0.8980 by iteration 200, significantly outperforms entire existing methods. This remarkable F1 score indicates that the ISSAE-PPBO achieves a remarkable balance among correctly identifying diseases as well as minimizing false positives, making it highly reliable in actual disease detection scenarios. Its potential as an innovative method of categorizing night jasmine leaf disease is demonstrated by its continuously better F1 score effectiveness. In terms of F1 score for identifying and categorizing night jasmine leaf disease, the ISSAE-PPBO model outperforms MULTINET, custom CNN, SE-VRNet, and LF-Mamba by 12.84%, 10.93%, 9.71% and 8.59%, respectively.

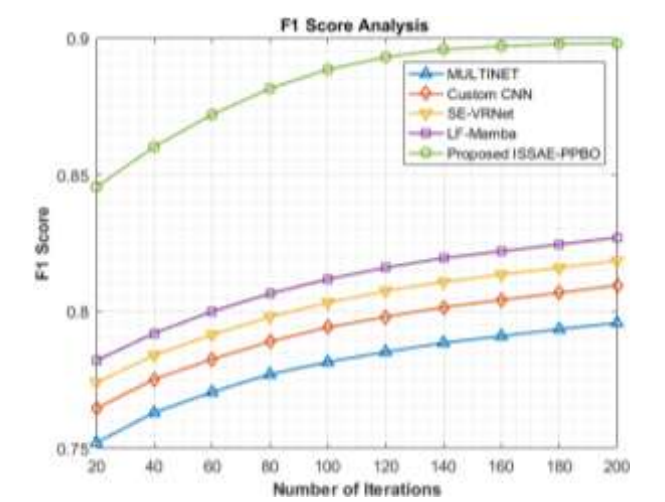


Figure 8. F1 score analysis

4.8 Specificity analysis

Every technique's ability to identify healthy (non-diseased) leaves across ten iterations (20 to 200) is highlighted in Figure 9 analyzing specificity for the night jasmine leaf disease detection and classification model. While the custom CNN improves from 0.8235 to 0.8998, showing better non-disease diagnosis, MULTINET shows steady improvement, rising from 0.8102 to 0.8935. LF-Mamba continues to enhance, reaching 0.9162 in the most recent version, while SE-VRNet shows an even larger effectiveness gain, increasing from 0.8331 to 0.9060. However, starting at an excellent 0.9107 and reaching the maximal specificity of 0.9641 after 200 iterations, the proposed ISSAE-PPBO method stands out. According to this, ISSAE-PPBO is very good at correctly ruling out healthy leaves, lowering false positives, as well as making sure that only instances that are actually contaminated are detected. For agricultural uses, this level of specificity is crucial since inflating the availability of a disease might lead to unnecessary interventions as well as wasteful resource usage. For night jasmine leaf disease detection and classification, the ISSAE-PPBO model outperforms MULTINET, custom CNN, SE-VRNet, and LF-Mamba by 7.90%, 7.15%, 6.41% and 5.23%, respectively, in terms of the specificity.

4.9 FNR analysis

The effectiveness related to every strategy in reducing

missed detections (diseased leaves mistakenly labelled as healthy) across ten iterations (varying from 20 to 200) is shown in the FNR analysis in Figure 10 for the detection and classification model of night jasmine leaf disease. While MULTINET begins with a FNR of 0.1898 and progressively lowers it to 0.1205, custom CNN improves from 0.1784 to 0.1102, demonstrating better control over false negatives. While LF-Mamba continues this trend, falling FNR from 0.1587 to 0.0919, SE-VRNet performs better, dropping from 0.1679 to 0.1024. However, starting at a significantly lesser 0.0893 as well as reaching just 0.0359 by the final iteration, the proposed ISSAE-PPBO method achieves the largest drop. The suggested method's improved capacity to precisely detect diseased leaves, lowering the frequency of missed instances, is highlighted by its consistently minimal FNR. Since missed infections can spread rapidly, a minimal FNR is crucial for identifying agricultural diseases. The efficacy related to ISSAE-PPBO emphasizes its usefulness in safeguarding crop health. The ISSAE-PPBO model outperforms MULTINET, custom CNN, SE-VRNet, and LF-Mamba in terms of FNR for identifying and categorizing night jasmine leaf disease by 70.21%, 67.42%, 64.94% and 60.94%, respectively.

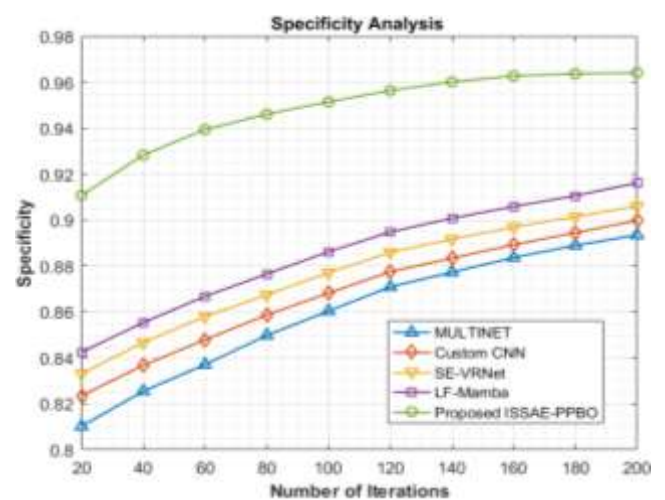


Figure 9. Specificity analysis

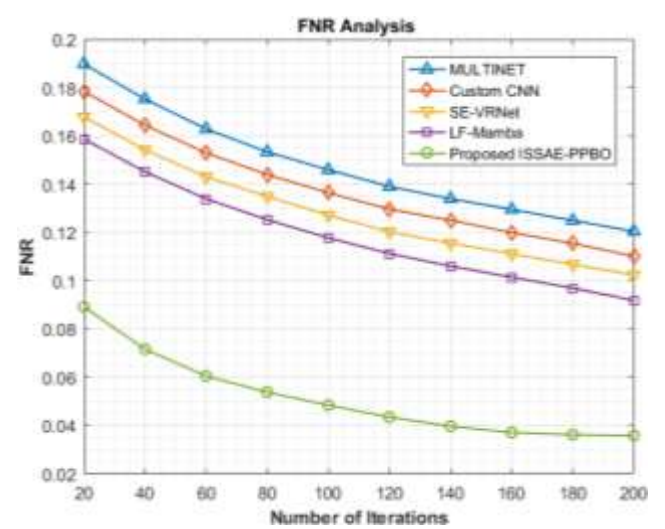


Figure 10. FNR analysis

4.10 Confusion matrix analysis

The night jasmine leaf disease detection model's

performance in 3-class and 4-class classifications is illustrated by the confusion matrices in Figure 11. Although some healthy leaves are mistakenly classified as rust, the method shows good accuracy in the three-class configuration (healthy, rust, multiple), mostly properly categorizing healthy as well as multiple classes. The method generates strong predictions for healthy as well as rust in the 4-class configuration (healthy,

rust, bacterial blight, and leaf spot); however, the misclassifications show that there exists more misunderstanding among bacterial blight as well as leaf spot. The 4-class approach does a good job of handling the added complexity, but it causes greater misunderstanding within classes, especially when it comes to comparable disease categories.

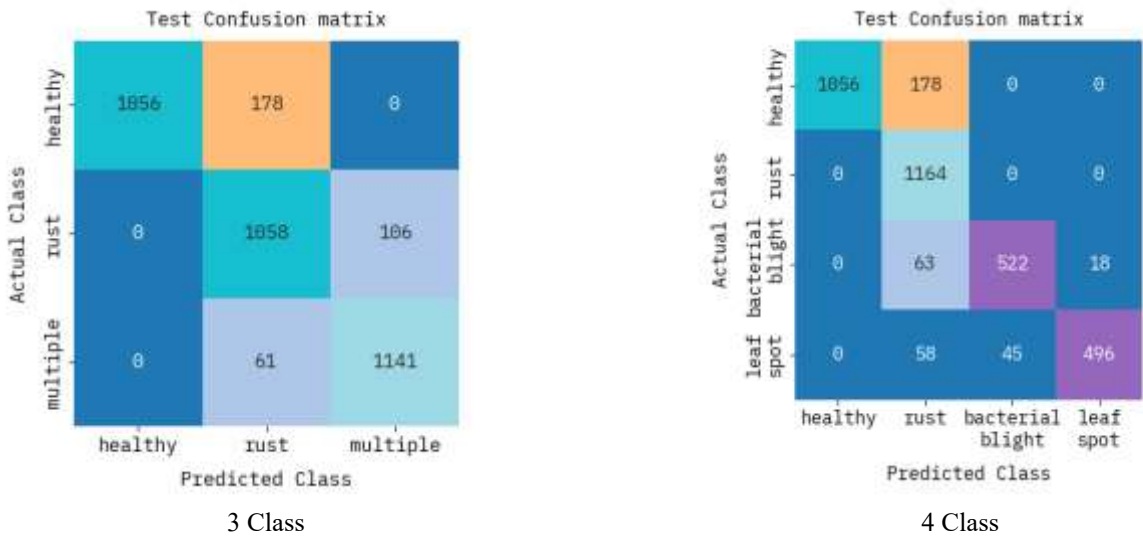


Figure 11. Confusion matrix analysis

4.11 Precision recall curve analysis

In both the 3-class and 4-class setups, the precision-recall curves for the night jasmine leaf disease detection model show exceptional effectiveness as in Figure 12. Every class in the three-class model (healthy, rust, and multiple) has high Average Precision (AP) ratings of greater than 0.97, indicating a strong balance among recall as well as precision. In the four-class method (healthy, rust, bacterial blight, and leaf spot), rust as well as healthy both have almost ideal APs (~0.99), while bacterial blight has a good score of 0.9667, but leaf spot has a lesser AP of 0.8892, suggesting that this class is more difficult to distinguish. In general, even as categorization complexity increases, the method continuously exhibits high precision-recall efficacy.

4.12 ROC curve analysis

The night jasmine leaf disease detection model's ROC curve study shows excellent discriminative abilities in both 3-class and 4-class tasks, as in Figure 13. The AUC values in the three-class setup (healthy, rust, multiple) are higher than 0.987, indicating almost perfect classification having minimal false positives. Healthy as well as rust achieve exceptionally high AUCs (~0.995–0.9975) in the complex 4-class arrangement (bacterial blight, leaf spot, rust, and healthy), whereas bacterial blight (0.9852) and leaf spot (0.9698) show fewer but strong effectiveness. Despite the rise in classification complexity, the method exhibits high True Positive Rates (TPRs) while keeping very few False Positive Rates (FPRs).

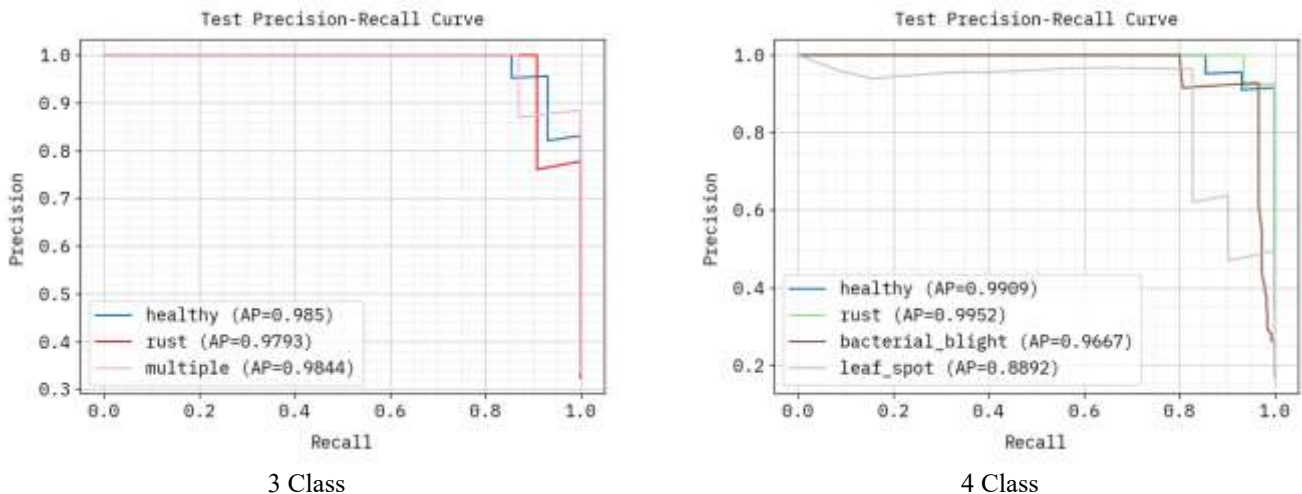


Figure 12. Precision recall curve analysis

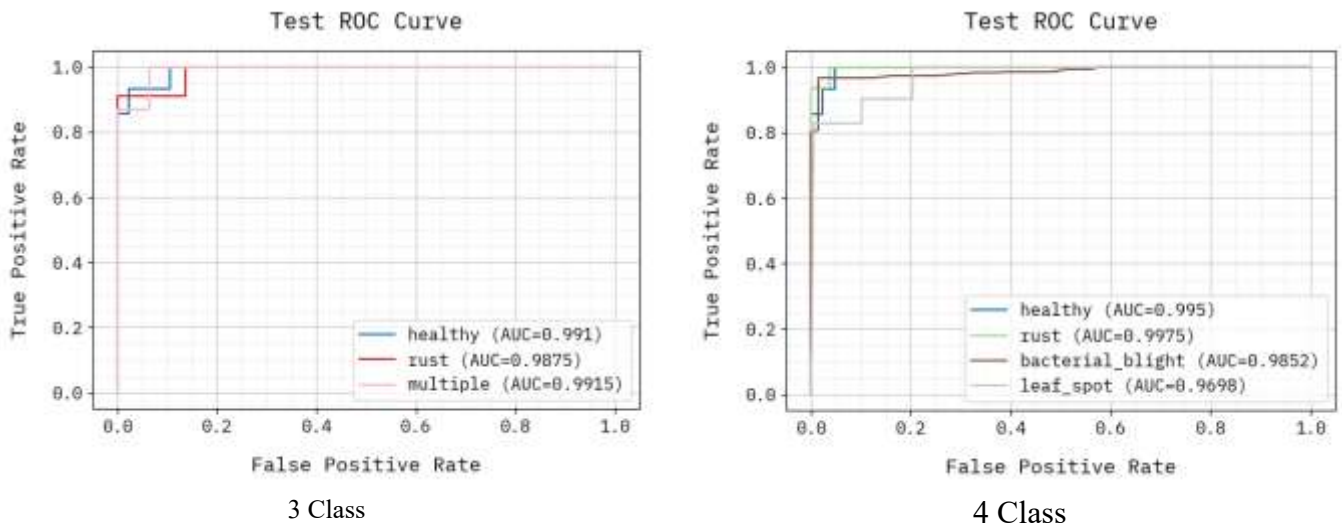


Figure 13. ROC curve analysis

4.13 Ablation analysis of the proposed ISSAE–PPBO framework

To demonstrate the contribution of each module within the proposed classification pipeline, an ablation study was conducted by evaluating three configurations:

- (i) Baseline SSAE,
- (ii) ISSAE without PPBO tuning, and
- (iii) Full ISSAE–PPBO model.

The baseline SSAE was trained with fixed sparsity and learning parameters, representing the conventional stacked sparse autoencoder. In the ISSAE variant, the adaptive sparsity adjustment and progressive fine-tuning were enabled, while the optimization stage was disabled. The final configuration integrated PPBO-driven hyperparameter refinement. The results in Table 4 indicate that ISSAE alone provides improved feature separation due to adaptive sparsity enforcement and layer-wise refinement.

When PPBO optimization is introduced, further improvement is achieved in classification accuracy, recall, and training stability. These findings confirm that each component contributes independently to the overall performance and that the complete ISSAE–PPBO framework provides the best detection capability. The baseline SSAE achieves moderate performance due to its fixed sparsity constraint and sensitivity to initial parameter selection. When adaptive sparsity control and progressive layer-wise refinements are introduced (ISSAE), the model shows clear improvement across all metrics, indicating enhanced feature learning. The complete

ISSAE–PPBO model achieves the highest performance, confirming that PPBO-driven hyperparameter tuning further stabilizes learning, minimizes reconstruction errors, and enhances the classifier’s ability to separate healthy and diseased leaf classes.

4.14 PPBO effectiveness analysis

To validate the role of the Paper Publishing Based Optimization (PPBO) algorithm in improving the stability and discriminative capability of ISSAE, an additional experiment was conducted by comparing manually tuned ISSAE with PPBO-optimized ISSAE. Manual tuning relied on trial-and-error selection of sparsity penalty, learning rate, and hidden neuron count, while PPBO automatically identified optimal values through its exploration–exploitation mechanism. The comparison results are summarized in Table 5.

The manually tuned ISSAE baseline achieves good performance but remains sensitive to hyperparameters, resulting in lower recall and higher training variance. The PPBO-optimized model consistently outperforms manual tuning across all metrics, with notable improvements in accuracy (+2.25%), precision (+3.20%), and recall (+3.22%). Additionally, the significant reduction in training variance (from 0.021 to 0.008) demonstrates that PPBO provides more stable convergence by guiding ISSAE toward well-generalized hyperparameter configurations. These results confirm that PPBO contributes meaningfully to enhancing model robustness and classification reliability.

Table 4. Ablation experimentation analysis

Model Variant	Accuracy (%)	Precision (%)	Recall (%)	F1 score (%)
Baseline SSAE (No improvements)	91.84	90.92	89.77	90.34
ISSAE without PPBO (Adaptive sparsity + fine-tuning only)	94.27	93.88	93.11	93.49
Proposed ISSAE–PPBO (Full model)	96.52	97.08	96.33	96.70

Table 5. Manual tuning vs. PPBO optimization

Model Configuration	Accuracy (%)	Precision (%)	Recall (%)	F1 score (%)	Training Variance*
ISSAE (Manual tuning)	94.27	93.88	93.11	93.49	0.021
ISSAE-PPBO (Optimized)	96.52	97.08	96.33	96.70	0.008

4.15 Parameter sensitivity analysis

The robustness of the ISSAE–PPBO model was further examined by analysing the sensitivity of three key hyperparameters: sparsity penalty (λ), learning rate (η), and the number of hidden neurons (H). Each parameter was varied around its PPBO-selected optimal value while keeping the others fixed. Table 6 summarises the variations tested and their corresponding accuracy values.

Table 6. Sensitivity of ISSAE–PPBO to key hyperparameters

Parameter Varied	Value Tested	Accuracy (%)
Sparsity Penalty (λ)	0.005	94.81
	0.010 (optimal)	96.52
	0.020	95.37
	0.040	93.92
Learning Rate (η)	0.0005	94.26
	0.001 (optimal)	96.52
	0.005	95.03
	0.010	92.74
Hidden Neurons (H)	64	94.58
	128 (optimal)	96.52
	256	95.49
	512	93.68

The results show that the ISSAE–PPBO model achieves peak performance at the PPBO-identified hyperparameter settings. Deviations from the optimal sparsity penalty cause either excessive neuron activation (λ too low) or loss of meaningful features (λ too high), both reducing accuracy. Similarly, learning rates smaller than the optimal slow convergence, while higher values destabilize training. The hidden neuron count demonstrates a threshold effect in which both under-parameterization ($H = 64$) and over-parameterization ($H \geq 256$) degrade performance due to insufficient feature representation or overfitting. This analysis confirms that PPBO plays a critical role in selecting balanced hyperparameters that maximize classification accuracy and training stability.

4.16 Computational feasibility

The ISSAE–PPBO model was benchmarked on two hardware platforms to assess its suitability for on-site agricultural screening and results are provided in Table 7.

Table 7. Computational evaluation results

Metric	Laptop CPU (i5-8250U)	Mobile SoC (Snapdragon 720G)
Model Size	7.4 MB	7.4 MB
FLOPs per Inference	11.2 MFLOPs	11.2 MFLOPs
Avg. Inference Time	23.8 ms	41.5 ms
Frames Per Second	42 FPS	24 FPS

The results confirm that the model is computationally lightweight and capable of real-time or near-real-time processing on standard CPU and smartphone hardware.

4.17 Robustness towards simulated agricultural conditions

To approximate real field variability, synthetic distortions were applied, including illumination shifts, shadow masks,

and partial leaf occlusion and results are provided in Table 8.

The model demonstrates strong resilience to common environmental disturbances such as uneven lighting and background interference, supporting its potential for field-level use.

Table 8. Robustness evaluation

Condition	Accuracy (%)	F1 score (%)
Normal dataset	96.52	96.70
Low light	93.41	93.88
High light	94.26	94.74
Leaf occlusion (15%)	92.37	92.84
Background clutter	94.88	95.03

4.18 Edge-device deployment feasibility

The model was further tested on a low-cost edge device to assess its practicality in farm environments and results depicted in Table 9.

Table 9. Raspberry Pi deployment results

Metric	Result
Inference Time	79.2 ms
Frames Per Second	12.6 FPS
CPU Utilization	78%

The model operates at usable speed (>10 FPS) even on resource-limited edge hardware, suggesting feasibility for farm-side kiosks or portable detection units.

4.19 Diagnostic evaluation of the proposed ISSAE–PPBO model

To ensure reliable performance evaluation under class-imbalance conditions, additional diagnostic metrics—sensitivity, specificity, and AUC—were computed alongside precision, recall, and F1 score. Sensitivity quantifies the model’s ability to correctly detect diseased leaves, whereas specificity measures its ability to correctly identify healthy samples. AUC provides a threshold-independent estimate of the system’s discrimination capability. The results for the ISSAE–PPBO model are presented in Table 10.

Table 10. Diagnostic evaluation of the proposed ISSAE–PPBO model

Metric	Value (%)
Accuracy	96.52
Precision	97.08
Recall (Sensitivity)	96.33
Specificity	95.87
F1 score	96.70
AUC	0.972

As seen from Table 10, the proposed model achieves high sensitivity (96.33%), indicating strong capability in identifying diseased leaves even under class-imbalance conditions. The specificity of 95.87% confirms that healthy leaves are also reliably classified, minimizing unnecessary alarms. The AUC value of 0.972 further demonstrates strong threshold-independent discrimination. Together, these diagnostic metrics validate that the ISSAE–PPBO framework maintains balanced performance across all classes despite variations in disease prevalence.

4.20 Discussion

The comprehensive performance evaluation in terms of MCC, F1 score, specificity, accuracy, sensitivity, precision, as well as FNR amply demonstrates the superiority related to the proposed ISSAE-PPBO technique for identifying and categorizing diseases of night jasmine leaves. ISSAE-PPBO continuously outperforms MULTINET, custom CNN, SEVRNet, as well as LF-Mamba by significant margins in entire ten iterations (20–200), achieving higher accuracy (up to 0.9497), enhanced sensitivity (0.8874), enhanced precision (0.9203), and higher MCC (0.8540), as well as superior F1 scores (0.8980) and specificity (0.9641). Most significantly, its exceptionally minimal FNR (as low as 0.0359) demonstrates its exceptional capacity to avoid missed detections, which is essential for controlling the spread of disease in agricultural settings. ISSAE-PPBO has good learning as well as generalization capabilities, with improvement ranges of around 5% to 17% across many measures when compared to alternative approaches. These results highlight ISSAE-PPBO as a very promising as well as successful precision agricultural tool, offering accurate, reliable, and effective detection to safeguard crop health and improve disease control strategies

5. CONCLUSIONS

This research used a novel optimization technique centered on deep learning to detect as well as classify night jasmine leaf disease. The first source of the data was the Night Jasmine Leaf Database, a Kaggle resource. Bilateral filtering, median filtering, wiener filtering, as well as CLAHE methods were used to pre-process the collected data. This pre-processed data was segmented utilizing the adaptive threshold-based segmentation approach. This segmented data was now subjected to feature extraction by the VGG-16 method. Finally, night jasmine leaf disease was identified and categorized using the new ISSAE method. The optimization technique known as PPBO, which considered the maximization of returning accuracy as the fitness function, was used to change the parameters of standard SSAE. According to the experimental findings, the recommended approach provided a more efficient way to recognize as well as classify illnesses of night jasmine leaves. For the diagnosis as well as classification of night jasmine leaf disease, the proposed ISSAE-PPBO model performed 10.80% and 12.61% better than remaining existing techniques, respectively. Since the study uses only publicly available image datasets, no personal or farmer-specific information is collected, and all data usage complies with standard research permissions. Future deployment of the model will require formal data-usage agreements, clear accountability frameworks for AI-assisted decisions, and ethical safeguards to mitigate risks associated with misdiagnosis in agricultural settings. Future work will integrate pathogen-specific labels, infection-stage information, and expert annotations through collaboration with plant-pathology specialists to enhance biological interpretability and clinical relevance. Also, Future work will incorporate cost-sensitive evaluation frameworks where false negatives—posing higher agricultural risk—are penalized more heavily than false positives to better reflect real-world disease management priorities.

REFERENCES

- [1] Lu, J., Tan, L., Jiang, H. (2021). Review on Convolutional Neural Network (CNN) applied to plant leaf disease classification. *Agriculture*, 11(8): 707-715. <https://doi.org/10.3390/agriculture1108070>
- [2] Atila, Ü., Uçar, M., Akyol, K., Uçar, E. (2021). Plant leaf disease classification using EfficientNet deep learning model. *Ecological Informatics*, 61: 101182. <https://doi.org/10.1016/j.ecoinf.2020.101182>
- [3] Bankina, B., Bimšteine, G., Kaņeps, J., Plūduma-Pauniņa, I., Gaile, Z., Paura, L., Stoddard, F.L. (2021). Discrimination of leaf diseases affecting faba bean (*Vicia faba*). *Acta Agriculturae Scandinavica, Section B—Soil & Plant Science*, 71(5): 399-407. <https://doi.org/10.1080/09064710.2021.1903985>
- [4] Islam, A., Islam, R., Haque, S.R., Islam, S.M., Khan, M.A.I. (2021). Rice leaf disease recognition using local threshold based segmentation and deep CNN. *International Journal of Intelligent Systems and Applications*, 14(5): 35-45. <https://doi.org/10.5815/ijisa.2021.05.04>
- [5] Metre, V.A., Sawarkar, S.D. (2021). Research review on plant leaf disease detection utilizing swarm intelligence. *Turkish Journal of Computer and Mathematics Education*, 12(10): 177-185. <https://turcomat.org/index.php/turkbilmater/article/view/4075>
- [6] Zhou, C., Zhang, Z., Zhou, S., Xing, J., Wu, Q., Song, J. (2021). Grape leaf spot identification under limited samples by fine grained-GAN. *IEEE Access*, 9: 100480-100489. <https://doi.org/10.1109/ACCESS.2021.3097050>
- [7] Manoharan, J.S., Braveen, M., Subramanian, G.G. (2021). A hybrid approach to accelerate the classification accuracy of cervical cancer data with class imbalance problems. *International Journal of data mining and Bioinformatics*, 25(3-4): 234-261. <https://doi.org/10.1504/IJDMB.2021.12286>
- [8] Malik, A., Vaidya, G., Jagota, V., Eswaran, S., et al. (2022). Design and evaluation of a hybrid technique for detecting sunflower leaf disease using deep learning approach. *Journal of Food Quality*, 2022(1): 9211700. <https://doi.org/10.1155/2022/9211700>
- [9] Storey, G., Meng, Q., Li, B. (2022). Leaf disease segmentation and detection in apple orchards for precise smart spraying in sustainable agriculture. *Sustainability*, 14(3): 1458. <https://doi.org/10.3390/su14031458>
- [10] Ashokkumar, K., Dharshini, M., Janani, T., Shrivani Sri, V., Subhasidha, R. (2024). *Nyctanthes arbor-tristis* Linn.(Night Jasmine): Extraction techniques, phytochemical constituents, and biological impacts of extracts and essential oil. *Future Journal of Pharmaceutical Sciences*, 10(1): 117. <https://doi.org/10.1186/s43094-024-00694-2>
- [11] Yogeshwari, M., Thailambal, G. (2023). Automatic feature extraction and detection of plant leaf disease using GLCM features and convolutional neural networks. *Materials Today: Proceedings*, 81: 530-536. <https://doi.org/10.1016/j.matpr.2021.03.700>
- [12] Uğuz, S., Uysal, N. (2021). Classification of olive leaf diseases using deep Convolutional Neural Networks. *Neural Computing and Applications*, 33(9): 4133-4149. <https://doi.org/10.1007/s00521-020-05235-5>

- [13] Thorat, Y.S., Tiwari, B.D., Paralkar, S.D., Lokhande, D.S. et al. (2023). Studies of antipyretic activity of leaf extract of *nyctanthes arbor-tristis* linn. (night jasmine). *International Journal of Novel Research and Development (IJNRD)*, 8(11): 66-69. <https://ijnrd.org/papers/IJNRD2311307.pdf>.
- [14] Chelloug, S.A., Alkanhel, R., Muthanna, M.S.A., Aziz, A., Muthanna, A. (2023). Multinet: A multi-agent DRL and efficientnet assisted framework for 3d plant leaf disease identification and severity quantification. *IEEE Access*, 11: 86770-86789. <https://doi.org/10.1109/ACCESS.2023.3303868>
- [15] Hosny, K.M., El-Hady, W.M., Samy, F.M., Vrochidou, E., Papakostas, G.A. (2023). Multi-class classification of plant leaf diseases using feature fusion of deep convolutional neural network and local binary pattern. *IEEE Access*, 11: 62307-62317. <https://doi.org/10.1109/ACCESS.2023.3286730>
- [16] Xiao, Z., Shi, Y., Zhu, G., Xiong, J., Wu, J. (2023). Leaf disease detection based on lightweight deep residual network and attention mechanism. *IEEE Access*, 11: 48248-48258. <https://doi.org/10.1109/ACCESS.2023.3272985>
- [17] Vijayalakshmi, S., Manoharan, J.S., Nivetha, B., Sathiya, A. (2025). Multi-task deep learning framework combining CNN: Vision transformers and PSO for accurate diabetic retinopathy diagnosis and lesion localization. *Scientific Reports*, 15(1): 35076. <https://doi.org/10.1038/s41598-025-18742-z>
- [18] Arivumani, S.S., Nagarajan, M. (2024). Adaptive convolutional-LSTM neural network with NADAM optimization for intrusion detection in underwater IoT wireless sensor networks. *Engineering Research Express*, 6(3): 035243. <https://doi.org/10.1088/2631-8695/ad7935>
- [19] Shrotriya, A., Sharma, A.K., Prabhu, S., Bairwa, A.K. (2024). An approach toward classifying plant-leaf diseases and comparisons with the conventional classification. *IEEE Access*, 12: 117379-117398. <https://doi.org/10.1109/ACCESS.2024.3411013>
- [20] Polly, R., Devi, E.A. (2024). Semantic segmentation for plant leaf disease classification and damage detection: A deep learning approach. *Smart Agricultural Technology*, 9: 100526. <https://doi.org/10.1016/j.atech.2024.100526>.
- [21] Parekh, S., Soni, A., Soni, A., Sharma, P. (2018). Preliminary phytochemical screening and quantitative analysis of crude extracts of *nyctanthes arbor-tristis* indigenous to South Gujarat region. *International Journal of Life Sciences Research*. 6(3): 25-33. <http://doi.org/10.2139/ssrn.5280376>
- [22] Rahman, K.N., Banik, S.C., Islam, R., Al Fahim, A. (2025). A real time monitoring system for accurate plant leaves disease detection using deep learning. *Crop Design*, 4(1): 100092. <https://doi.org/10.1016/j.crope.2024.100092>
- [23] Wang, X., Tang, S.H., Ariffin, M.K.A.B.M., Ismail, M. I.S.B., Zhao, R. (2025). LeafMamba: A novel IoT-integrated network for accurate and efficient plant leaf disease detection. *Alexandria Engineering Journal*, 123: 415-424. <https://doi.org/10.1016/j.aej.2025.03.033>