[1] A. S. Devi

[2] S. Albert Antony Raj

Enhancing Betel Leaf Disease Detection Integrating Dcnn and Rpo Optimization for Accurate Classification

Journal of Electrical Systems

Abstract: - Betel leaf, renowned for its cultural and medicinal importance, faces severe threats from various diseases, jeopardizing its cultivation and economic viability. Detecting diseases in betel leaves presents challenges hindering accurate diagnosis. This study addresses these challenges using a comprehensive approach. Initially, raw images are standardized to a pixel size and subjected to pre-processing using Contrast Limited Adaptive Histogram Equalization (CLAHE). Subsequently, segmentation techniques, including Gaussian Mixture Model (GMM) and Fuzzy C-Means (FCM), partition the images into meaningful segments based on intensity, color, and texture criteria, facilitating the extraction of relevant information and delineation of regions of interest. The segmented images undergo data augmentation to ensure balanced representation across classes and enhance model robustness through techniques like horizontal and vertical flipping. Feature extraction is performed using the Gray-Level Co-occurrence Matrix (GLCM) method on augmented images. Extracted features are then inputted into a classification phase, utilizing a Deep Convolutional Neural Network (DCNN) optimized by the Red Panda Optimization (RPO) algorithm. Through this methodology, the study seeks to improve disease detection accuracy in betel leaves, contributing to the preservation of their cultivation and economic significance. Experimental results demonstrate the effectiveness of the proposed approach in accurately classifying betel leaf diseases.

Keywords: Betel Leaf Disease Detection; CLAHE; FCM; GLCM; DCNN; RPO

1. Introduction

Betel leaf, renowned for its cultural significance and medicinal properties, faces significant threats from various diseases that threaten its cultivation and economic viability. From ancient times, betel leaves have held a prominent place in Indian traditions and folklore, cherished for their mouth-freshening qualities and therapeutic benefits. However, the cultivation of betel leaves is beset by numerous challenges, chief among them being the prevalence of diseases that afflict these plants. Among the most common diseases affecting betel leaves are leaf rot, foot rot, powdery mildew, and fusarium wilt. These diseases pose a substantial threat to betel leaf cultivation, causing extensive damage to crops and diminishing yields. Leaf rot and foot rot, in particular, are widespread infections that can devastate entire plantations if left unchecked. With betel leaf cultivation being a vital economic activity in regions like Bangladesh, where it is the second-largest producer globally, and the impact of these diseases extends beyond individual farmers to broader agricultural economies. Leaf rot, characterized by rapid deterioration and discoloration of betel leaves, is a highly contagious disease that spreads swiftly, especially in humid conditions. Affected leaves exhibit waterlogged spots and patches, gradually turning yellowish as the disease progresses.

Similarly, foot rot, another prevalent infection, manifests as dark lesions on the stems and leaves, leading to wilting and decay of the plant. These diseases not only compromise the health and vitality of betel leaf plants but also pose significant challenges for farmers striving to maintain sustainable cultivation practices. Efficient detection and management of betel leaf diseases are essential for preserving crop health and ensuring the continuity of betel leaf cultivation. Timely identification of diseased plants enables farmers to implement appropriate control measures, preventing the spread of infections and minimizing crop losses. However, traditional methods of disease detection often rely on visual inspection, which may not be sufficiently accurate or timely to address the dynamic nature of plant diseases. In response to these challenges, researchers and agricultural experts are increasingly turning to advanced technologies, such as machine learning, image processing, and Internet of Things (IoT), to revolutionize disease detection and management in agriculture.

[1,2] Department of Computer Applications, Faculty of Science and Humanities,

SRM Institute of Science and Technology, Kattankulathur-603 203, Tamil Nadu, India

These technologies offer innovative solutions for automating disease detection processes, enabling early identification of infections and facilitating targeted interventions to mitigate their impact. The motivation behind betel leaf disease detection stems from the important need to safeguard the economic viability and cultural significance of betel leaf cultivation. With betel leaf being a crucial crop in regions like Bangladesh, where it plays a vital role in the agricultural economy, effective disease detection is essential for minimizing crop losses and ensuring food security. By employing advanced technologies for early detection and management of diseases, we can empower farmers to protect their livelihoods and preserve the rich cultural heritage associated with betel leaf cultivation.

The contributions of this paper are manifested below,

- Utilization of CLAHE for pre-processing enhances image quality by improving contrast and reducing noise, laying the groundwork for effective disease detection.
- Implementation of GMM and FCM segmentation techniques aids in partitioning images into meaningful regions based on intensity, color, or texture, facilitating the extraction of relevant information and identification of regions of interest.
- Through augmentation techniques like horizontal and vertical flipping, the dataset's robustness is enhanced, preventing overfitting and enabling exposure to a diverse range of scenarios, thereby improving model performance across all classes.
- The study employs GLCM for feature extraction from augmented images, capturing textural information that is instrumental in disease identification and classification.
- Leveraging DCNN architecture optimized by RPO algorithm facilitates accurate image classification based on extracted features, harnessing the combined power of advanced neural networks and optimization techniques for precise disease diagnosis.

The rest of this paper is organized as follows. The section II provides both related works and problem statement. The proposed methodology is explained in the section III. The result and discussion is then presented in the section IV, followed by the conclusion in the section V.

2. Literature Review

In 2024, Kusuma and Jothi [16] developed and refined for centuries, the use of betel leaves spans traditional medicine practices and culinary traditions across various cultures. However, the threat of plant diseases poses a significant risk to the availability of this valuable resource. To address this challenge, advanced technologies such as deep learning models have been leveraged to swiftly identify and mitigate potential disease outbreaks.

Adaptive Neuro-Fuzzy Inference System (ANFIS) classifiers and Artificial Neural Network (ANN) were utilized by Dey *et al.* [17] in 2022 to estimate betel leaf pathogens. While the ANFIS architecture uses IF-THEN-IF fuzzy rules to analyze the effect of color spaces on recognition metrics, the suggested ANN has two hidden layers.

To ensure industry quality, safety, and healthy development, Hridoy *et al.* [18] introduced an efficient deep neural network-based detection approach in 2022 that can quickly diagnose betel plant problems. Transfer learning allows deep learning models to be generalized using a dataset of 10,662 photos of betel leaves.

In 2023, Shovon *et al.* [19] proposed PlantDet, a robust deep ensemble model integrating various techniques such as data augmentation, preprocessing, and regularization. PlantDet, based on InceptionResNetV2, EfficientNetV2L, and Xception, effectively addresses underfitting and overfitting issues, ensuring high performance even with sparse datasets.

In 2023, Meenakshi [20] introduced a novel approach employing the Modified Logistic Regression algorithm for leaf disease identification in medicinal plants. By incorporating adaptive gamma correction, k-means clustering, and GLCM feature extraction, the proposed method achieves superior classification accuracy compared to existing logistic regression techniques.

In 2023, Prodhan *et al.* [21] used a novel method combining QuEChERS extraction with GC-FTD/ECD was developed and validated for quantifying eight pesticide residues in betel leaves. Optimal cleanup materials were

determined, achieving recoveries ranging from 86% to 108% with RSDr ≤ 9%. Matrix-matched calibration exhibited good linearity, with limits of detection and quantification below EU-MRLs. Matrix effects were assessed, identifying significant impacts for cypermethrin.

In 2023, Hridoy *et al.* [22] developed a hybrid approach using deep learning techniques, combining Xception and ResNet50 models through stacking ensemble learning. Tested on a diverse multi-plant dataset, the hybrid model achieved superior accuracy of 99.20%, surpassing individual CNN models. This approach demonstrates promising potential for precise multi-plant disease diagnosis.

2.1. Problem Statement

Betel leaf disease detection encounters several challenges that hinder accurate and efficient diagnosis. Firstly, farmers may lack awareness about the various diseases affecting betel leaves, making it difficult to recognize symptoms and diagnose diseases effectively. Additionally, symptoms of betel leaf diseases can vary widely, making it challenging to develop a standardized approach for detection. Factors such as the stage of infection, environmental conditions, and plant health contribute to this variability. Understanding the underlying pathogenesis of betel leaf diseases requires extensive research and knowledge of plant pathology, further complicating diagnosis. Moreover, traditional methods of disease diagnosis involving manual inspection of individual plants can be time-consuming, labor-intensive, and costly. Early detection is crucial for effective management and prevention of crop damage, but early-stage symptoms may be subtle and easily overlooked, leading to delayed diagnosis and treatment. Furthermore, betel leaf images used for disease detection may exhibit variability in terms of lighting conditions, background clutter, leaf orientation, and disease severity, affecting the performance of automated detection algorithms. Finally, farmers in certain regions may have limited access to resources such as advanced diagnostic tools, trained agronomists, and disease-resistant plant varieties, further complicating disease detection and management. Addressing these challenges requires the development of innovative solutions leveraging advancements in technology to improve the accuracy, speed, and accessibility of betel leaf disease detection methods.

3. Proposed Methodology

Betel leaf disease detection presents challenges due to variations in symptoms, lack of farmer awareness, and manual inspection limitations. Early diagnosis is crucial for effective management, but subtle symptoms and resource constraints hinder timely detection. Leveraging advanced technology like deep learning and image processing can enhance accuracy and speed in detecting diseases. Innovative solutions are needed to address these challenges and improve betel leaf disease diagnosis for sustainable agriculture. This work developed a novel approach by combining DCNN with RPO algorithm.

3.1. Pre-Processing

In this study, raw images are initially resized to a standardized pixel size of 250x250. Following this, pre-processing techniques are applied using CLAHE.

3.1.1. CLAHE Image Enhancement

CLAHE is utilized to enhance the contrast and improve the visual quality of the images. Unlike traditional histogram equalization methods that operate globally on the entire image, CLAHE applies adaptive histogram equalization (AHE) locally to different regions of the image. Fig. 1 illustrates the overall architecture of the proposed system, depicting the flow and components of the solution.

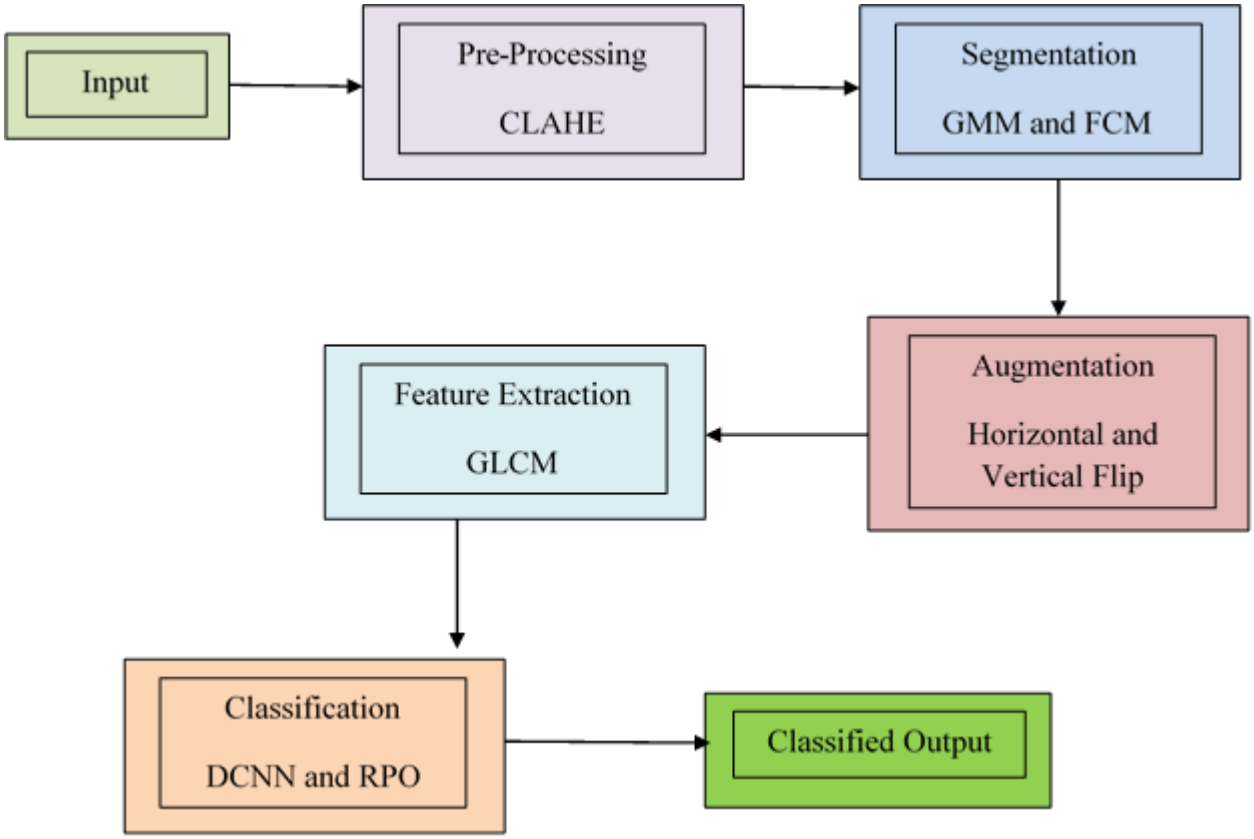


Figure 1: Overall Proposed Architecture

This allows for better preservation of details and avoids over-amplification of noise, which can occur with global methods. By applying histogram equalization locally, CLAHE is able to enhance the contrast of different regions of the image independently, leading to improved visibility of details and textures without introducing unwanted artifacts. It is particularly useful for images with varying lighting conditions or regions of low contrast. The steps followed in CLAHE are given below,

- Image Division

The original image is divided into smaller, non-overlapping blocks or tiles.

- Local Histogram Calculation

A histogram is computed for each block, representing the distribution of pixel intensities within that region.

- Contrast Limiting

To prevent excessive amplification of contrast, CLAHE applies a contrast limiting mechanism. This means that the enhancement is restricted within each block to ensure that the resulting image does not appear overly sharpened or noisy.

- Histogram Equalization

Histogram equalization is applied independently to each block. This process redistributes the pixel intensities within each block to achieve a more uniform distribution, thereby enhancing the local contrast.

- Interpolation

After histogram equalization, interpolation techniques are often used to blend the enhanced blocks seamlessly, reducing the appearance of artificial boundaries between adjacent regions. The pre-processed images undergo segmentation in the subsequent phase.

3.2. Segmentation

During segmentation, the image is partitioned into meaningful regions or segments based on certain criteria such as intensity, color, or texture. This process aims to extract relevant information from the image and delineate distinct objects or regions of interest. In this work, segmentation is performed using two distinct methods: GMM and FCM.

3.2.1. GMM

The distribution of data points is represented by the probabilistic GMM model as a combination of several Gaussian distributions. In image segmentation, GMM assigns each pixel to one of several clusters based on the probability that the pixel belongs to each cluster. This method assumes that the intensity values within each segment follow a Gaussian distribution. By fitting a mixture of Gaussian distributions to the pixel intensities, GMM effectively separates the image into distinct regions or classes. GMM is a powerful probabilistic modeling technique utilized in image segmentation, offering several advantages:

Probabilistic Approach: GMM assigns each pixel in an image a probability of belonging to different clusters, enabling a probabilistic interpretation of segmentation results.

Modeling Complex Distributions: GMM can effectively model complex data distributions in images, making it suitable for capturing variations and patterns present in the image data.

Robustness to Noise and Outliers: Due to its probabilistic nature, GMM is more robust against noise and outliers compared to deterministic segmentation techniques.

GMM constructs a mixture model representing the image as a combination of Gaussian distributions. In a segmentation problem where the goal is to separate objects from the background, the image distribution can be expressed as a mixture of two Gaussian distributions as per Eq. (1).

$$p(i) = p(i|r_o) + p(i|r_b) \quad (1)$$

Where $p(i|r_o), p(i|r_b)$ is represented as per Eq. (2) and Eq. (3).

$$p(i|r_o) \sim N(\mu_o, \Sigma_o) \quad (2)$$

$$p(i|r_b) \sim N(\mu_b, \Sigma_b) \quad (3)$$

Where, the background's parameters are μ_b and Σ_b, and the object's distribution's mean and variance are μ_o and Σ_o, respectively.

3.2.2. FCM

A set of N elements, denoted as $X = \{X_1, X_2, \ldots, X_N\}$, is divided into C clusters by the FCM algorithm according to specified criteria. Every element $X_i \epsilon X$, where i is a vector in the range of $1\ to\ N$, is d-dimensional. The cluster centers $v_1, v_2, \ldots, v_C$ are defined by the algorithm. The approach gives each element in each cluster a membership value by using a representative matrix U. Values ranging from 0 to 1, which indicate the degree of confidence that an element belongs to a certain cluster, are among matrix U properties. Using a probabilistic method, heterogeneous datasets can be effectively clustered.

- The membership value of element X_i in a cluster with center v_k is denoted as $U(i,k)$, where $1 \leq i \leq N;\ 1 \leq k \leq C$
- For every X_i, $0 \leq U(i,k) \leq 1, 1 \leq i \leq N;\ 1 \leq k \leq C$ and $\sum_{k=1}^{C} \quad U(i,k) = 1$
- The degree of confidence that element X_i belongs to the cluster k increases with increasing $U(i,k)$. According to Eq. (4), an objective function is created in a way that the clustering method minimizes it.

$$A(U,v) = \sum_{i=1}^{N} \quad \sum_{k=1}^{C} \quad U(i,k)^m d(i,k)^2 \quad (4)$$

where m is the algorithm's fuzzification coefficient and $[\![d(i,k)^2 = \|X_i - v_k\|^2$ is the squared distance between the element X_i and cluster center v_k.

3.3. Image Augmentation

In the data augmentation phase, the segmented images are utilized as input. Each class consists of 411 images, ensuring a balanced representation across classes. Using methods for augmenting data, such as flipping data horizontally and vertically. Through a greater variety of scenarios, this method seeks to improve the model's robustness and prevent overfitting. By ensuring each class has a sufficient number of samples, the model can better learn the underlying patterns and improve its performance across all classes.

3.3.1. Horizontal and Vertical Flip

Horizontal and vertical flipping are common data augmentation techniques used to increase the diversity of training data in machine learning and computer vision tasks, particularly in image classification. Horizontal flipping, also known as left-right flipping, involves mirroring an image along its vertical axis. This transformation results in a new image where the content appears as if viewed in a mirror. If an image contains a person facing left, horizontal flipping will produce an image of the same person facing right. This augmentation technique helps the model learn invariant features that are symmetric with respect to left-right orientation.

Vertical flipping, on the other hand, mirrors the image along its horizontal axis, effectively flipping it upside down. This transformation can be beneficial in scenarios where objects in the images have inherent vertical symmetry or when variations in the orientation of objects are expected. Both horizontal and vertical flipping are simple yet effective techniques for augmenting training datasets, improving model generalization, and reducing overfitting by introducing variations in the training data without altering the underlying semantics of the images.

3.4. Feature Extraction

In the feature extraction phase, the augmented images undergo feature extraction using Gray-Level Co-occurrence Matrix (GLCM) method.

3.4.1. GLCM

GLCM analyzes image texture by capturing pixel value relationships, derived from grayscale conversion. Spatial features are extracted using a kernel, generating a matrix for texture information. Key features include Contrast, Homogeneity, Energy, and Correlation.

Contrast: The spatial frequency or local fluctuations in the image are measured by contrast. It measures the variation in the GLCM's highest and lowest pixel values. Greater local textural differences in the image are indicated by a higher contrast value. Equation (5) is used in this feature.

Homogeneity: The inverse difference moment, or homogeneity, quantifies how closely the elements in the GLCM are distributed. It is sensitive to the presence of near-diagonal elements and increases when the elements in the image are similar. High homogeneity values indicate uniformity in texture. This feature is using Eq. (6).

Energy: Energy, computed as the square root of the angular second moment, represents the uniformity or repeatability of pixel pairs in the image. Higher energy values indicate a more orderly distribution of pixel pairs and greater textural uniformity. This feature is using Eq. (7).

Correlation: The linear relationships between the image's gray tones are measured by correlation. It shows the extent to which pixel brightness values are connected throughout the image. Strong linear correlations between pixel values are shown by high correlation values, which point to a more consistent texture pattern. This feature is using Eq. (8).

$$contrast = \sum_{i=0}^{B-1} \quad \sum_{j=0}^{B-1} \quad (i-j)^2 \tag{5}$$

$$homogeneity = \sum_{i=0}^{B-1} \quad \sum_{j=0}^{B-1} \quad \frac{Z(i,j)}{1+(i-j)^2} \tag{6}$$

$$energy = \sqrt{\sum_{i=0}^{B-1} \quad \sum_{j=0}^{B-1} \quad Z(i,j)^2} \tag{7}$$

$$correlation = \sum_{i=0}^{B-1} \quad \sum_{j=0}^{B-1} \quad \frac{(i-\mu_i)(j-\mu_j)}{\sqrt{(\sigma_i)(\sigma_j)}} \tag{8}$$

The normalized gray-scale value at position (i, j) within the kernel is indicated by $Z(i, j)$, and the number of gray levels is represented by B. This ensures that the sum is 1. Utilizing rotations at 0°, 45°, 90°, and 135° for every feature, textural features were extracted from grayscale images.

3.5. Image Classification

In this study, the extracted features are inputted into the image classification phase. Classification is conducted using DCNN optimized by RPO algorithm. This process aims to accurately classify images based on the extracted features, leveraging the power of DCNN and the optimization capabilities of the RPO.

3.5.1. DCNN

DCNN is a type of artificial neural network particularly suited for analyzing visual data such as images. It consists of multiple layers, including convolutional layers, pooling layers, fully connected layers, and activation functions. Here's a brief explanation of each component:

Convolutional Layer: This layer applies convolution operations to the input data using learnable filters. The output of each filter, known as a feature map, captures specific patterns or features from the input. The convolution operation is represented as per Eq. (9).

$$Conv(i,j) = \sum_{m=0}^{m-1} \quad \sum_{n=0}^{n-1} \quad I(i+m, j+n) \times K(m,n) \tag{9}$$

Where I is the input matrix, K is the filter/kernel, and m and n are the dimensions of the filter.

Rectified Linear Unit (ReLU): ReLU is an activation function that introduces non-linearity to the network by replacing negative values with zero. It is defined as per Eq. (10).

$$relu(x) = max(0, x) \tag{10}$$

Pooling Layer: By lowering spatial dimensions, layer down samples the feature maps that were acquired from convolutional layers. A common pooling operation is max pooling, where the maximum value within each pooling window is retained. It helps in reducing computational complexity and controlling overfitting.

Fully Connected Layer: Also known as the dense layer, this layer connects every neuron from the previous layer to every neuron in the current layer. It learns complex patterns by combining features extracted from previous layers.

Softmax Layer: This layer is typically used as the output layer in classification tasks. It converts the raw scores (logits) from the previous layer into probabilities for each class using the Softmax function using Eq. (11).

$$SOp_x \frac{e^{z_x}}{\sum_{x=1}^{M} e^{z_x}} \tag{11}$$

Where, z is the vector of logits, z_x represents output-count, SOp_x denotes softmax output, and M represents the totality of output nodes.

3.5.2. RPO

The red panda, native to southern China and the eastern Himalayas, is a small mammal known for its reddish-brown fur and distinctive markings. It thrives in temperate forests with dense bamboo cover and is adept at climbing trees. Feeding mainly on bamboo leaves and shoots, it relies on keen senses and climbing abilities. The RPO approach's design is influenced by these organic characteristics.

3.5.2.1. Mathematical Modeling

3.5.2.1.1. Initialization

As a population-based metaheuristic algorithm, the RPO technique uses red pandas to symbolize each individual member. Depending on where it is in the search space, each red panda offers a potential value for a problem variable, acting as a candidate solution. Each red panda (or potential solution) is represented mathematically as a vector. Together, these red pandas create a matrix where each row is a potential solution and each column has values that could be offered for the associated problem variable. First, red panda coordinates in the search space are initialized at random using Eq. (12) and Eq. (13).

$$Y = [Y_1 \dots Y_i \dots Y_M]_{M\times n} = \left[Y_{1,1} \dots Y_{1,j} \dots Y_{1,n} \dots Y_{i,1} \dots Y_{i,j} \dots Y_{i,n} \dots Y_{M,1} \dots Y_{M,j} \dots Y_{M,n}\right]_{M\times n} \quad (12)$$

$$y_{i,j} = lob_j + r_{i,j} \cdot \left(upb_j - lob_j\right), i = 1,2, \dots, M, j = 1,2, \dots, n \quad (13)$$

The population matrix holding the red panda locations is represented by Y in the RPO technique, where Y_i stands for the ith red panda (possible solution) and $Y_{i,j}$ for its jth dimension (problem variable). M is the total number of red pandas, and n is the number of variables that are the problem. The jth problem variable's lower and upper limits are indicated by the variables lob_j and upb_j, respectively, and random integers $r_{i,j}$ inside the interval [0,1] are used. The positions of each red panda act as potential solutions, making it possible to assess the objective function associated with each one. A matrix of the form provided by Eq. (14) can be used to represent the final set of evaluated objective function values.

$$f = [f_1 \dots f_i \dots f_M]_{M\times 1} = [f(Y_1) \dots f(Y_i) \dots f(Y_M)]_{M\times 1} \quad (14)$$

The value obtained by the ith red panda is indicated by f_i, and f represents the vector of values of the objective function. These values of the objective function are essential for evaluating the caliber of potential solutions. The greatest and lowest values of the objective function are used to identify the best and worst potential solutions, respectively. These potential solutions are modified appropriately during every iteration. Iterative upgrades to potential solutions for the best possible problem-solving are part of the RPO's exploration and exploitation phases.

3.5.2.1.2. Phase 1: Exploration Strategy - Foraging

Red pandas' positions during the first stage of RPO resemble how they would forage in the wild. They excel in detecting and moving towards food sources using their keen senses. In the algorithm, each red panda considers the locations of others that yield superior objective function values as potential food sources. These proposed food positions are determined based on objective function value comparisons, with one position randomly chosen by each red panda using Eq. (15).

$$pfs_i = \{Y_k | k \in \{1,2,..,M\} and\ f_k < f_i\} \cup \{Y_{best}\} \quad (15)$$

Based on a comparison with the location of the best candidate solution Y_{best}, the suggested food sources for each red panda $pfs_i$are identified. Approaching these sources causes large positional shifts that improve ability of algorithm to globally search and explore. By determining new locations in relation to the food source (best candidate solution), red pandas' foraging behavior can be replicated. Eq. (16) and Eq. (17) are used to update the red panda's location if the objective function value improves at the new location.

$$Y_i^{p1}: y_{i,j}^{p1} = y_{i,j} + r.\left(sfs_{i,j} - Is.y_{i,j}\right) \quad (16)$$

$$Y_i = \{Y_i^{p1}, f_i^{p1} < f_i\ Y_i, else \quad (17)$$

The new location of the ith red panda as ascertained from the RPO's first phase is represented by Y_i^{p1}. Objective function is denoted by f_i^{p1}, and its position in the jth dimension is indicated by $y_{i,j}^{p1}$. For the ith red panda, sfs_i denotes the preferred food source, and $sfs_{i,j}$ denotes its location in the jth dimension. Is is a randomly chosen number from the set {1, 2}, and the variable r is a random value between 0 and 1.

3.5.2.1.3. Phase 2: Proficiency in ascending and perching on trees (exploitation)

In the second stage of the RPO, red pandas' ability to climb and rest on trees determines where they are positioned. Red pandas usually spend much of their time in repose on trees; they climb adjacent trees to obtain food after foraging on the ground. This behavior leads to minor positional changes, which improve the exploitation and local search capabilities of the RPO algorithm in areas of potential interest. Mathematically modeling this tree-climbing behavior involves calculating new positions for each red panda and replacing previous positions if the objective function improves and represented as per Eq. (18) and Eq. (19).

$$Y_{i,j}^{p2} = y_{i,j} + \frac{lob_j + r_{i,j}\cdot\left(upb_j - lob_j\right)}{t}, i = 1,2, \dots, M, j = 1,2, \dots, n, t = 1,2, \dots, T \quad (18)$$

$$Y_i = \{Y_i^{p2}, f_i^{p2} < f_i \; Y_i, else$$
(19)

The ith red panda's modified position, obtained from the second phase of RPO, is represented by Y_i^{p2}. Objective function is shown by f_i^{p2}, and its position in the jth dimension is indicated by $Y_{i,j}^{p2}$. A random number between 0 and 1 represents the variable r. The symbol t denotes the algorithm's iteration counter, whereas T stands for the maximum iterations.

4. Result and Discussion

The proposed model is implemented using Python. The proposed model demonstrates strong performance with an accuracy of 94.53%, indicating its overall correctness in classification. Additionally, it exhibits high precision 95.38% and recall 94.53%, signifying its ability to accurately identify positive instances and retrieve relevant results. The F1 score, a harmonic mean of precision and recall, is 94.47%, reflecting a balanced performance. Furthermore, achieving 100% specificity underscores the model's effectiveness in correctly identifying negative instances.

4.1. Graphical Representation

Fig. 2 compares the original and pre-processed images, demonstrating the effects of image enhancement techniques. Fig. 3 presents a confusion matrix, visually summarizing the performance of a classification model on training and testing datasets. Fig. 4 displays the dataset after augmentation, showcasing the increased diversity and volume of data. Conversely, Fig. 5 showcases the dataset before augmentation. Fig. 6 and Fig. 7 depict histograms after and before equalization, respectively, highlighting the impact of histogram equalization on image contrast. Fig. 8 provides a graphical representation of a model's architecture or workflow. Fig. 9 illustrates the segmentation process, outlining distinct regions within an image. Fig. 10 and Fig. 11 depict the training and testing accuracy, as well as the training and testing loss, respectively, over the model training process.

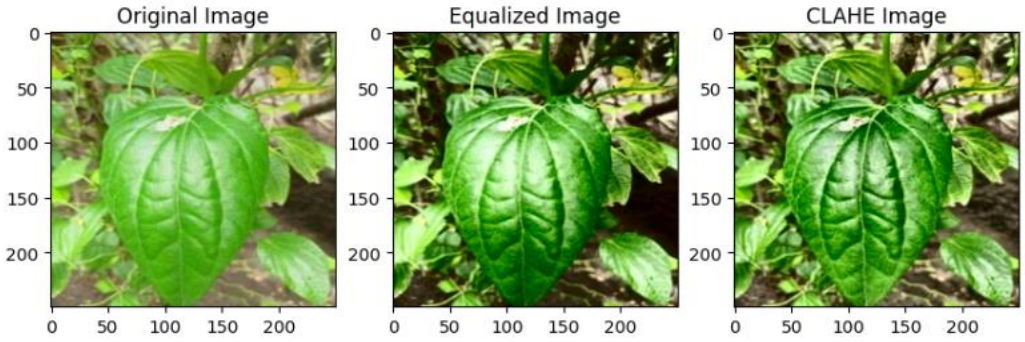
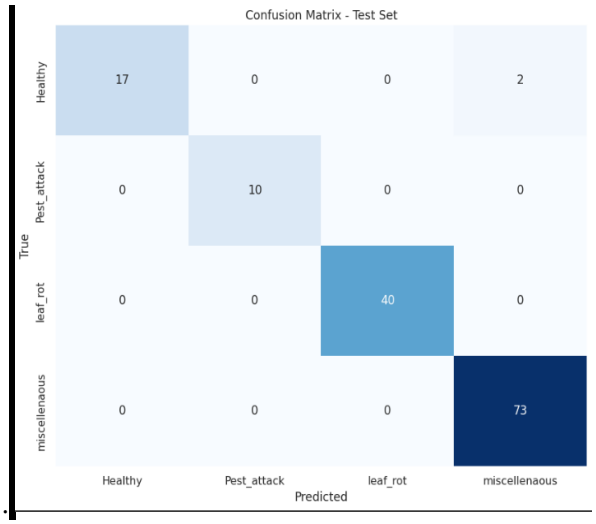


Figure 2: Comparison between Original and Pre-Processed Image



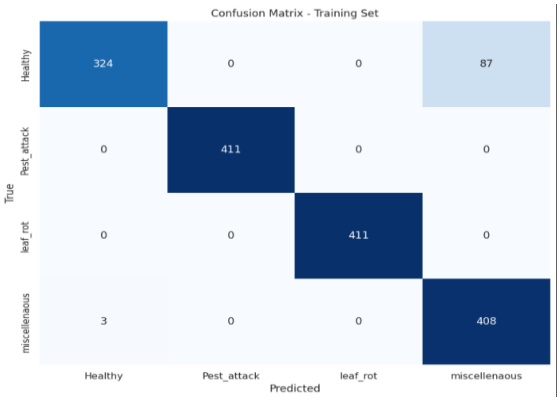


Figure 3: Confusion matrix illustrating training and testing sets

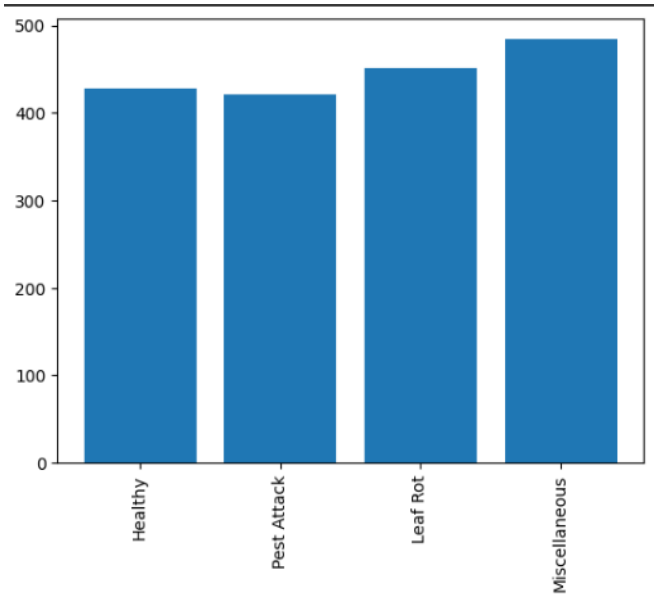


Figure 4: Dataset after Augmentation

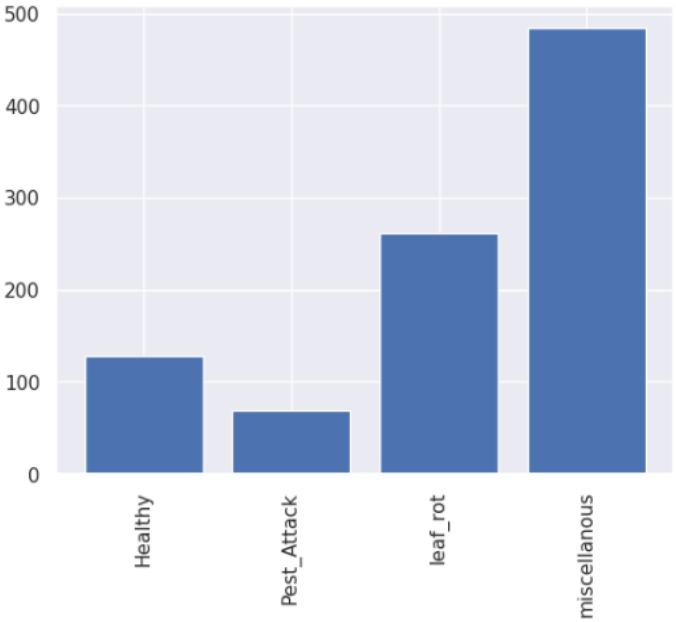


Figure 5: Dataset before Augmentation

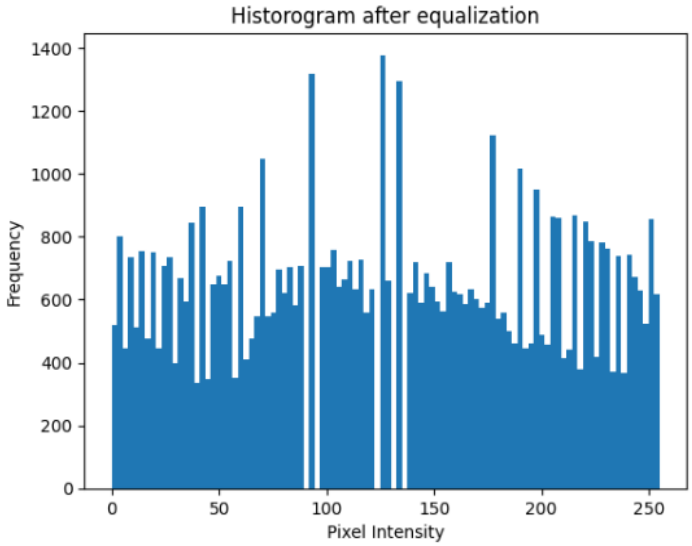


Figure 6: Histogram after Equalization

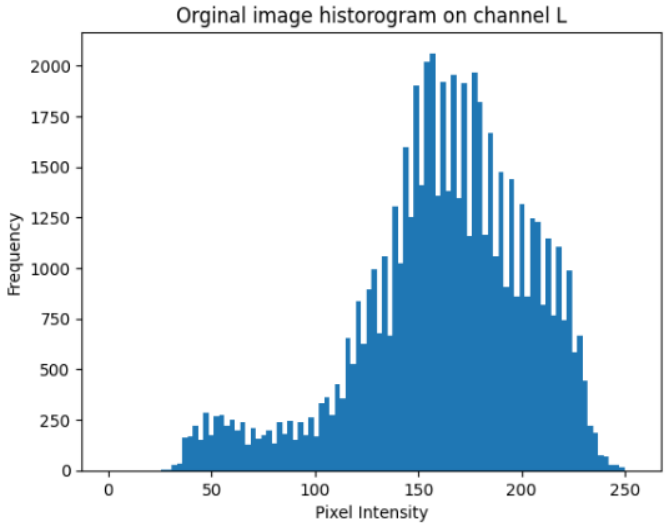


Figure 7: Histogram before Equalization

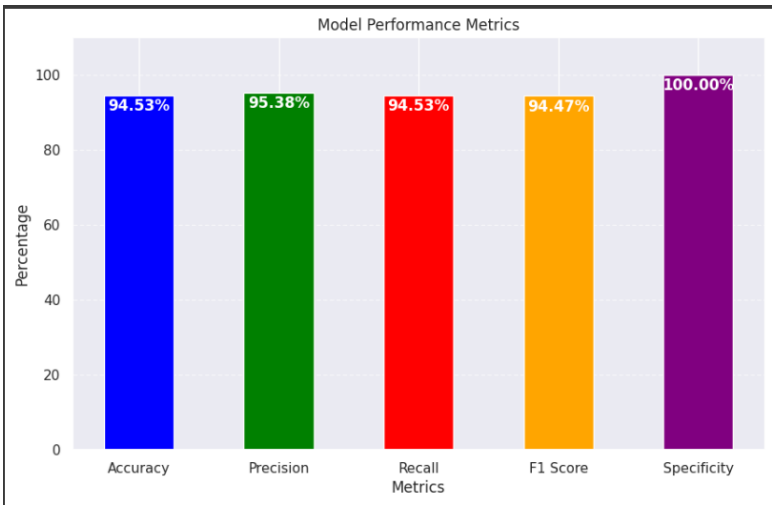


Figure 8: Model Graphical Representation

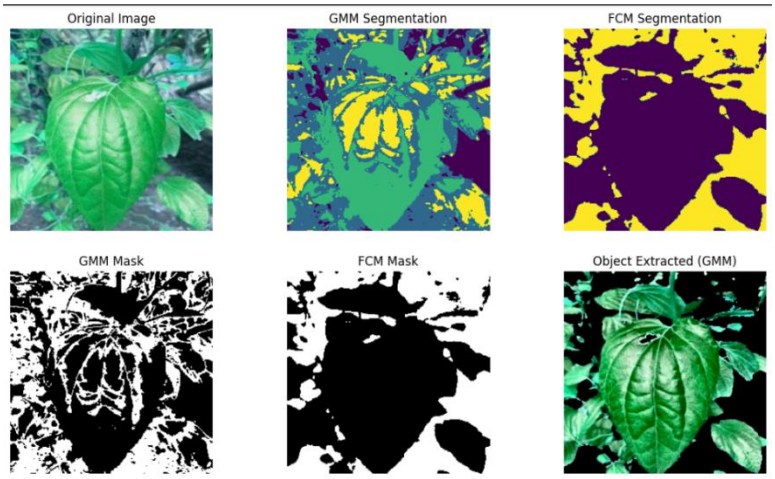


Figure 9: Segmentation

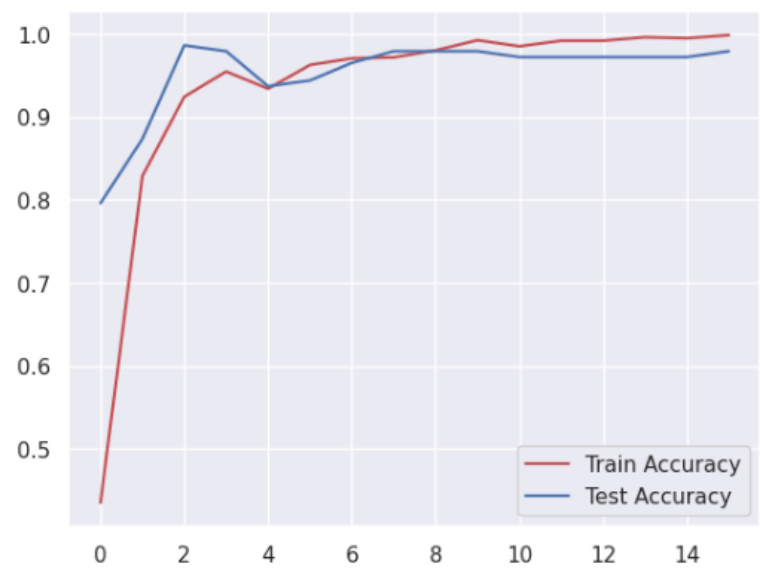


Figure 10: Training Accuracy vs. Testing Accuracy

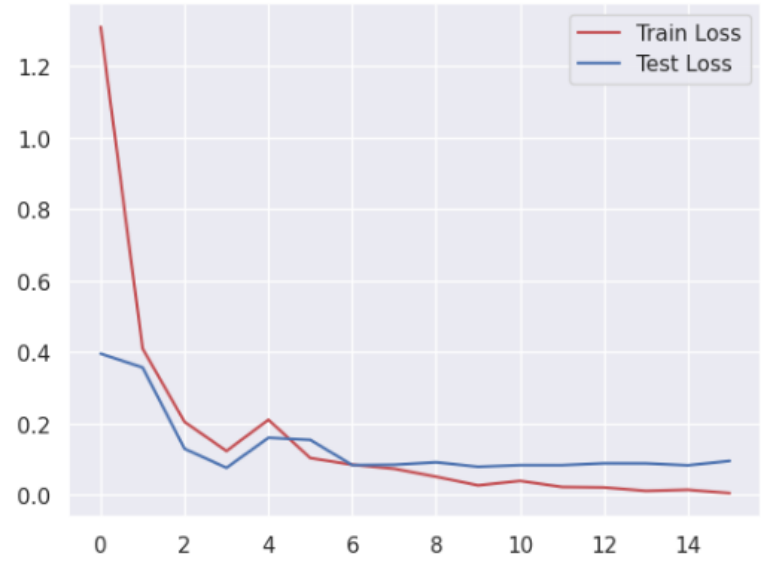


Figure 11: Training Loss vs. Testing Loss

5. Conclusion

Betel leaf, well-known for its therapeutic and cultural value, is seriously threatened by a number of illnesses, endangering both its commercial and agricultural sustainability. Accurate diagnosis is hampered by difficulties in detecting illnesses in betel leaves. This study employs a thorough methodology to resolve these issues. First, pre-processing with CLAHE is applied after standardizing raw images to a pixel size. Following this, segmentation methods such as FCM and GMM divide the images into meaningful segments according to intensity, color, and texture criteria. This makes it easier to extract pertinent data and identify regions of interest. Data augmentation techniques such as horizontal and vertical flipping are applied to the segmented images to provide equal representation across classes and improve model resilience. GLCM approach is used to extract features from augmented images. Following feature extraction, a DCNN tuned by RPO method is used to input the features into a classification phase. By using this technology, the study hopes to increase the accuracy of disease identification in betel leaves, which will help to maintain their cultivation and economic importance. The usefulness of the suggested method in precisely categorizing betel leaf illnesses is demonstrated by experimental data.

References

[1] Chowdhury, M.E., Rahman, T., Khandakar, A., Ayari, M.A., Khan, A.U., Khan, M.S., Al-Emadi, N., Reaz, M.B.I., Islam, M.T. and Ali, S.H.M., 2021. Automatic and reliable leaf disease detection using deep learning techniques. AgriEngineering, 3(2), pp.294-312.

[2] Harakannanavar, S.S., Rudagi, J.M., Puranikmath, V.I., Siddiqua, A. and Pramodhini, R., 2022. Plant leaf disease detection using computer vision and machine learning algorithms. Global Transitions Proceedings, 3(1), pp.305-310.

[3] Ashwinkumar, S., Rajagopal, S., Manimaran, V. and Jegajothi, B., 2022. Automated plant leaf disease detection and classification using optimal MobileNet based convolutional neural networks. Materials Today: Proceedings, 51, pp.480-487.

[4] Das, A., Dey, A.K. and Sharma, M., 2017. Leaf disease detection, quantification and classification using digital image processing. Int. J. Innov. Res. Electr. Electron. Instrum. Control Eng, 5(11).

[5] Madhumita, M., Guha, P. and Nag, A., 2019. Extraction of betel leaves (Piper betle L.) essential oil and its bio-actives identification: Process optimization, GC-MS analysis and anti-microbial activity. Industrial Crops and Products, 138, p.111578.

[6] Ahmed, S., Zaman, S., Ahmed, R., Uddin, M.N., Acedo Jr, A. and Bari, M.L., 2017. Effectiveness of non-chlorine sanitizers in improving the safety and quality of fresh betel leaf. LWT, 78, pp.77-81.

[7] Vallabhajosyula, S., Sistla, V. and Kolli, V.K.K., 2022. Transfer learning-based deep ensemble neural network for plant leaf disease detection. Journal of Plant Diseases and Protection, 129(3), pp.545-558.

[8] Harakannanavar, S.S., Rudagi, J.M., Puranikmath, V.I., Siddiqua, A. and Pramodhini, R., 2022. Plant leaf disease detection using computer vision and machine learning algorithms. Global Transitions Proceedings, 3(1), pp.305-310.

[9] Ashwinkumar, S., Rajagopal, S., Manimaran, V. and Jegajothi, B., 2022. Automated plant leaf disease detection and classification using optimal MobileNet based convolutional neural networks. Materials Today: Proceedings, 51, pp.480-487.

[10] Hameed Al-bayati, J.S. and Üstündağ, B.B., 2020. Evolutionary feature optimization for plant leaf disease detection by deep neural networks. International Journal of Computational Intelligence Systems, 13(1), pp.12-23.

[11] Vallabhajosyula, S., Sistla, V. and Kolli, V.K.K., 2022. Transfer learning-based deep ensemble neural network for plant leaf disease detection. Journal of Plant Diseases and Protection, 129(3), pp.545-558.

[12] Chowdhury, M.E., Rahman, T., Khandakar, A., Ayari, M.A., Khan, A.U., Khan, M.S., Al-Emadi, N., Reaz, M.B.I., Islam, M.T. and Ali, S.H.M., 2021. Automatic and reliable leaf disease detection using deep learning techniques. AgriEngineering, 3(2), pp.294-312.

[13] Atchudan, R., Edison, T.N.J.I., Perumal, S., Vinodh, R. and Lee, Y.R., 2019. Betel-derived nitrogen-doped multicolor carbon dots for environmental and biological applications. Journal of Molecular Liquids, 296, p.111817.

[14] Zhao, Y., Chen, Z., Gao, X., Song, W., Xiong, Q., Hu, J. and Zhang, Z., 2021. Plant disease detection using generated leaves based on DoubleGAN. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 19(3), pp.1817-1826.

[15] Pandian, J.A., Kanchanadevi, K., Kumar, V.D., Jasińska, E., Goňo, R., Leonowicz, Z. and Jasiński, M., 2022. A five convolutional layer deep convolutional neural network for plant leaf disease detection. Electronics, 11(8), p.1266.

[16] Kusuma, S. and Jothi, K.R., 2024. Early betel leaf disease detection using vision transformer and deep learning algorithms. International Journal of Information Technology, 16(1), pp.169-180.

[17] Dey, A.K., Sharma, M. and Meshram, M.R., 2022. Development of ANN and ANFIS classifier for betel leaf pathogen detection. Journal of The Institution of Engineers (India): Series B, 103(5), pp.1555-1562.

[18] Hridoy, R.H., Tarek Habib, M., Sadekur Rahman, M. and Uddin, M.S., 2022. Deep Neural Networks-Based Recognition of Betel Plant Diseases by Leaf Image Classification. In Evolutionary Computing and Mobile Sustainable Networks: Proceedings of ICECMSN 2021 (pp. 227-241). Singapore: Springer Singapore.

[19] Shovon, M.S.H., Mozumder, S.J., Pal, O.K., Mridha, M.F., Asai, N. and Shin, J., 2023. PlantDet: A Robust Multi-Model Ensemble Method Based on Deep Learning for Plant Disease Detection. IEEE Access.

[20] Meenakshi, T., 2023. Automatic detection of diseases in leaves of medicinal plants using modified logistic regression algorithm. Wireless Personal Communications, 131(4), pp.2573-2597.

[21] Prodhan, M.D.H., Afroze, M., Begum, A., Ahmed, M.S. and Sarker, D., 2023. Optimization of a QuEChERS based analytical method for the determination of organophosphorus and synthetic pyrethroid pesticide residues in betel Leaf. International Journal of Environmental Analytical Chemistry, 103(6), pp.1292-1303.

[22] Hridoy, R.H., Arni, A.D. and Haque, A., 2023. Improved vision-based diagnosis of multi-plant disease using an ensemble of deep learning methods. International Journal of Electrical and Computer Engineering (IJECE), 13(5), pp.5109-5117.

[23] Dataset taken from: “https://www.kaggle.com/datasets/devias/beetle-data”, dated 25-1-2023.