VOL. 11. NO. 3 FEBRUARY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480 /jitk.v11i3.7031

JITK (JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI KOMPUTER)

OPTIMIZATION OF MLP-NN FOR MANGO LEAF DISEASE PREDICTION USING IMAGE-BASED FEATURE EXTRACTION

Budi Triandi[1*]; Lili Tanti[2]

Informatics[1]
Information Systems[2]
Universitas Potensi Utama, Medan, Indonesia[1,2]
https://potensi-utama.ac.id/ [1,2]
buditriandi@gmail.com* lilitanti82@gmail.com

(*) Corresponding Author
(Responsible for the Quality of Paper Content)

Abstract*—Mango (*Mangifera indica Linn.*) is a nutrient-rich fruit, yet leaf diseases caused by microorganisms can significantly reduce crop productivity. Early detection is essential to prevent further damage and support effective disease management. This study proposes an optimized mango leaf disease prediction model using a multi-layer perceptron neural network (MLP-NN). Image-based feature extraction is performed using the Inception v3 architecture to obtain high-level color and texture features that improve classification performance. Unlike previous studies that rely solely on manually engineered features or full CNN training, this research introduces a hybrid approach that integrates deep feature extraction with MLP-NN optimization, offering a lightweight yet highly accurate alternative. Several hyperparameter combinations, including activation functions (ReLU, tanh, sigmoid) and optimization algorithms (Adam and SGD), were evaluated using the Orange platform. The optimized MLP-NN model with ReLU and Adam achieved the highest accuracy of 93.5%, demonstrating better stability and training efficiency compared to other configurations. These findings highlight the novelty and advantages of the proposed method, showing improved accuracy with lower computational cost relative to many existing approaches. This study provides an efficient solution for mango leaf disease prediction and supports future development of automated plant disease detection systems.*

Keywords*: Adam, Inception v3, Mango leaf disease, Neural network, Optimization.*

Intisari—*Mangga (*Mangifera indica Linn.*) merupakan buah yang kaya nutrisi, namun penyakit pada daun yang disebabkan oleh mikroorganisme dapat menurunkan produktivitas secara signifikan. Deteksi dini sangat penting untuk mencegah kerusakan lebih lanjut dan mendukung pengelolaan penyakit yang efektif. Penelitian ini mengusulkan model prediksi penyakit daun mangga yang dioptimalkan menggunakan algoritma multi-layer perceptron neural network (MLP-NN). Ekstraksi fitur berbasis citra dilakukan menggunakan arsitektur Inception v3 untuk memperoleh fitur warna dan tekstur tingkat tinggi yang meningkatkan kinerja klasifikasi. Berbeda dari penelitian sebelumnya yang hanya mengandalkan fitur buatan secara manual atau pelatihan CNN secara penuh, penelitian ini menawarkan pendekatan hibrida yang mengintegrasikan ekstraksi fitur deep learning dengan optimasi MLP-NN, sehingga memberikan alternatif yang lebih ringan namun tetap akurat. Berbagai kombinasi hiperparameter, termasuk fungsi aktivasi (ReLU, tanh, sigmoid) dan algoritma optimasi (Adam dan SGD), diuji menggunakan platform Orange. Model MLP-NN teroptimasi dengan ReLU dan Adam mencapai akurasi tertinggi sebesar 93,5%, menunjukkan stabilitas dan efisiensi pelatihan yang lebih baik dibandingkan konfigurasi lainnya. Temuan ini menegaskan kebaruan dan keunggulan metode yang diusulkan, dengan akurasi lebih tinggi dan biaya komputasi lebih rendah dibanding banyak pendekatan terdahulu. Penelitian ini memberikan solusi yang efisien untuk prediksi penyakit daun mangga serta mendukung pengembangan sistem deteksi penyakit tanaman otomatis di masa depan.*

Kata Kunci*: Adam, Inception v3, Penyakit daun mangga, Jaringan saraf tiruan, Optimasi.*

VOL. 11. NO. 3 FEBRUARI 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i3.7031

INTRODUCTION

The agricultural sector in Indonesia supports the livelihoods of approximately 30–40% of the population, with farming serving as the primary source of income for many. However, climate change poses significant challenges, affecting crop yields and contributing to the increasing prevalence of plant diseases. These diseases are primarily caused by microorganisms such as bacteria, fungi, and viruses, which have a substantial impact on crop production. Plant disease identification has conventionally been carried out using manual approaches, which require significant time, are susceptible to human mistakes, and frequently result in inconsistent or inaccurate diagnoses [1]. At present, the lack of professional and reliable plant disease diagnostic tools results in inefficiencies, including the improper application of insecticides [2].

Mango (Mangifera indica), a popular fruit in Indonesia, faces significant threats from microorganisms such as fungi, bacteria, and parasites that reduce productivity. As the second-largest mango producer in the world, Indonesia's mango farming sector would benefit from early disease detection to stabilize and enhance production. According to the Food and Agriculture Organization (FAO), early-stage detection is essential for sustaining consistent levels of mango production.

Recent developments in machine learning (ML) have led to disease classification models that can automatically derive important features from segmented image data. Nevertheless, many conventional ML approaches still rely on manually engineered features and small, highly curated datasets, limiting automation and reducing generalization ability [3] [4]. Deep learning approaches, especially Convolutional Neural Networks (CNNs), extracts image features automatically in a hierarchical manner and demonstrates high performance on well-controlled datasets. However, CNN-based systems are often constrained by the need for large, diverse, and well-labeled datasets and remain susceptible to overfitting and reduced robustness under variable real-world conditions [5]. These practical limitations have driven research toward hybrid architectures and transfer strategies to improve adaptability and resilience in real operational environments.

Other studies have combined multilayer Artificial Neural Networks (ANNs) with feature-selection techniques such as genetic algorithms (GAs) to reduce dimensionality and improve accuracy. Nevertheless, these processing pipelines tend to raise computational demands and are frequently unsuitable for real-time applications or devices with limited resources [6]. In response to these challenges, deep learning (DL) techniques particularly CNNs have grown increasingly popular, as DL can overcome the limitations of traditional ML by leveraging large datasets to improve accuracy and minimize overfitting [7]. CNNs, widely used in agriculture and bioinformatics, have been shown to be highly effective in image-recognition tasks due to their ability to extract features automatically without manual input [8]. Nevertheless, CNN performance remains heavily dependent on large, diverse, and well-annotated datasets [9].

On the other hand, several studies have explored plant disease detection by integrating multilayer ANNs with GA-based feature selection [10]. These approaches are designed to enhance both the precision and effectiveness of plant disease detection through the application of machine learning techniques. The combination of ANN and GA is considered innovative because GA can optimize feature selection, reduce data redundancy, and enhance classification accuracy. However, these studies also indicate that GA-driven feature elimination increases computational overhead, making the ANN–GA integration less suitable for real-time detection.

In response to these challenges, this study proposes a hybrid solution that employs Inception v3 for automated image-based feature extraction and an optimized MLP-NN for classification. This strategy aims to (1) leverage deep features extracted by a well-established CNN architecture, while (2) reducing computational load by delegating the classification stage to a more lightweight MLP, thereby improving feasibility for deployment in resource-limited environments. Initial experimental results indicate that the proposed hybrid setup delivers comparable accuracy while requiring reduced computational resources relative to end-to-end CNN training on small-scale datasets, effectively addressing key research gaps related to accuracy, robustness, and deployability under real-world conditions.

Prior to integrating optimization techniques, the dataset was evaluated using various hyperparameter combinations, such as activation functions ReLU, tanh, and sigmoid as well as optimization methods like Adam and SGD, to determine the most effective hybrid configuration [11]. In contrast to earlier studies that relied on limited datasets or manual feature extraction, this work proposes an automated image-based feature-extraction approach to reduce human dependency

VOL. 11. NO. 3 FEBRUARY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480 /jitk.v11i3.7031

and enhance detection efficiency. Early results confirm the efficacy of this approach in identifying mango leaf diseases by means of automated image recognition and feature extraction [12].

MATERIALS AND METHODS

THEORETICAL BACKGROUND

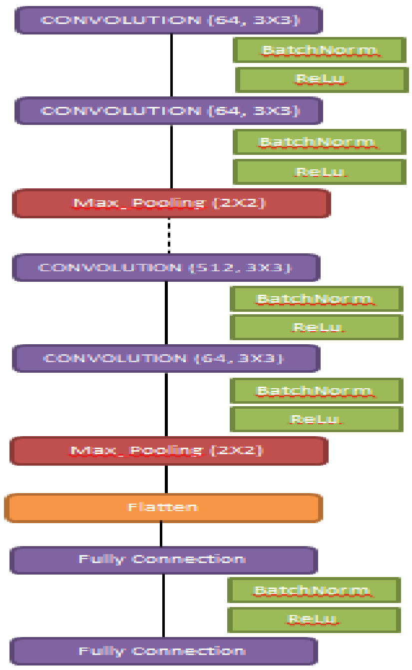
Artificial Neural Network Model

The potential gap between human and machine capabilities has been filled through the development of Artificial Intelligence [AI] technology. Artificial Neural Networks is a computational model that imitates the way the human brain works to learn from data through mathematical models that describe the function of the brain, neurons, synapses, and their interconnections. With a structure consisting of neurons, layers, and connections, and the use of activation functions and training processes, ANN can solve various machine learning problems and provide predictions or classifications based on complex data. The notion of machine learning that mimics human cognition serves as a fundamental motivation in this domain, supporting tasks such as image analysis, classification, and related applications.

Humans acquire knowledge through perceptual processes, whereas machines are trained using supervised learning, enabling them to function as powerful tools comparable to the human brain [13][14]. Several studies have been conducted, including research by Mona Jamjoom *et al.* [15] which proposed a segmentation method using K-means clustering, followed by texture feature extraction (GLCM & LBP), and classification using SVM, achieving 97% accuracy in plant leaf disease classification. This study explains the effectiveness of combining segmentation processes and feature extraction for machine learning–based classification. Demilie W. B. [16] explained the evaluation of several classifiers (including ANN and SVM) on features automatically extracted from plant leaves using image segmentation and feature engineering, demonstrating that ANN can outperform other classical methods.

Convolution Neural Network

Convolutional Neural Network (CNN) is widely used in various studies for the purpose of identifying, classifying, and recognizing plant diseases that involve large-scale image processing, achieving high validation accuracy, and enabling the discovery of specific patterns within recognition frameworks [17]. Mathematical operations in CNN called convolution, in extracting image features the concept of convolution uses a kernel that starts from the top left to the bottom right [18]. The convolutional layer + ReLU is the first of CNN's four layers, followed by the max-pooling layer, the fully connected layer, and the output layer. The chosen characteristics are dispersed throughout the image in the convolutional layer. ReLU, the activation for this layer, substitutes zero for every negative value such that the total of all values is not zero. The image size is then decreased by the pooling layer. The completely connected layer transforms the image into a single layer or vector using Softmax activation [19]. CNN is a very flexible model, depending on the requirements of the task to be met.



Source: (Research Results, 2025)
Figure 1 CNN Architecture Approach

Figure 1 illustrates several CNN architectures widely adopted by researchers worldwide, including VGG, GoogLeNet, AlexNet, ResNet, and Inception V3. Although these models share similar structural characteristics, they vary in parameter settings such as the number of units, learning rate, and dropout rate.

For this study, Inception v3 was selected over other CNN architectures like VGG and ResNet due to its superior performance in handling large and complex datasets, which is crucial for image classification tasks in plant disease detection [20]. Inception v3 utilizes a more efficient design by employing 'factorized convolutions,' which allow for a deeper network without a significant increase in computational cost. This design not only enhances the model's ability to learn fine-grained

VOL. 11. NO. 3 FEBRUARI 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i3.7031

patterns but also optimizes the use of computational resources. Compared to VGG, which tends to be computationally expensive due to its deep architecture with a large number of parameters, Inception v3 achieves high accuracy while reducing overfitting by using a smaller number of parameters [21]. Additionally, Inception v3 has shown better generalization ability over ResNet for tasks involving fine-grained image features, such as distinguishing between different types of plant diseases. Its architecture, which combines multiple convolutional filters at different levels, makes it particularly suited for extracting diverse features from complex images [22].

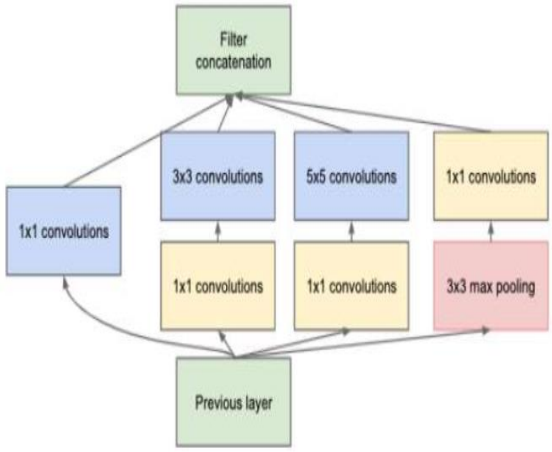
Inception V3

In this study, we employ the Inception v3 architecture as the feature extraction backbone for the image recognition and classification process. Inception v3 integrates multiple convolutional kernels of different sizes within the same module, enabling the model to capture both fine-grained and high-level features efficiently. To reduce overfitting, the architecture incorporates dimensionality reduction through 1×11 \times 11×1 convolutions, grid-size reduction, and regularization strategies such as factorized convolutions and auxiliary classifiers, which collectively minimize computational cost while maintaining high representational power [23].

The selection of Inception v3 over other widely used CNN architectures such as VGG or ResNet is based on several technical considerations. Compared to VGG, which requires a significantly larger number of parameters and higher computational cost, Inception v3 achieves comparable or better accuracy with far greater efficiency due to its multi-branch architecture and factorized convolutions.

Meanwhile, although ResNet offers strong performance through residual connections, it typically requires deeper networks and larger training datasets to reach optimal performance. Inception v3 provides a balanced trade-off between depth, computational efficiency, and feature richness, making it more suitable for medium-sized datasets such as mango leaf images used in this study. Additionally, the availability of a robust pre-trained model enhances its ability to generalize well when used for transfer learning in agricultural image analysis tasks [24].

Figure 2 presents the detailed architecture of the Inception v3 model.



Source: (Research Results, 2025)
Figure 2 Inception V3 Model

The introduced model has sequential modeling with a series of layers that process the input image. Starting from the first and second layers are convolution layers with 64 filters followed by Batch normalization and ReLU activation. The max pooling layer is the third layer with a pool size of 1x1, which reduces the size of the given image. Then followed by the max pooling layer, which is with a pool size of 1x1. The next layer is the convolution layer with 512 filters, which is followed by the batch normalization function and ReLU. This is followed by the max pooling layer with a pool size of 3x3 and 5x5, which is the sixth layer with a pool size of 1x1. The output layer is a fully connected layer that uses four output neurons to generate accurate class labels for the expected class.

Optimizer Algorithm

The application of various machine learning optimization techniques, such as Adam (Adaptive Moment Estimation) and SGD (Stochastic Gradient Descent), depends on the nature of the issue, the amount of data, and the complexity of the model. Although they have different features, both optimizers are often employed in machine learning and deep learning model training.

Adam Optimizer

In the learning process, weight is the main factor in determining the good and bad results of machine learning, adaptive learning rate is needed because the data set is changing. Adam optimizer is one of the optimization algorithms that has the ability to update weight values and adaptive learning rate [25].

Adam optimizer can provide solutions to complex problems in sparse gradients and can provide good performance by utilizing the first and second moments of the gradient so that it can adapt to the learning rate [26]. The first and second moments in this case are symbolized by the variables m and v by calculating the moment estimation gradient, the biased moment estimation will be corrected at each moment (t) [27], with the following equation:

Adam uses two momenta: m_t : The first average of the gradients, which helps speed up convergence to consistent gradients. v_t : The second average of the squares of the gradients, which helps adjust the update step to the gradient variance, reducing large updates in directions with high variance.

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \quad (1)$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \quad (2)$$

From equations (1) and (2) g_t is the gradient at iteration t and β_1 and β_2 are parameters that control the level of influence of the first and second moments usually (β_1=0.9 and β_2=0.999). Calculating the corrected bias (since the initial estimates of m_t and v_t may be very biased towards 0) then the bias can be corrected with the following equation:

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t} \quad (3)$$

$$\hat{v}_t = \frac{v_t}{1-\beta_2^t} \quad (4)$$

Parameter update:

$$\theta_{t+1} = \theta_t - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \quad (5)$$

From equation (5) it is clear that θ_t is the parameter being optimized, η is the learning rate, ϵ is a small value (e.g. 10^{-8}) to prevent division by zero. Adam is an efficient optimizer algorithm in handling various kinds of problems because it can handle gradients with rare values or large variances and can work well on varied data or parameters that have different scales [18].

SGD (Stochastic Gradient Descent) Optimizer

Within machine learning, Stochastic Gradient Descent (SGD) is commonly employed as an optimization technique to train models by iteratively adjusting parameters, including weights and biases, in order to reduce the value of the loss function [28].

How SGD works is 1) Take one random data sample (or mini-batch) from the dataset. 2) Calculate model predictions and loss values based on the data. 3) Calculate the gradient (derivative) of the loss against the model parameters. 4) Update the parameters using the formula:

$$\theta := \theta - \eta \cdot \nabla_\theta J(\theta; x^{(i)}, y^{(i)}) \quad (6)$$

$$\theta := \theta - \eta \cdot \frac{1}{m} \sum_{j=1}^{m} \nabla_\theta J(\theta; x^{(j)}, y^{(j)}) \quad (7)$$

5) Repeat this process for all data for several epochs (full iterations over the dataset).

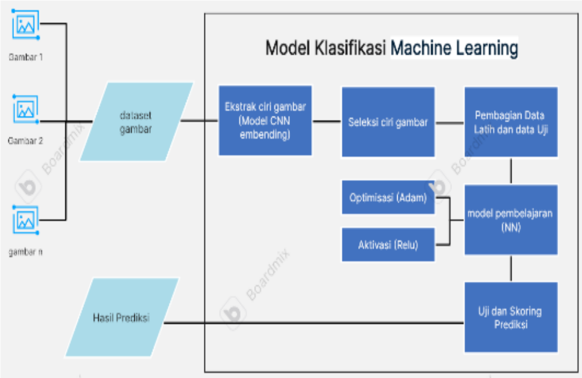
SGD is a very suitable optimization algorithm for fundamentals and is widely used in machine learning model training, especially for neural network, regression/classification and logistic based models [29].

METHODS

The method used to detect plant diseases is by using the characteristics of mango leaves. Data classification is carried out with a collection of various changes in mango leaves infected with the disease, different leaf characteristics will be the coefficient values learned by the machine learning algorithm, with efforts to find the right learning parameters, it will produce an optimal prediction model and then the accuracy of the diagnosis of disease in mango leaves will be obtained, so in this study the framework is divided into several stages:

a) Step 1 We use a mango leaf image dataset imported from the "MangoLeafBD Dataset" archive
b) Step 2 The mango leaf image dataset is extracted with the "inception v3" extra feature model which adopts the CNN convolutional neural network architecture to embed images into coefficient value data
c) Step 3 we select features that are considered not to have a major contribution to learning data patterns such as image dimensions and image names
d) Step 4 then, determining the ratio of training data and test data and here we use k-flods with a value of 10
e) Step 5 performs a comparison of hyperparameter tuning by comparing combinations of algorithms for activation and optimization parameters, to produce the best parameters for the problem.
f) Step 6 finally our work by inputting the data set into the NN model to predict mango disease through leaves.

VOL. 11. NO. 3 FEBRUARI 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i3.7031

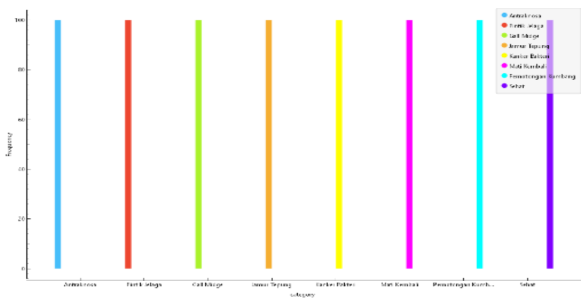


Source: (Research Results, 2025)
Figure 3 Block Diagram of Mango Plant Leaf Disease Classification Methodology

This block diagram shows the steps in optimizing the prediction model and classifying mango plant diseases through leaves with a collection of image data that is processed before being entered into the learning model shown in Figure 3.

Data Distribution

The data distribution used in this study consists of 800 mango leaf images categorized into eight disease classes, namely: Anthracnose, Sooty Spot, Gall Midge, Powdery Mildew, Bacterial Canker, Dieback, Beetle Damage, and Healthy. To provide a clearer understanding of the dataset composition, the number of images for each class is as follows: Anthracnose (100 images), Sooty Spot (100 images), Gall Midge (100 images), Powdery Mildew (100 images), Bacterial Canker (100 images), Dieback (100 images), Beetle Damage (100 images), and Healthy (100 images).



Source: (Research Results, 2025)
Figure 4 Frequency Distribution

As illustrated in Figure 4, the frequency distribution chart confirms that the dataset is perfectly balanced, with each category receiving an equal number of samples. This balanced distribution is advantageous during model training, as it ensures that the model learns distinguishing features fairly from all classes, prevents bias toward majority categories, and improves the model's ability to generalize and make accurate predictions across all disease types.

Image Feature Extraction and Data Per-Processing

The mango leaf dataset is obtained from the MangoLeafBD Dataset Kaggle dataset, this image consists of various colors of Red, Green, and Blue (RGB) with variations in resolution and format, here we extract image features from 800 image files by producing 2048 image features, to produce coefficient values from image features we propose an extra feature model "inception v3" which adopts a CNN convolutional neural network architecture for embedding images into coefficient value data, inception v3 is designed for image recognition tasks in the form of deep neural network architecture built using a base layer with convolution operations.

$$S(i,j) = (I * K)_{i,j} = \sum_{m=0}^{F-1} \sum_{n=0}^{F-1} I_{i+m,j+n} \cdot K_{m,n} \quad (8)$$

From equation 8 which is the basis of convolution where S is the convolution result, $(I * K)_{i,j}$ is the pixel value at position (i,j) of the resulting feature map, $I_{i+m,j+n}$ is the input image pixel value at position $(i+m, j+n)$, $K_{m,n}$ is the value of the kernel at position (m,n). Convolution involves calculating the pixel value of the feature map by summing the results of the multiplication between the kernel and the input image patch. To reduce the training time, we do feature selection by selecting features that do not provide a strong contribution to the classification.

Source: (Research Results, 2025)
Figure 5 Example Image Dataset

RESULTS AND DISCUSSION

Model Training and Testing

As seen in Figure 5, this study used a dataset of 800 photos of mango leaves from nine classes (eight diseases and one healthy), including Bacterial

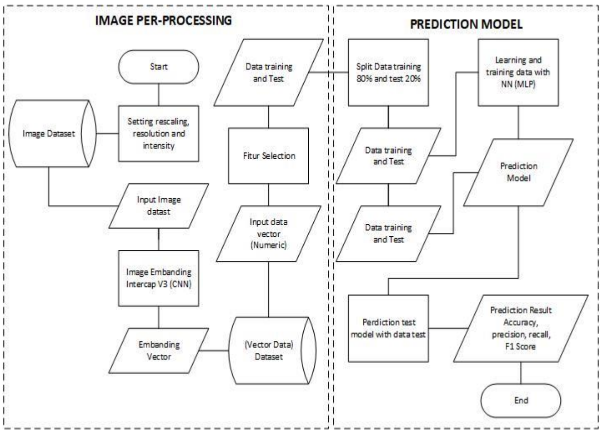
VOL. 11. NO. 3 FEBRUARY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480 /jitk.v11i3.7031

Canker, Cutting Weevil, Die Back, Sooty Mold, Anthracnose, Gall Midge, Powdery Mildew, Beetle Damage, and Healthy. One of Indonesia's main mango-producing regions, North Sumatra, is where the photos were gathered. Infected leaves were photographed without background from various angles under uniform lighting conditions to capture feature variations while maintaining image clarity.

To guarantee uniformity in image processing and strike the best possible balance between feature retention and computational efficiency, all images were standardized to a resolution of 240 × 320 pixels. This resolution was selected because medium-scale image sizes (ranging from 224 × 224 to 256 × 256 pixels) are widely adopted in plant disease classification studies, as they provide sufficient visual detail for identifying lesions, textures, and color variations while reducing computational overhead. Recent studies such [30] demonstrate that resolutions within this range yield high classification performance in CNN-based agricultural disease detection tasks while minimizing memory and training time requirements.

Thus, the use of 240 × 320 pixels aligns with recommended practices for deep learning–based plant disease recognition capturing critical features on mango leaves without introducing unnecessary computational cost. The dataset was then organized into the MangoLeafBD Dataset [31].

The process steps are shown in the figure. The initial step starts with rescaling the mango images to a lower resolution and equalizing the dimensions between images. The image contrast is increased to uniform the pixel intensity. Next, vector embedding is performed using the CNN Inception V3 architecture to convert the images into numerical representations, resulting in 2048 features per image.

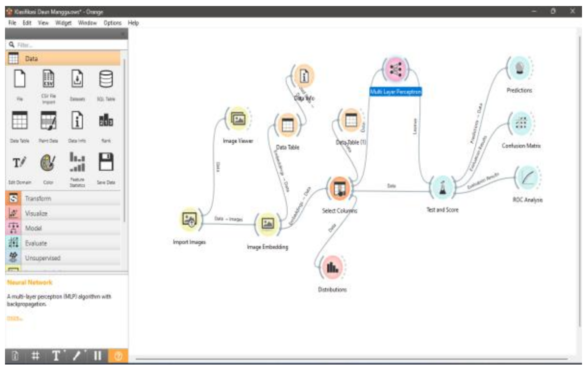


Source: (Research Results, 2025)
Figure 6 Block Diagram of Training and Testing Model

Proposed MLP-NN Algorithm

In this study, the mango leaf disease classification model was developed using a Multi-Layer Perceptron Neural Network (MLP-NN) on the Orange platform, which enables visual data processing, training, and evaluation without manual coding. Feature extraction was performed using the Inception V3 CNN architecture, producing 2048-dimensional feature vectors that serve as the input to the MLP classifier. The MLP model consists of three hidden layers with 100, 50, and 30 neurons. The selection of these neuron counts is based on both empirical evaluation and established neural network design principles. The first hidden layer with 100 neurons provides sufficient representational capacity to process the high-dimensional input features generated by Inception V3. The second and third layers, with 50 and 30 neurons respectively, progressively reduce the dimensionality and help the network learn more abstract representations while mitigating the risk of overfitting.

This "funnel-shaped" architecture is commonly used to encourage hierarchical feature compression and improve generalization. The configuration was not chosen arbitrarily; it was validated through systematic hyperparameter exploration in which multiple architectures (e.g., 256-128-64, 128-64-32, 80-40-20) were tested. The 100-50-30 structure consistently produced the best balance between accuracy, training stability, and computational efficiency during cross-validation experiments. Eight neurons with Softmax activity, which correspond to the eight illness types, make up the output layer. Accuracy, precision, recall, F1-score, and the confusion matrix were used to assess the model's performance after it was trained using the categorical cross-entropy loss function and the Adam optimizer. Overall, the selected architecture ensures an optimal trade-off between model capacity and computational cost.



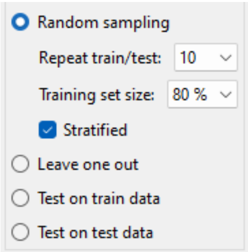
Source: (Research Results, 2025)
Figure 7 Orange Tools for Building Machine Learning Models

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 11. NO. 3 FEBRUARI 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i3.7031

Split of Training And Test Data

Using a random sample strategy, the dataset was split into 80% training data and 20% test data for the purpose of training and testing the MLP-NN model on the Orange application. The Repeat Train/Test approach, which automatically splits and retrains the model, was used ten times to increase the evaluation's dependability. While identifying possible overfitting or data integration, evaluation scores like accuracy, precision, recall, and F1-score are averaged to give a consistent view of model performance.



Source: (Research Results, 2025)
Figure 8 Split of Training Data and Test Data

Measuring Machine Learning Model Performance

Here, we evaluate how effectively a machine learning model predicts new data by using performance indicators. Metrics including accuracy, precision, recall, and F1-score are used in this assessment.

Classification Accuracy

The total number of successfully identified samples divided by the total number of samples used in the assessment yields a ratio that indicates the model's degree of effectiveness in categorizing data.

$$Accuracy\ (CA) = \frac{TP+TN}{TP+TN+FP+FN} \tag{9}$$

Precision

Quantitatively, it represents the proportion of sample units that are actually classified into the positive class compared to the total number of sample units, and is often used to describe the characteristics of class distribution in evaluating the performance of a classification model.

$$Precision\ (\mathrm{P}) = \frac{TP}{TP+FP} \tag{10}$$

Recall

Formally, this metric represents the proportion of samples that actually belong to the positive class out of all samples predicted as positive by the system, thus reflecting the level of accuracy of the model in providing positive predictions.

$$Recall\ (R) = \frac{TP}{TP+FN} \tag{11}$$

F1-score

F1 Score is a composite metric obtained from the harmonic mean of precision and recall, two key indicators in evaluating classification models. This metric is particularly useful in situations where the class distribution is uneven, as it provides a fairer representation of model performance than a simple accuracy metric.

$$F1\ Score = 2\ \mathrm{x}\ \frac{\text{Presisi x Recall}}{\text{Presisi+ Recall}} \tag{12}$$

Model Optimization

In order to find the best configuration for the MLP-based mango leaf disease classification model, this work uses a comparative hyperparameter tuning strategy. The learning rate, batch size, activation function, and optimizer method are the main hyperparameters taken into account. These factors all have a big impact on the model's capacity for generalization, training stability, and convergence speed.

The step size during gradient descent is controlled by the learning rate, which is ultimately set at 0.01. While a higher learning rate runs the danger of overshooting the ideal answer, a lower learning rate can offer consistent but delayed convergence. The batch size, set to 32, determines the number of training samples used to update the model weights in one iteration. Smaller batch sizes typically lead to more frequent updates that improve generalization, while larger batch sizes produce smoother gradient estimates but may reduce the model's ability to generalize. The chosen values were determined empirically through systematic evaluation, which tested multiple learning rates and batch sizes to identify the combination that achieved optimal performance for this dataset.

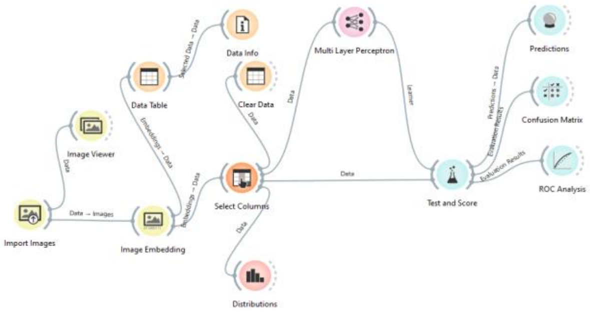
Additionally, we compared three activation functions (ReLU, Tanh, and Logistic) and two optimizer algorithms (SGD and Adam). This systematic hyperparameter comparison enabled the identification of the configuration that best balances convergence speed, accuracy, and generalization. The results are analyzed by considering the combined effects of activation function, learning rate, batch size, and optimizer

VOL. 11. NO. 3 FEBRUARY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480 /jitk.v11i3.7031

choice, ensuring that the final model configuration achieves stable and robust performance across all eight disease classes.

Experimental Results

This test was conducted using the Orange application by implementing the Multi-Layer Perceptron (MLP-NN) type Neural Networks algorithm. The testing process begins by preparing a dataset that has been cleaned and processed previously, then entered into the Orange workflow. An input layer, one or more hidden layers, and an output layer are among the layers of neurons that make up the MLP-NN method, which is used as a machine learning model. To optimize model performance, network parameters including the number of neurons and activation functions are changed in this test. To assess accuracy, precision, recall, and F1-score, the model is trained using training data and evaluated using test data. The test results show the extent to which the MLP-NN algorithm is able to effectively classify the dataset used.



Source: (Research Results, 2025)

Figure 9 Neural Networks with Multi-Layer Perceptron (MLP-NN) for Mango Leaf Disease Classification

Figure 9 shows a prototype of a mango leaf disease classification model using the Neural Network Multi-Layer Perceptron (MLP-NN) algorithm. This model imitates the way the human brain works in recognizing patterns from image features such as color, texture, and symptom shape. Features are extracted using the CNN Inception V3 architecture and fed into the input layer in numeric form. Data is processed through three hidden layers (100, 50, 30 neurons) using activation functions (ReLU, Tanh, Logistic). Before processing, a selection of 2048 feature data and one category label is carried out.

Model Optimization Results with Hyperparameter Tuning

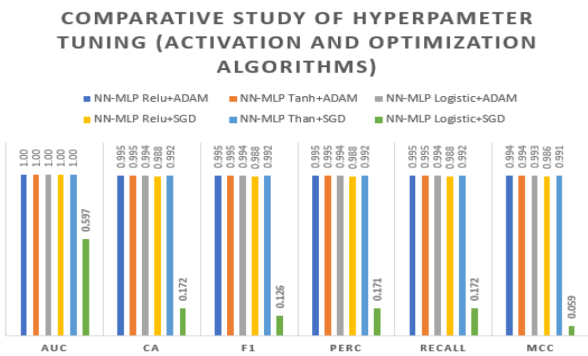
Hyperparameter tuning on the MLP-NN model for mango leaf disease prediction was carried out to improve accuracy and generalization. This process involved testing various combinations of activation functions (tanh, ReLU, and sigmoid) and optimization algorithms (Adam and SGD). Additionally, we set the number of epochs to 4-5, which was determined based on cross-validation results to achieve a balance between model convergence and overfitting. The batch size was set to 32, a typical choice in deep learning applications, to ensure efficient training and stable gradient updates. To improve the model's generalization and prevent overfitting, we applied L2 regularization with a regularization coefficient (alpha) of 0.1, which helps penalize large weights during training. These hyperparameters were fine-tuned to optimize performance and ensure the training process is replicable, providing transparency and reproducibility for future experiments. The final configuration of the hyperparameters, including epoch, batch size, and regularization, was selected after thorough testing and evaluation of the model's performance on validation data.

Model	AUC	CA	F1	Prec	Recall	MCC
NN-MLP (tanh SGD)	1.000	0.992	0.992	0.992	0.992	0.991
NN-MLP (tanh ADAM)	1.000	0.995	0.995	0.995	0.995	0.994
NN-MLP (Relu SGD)	1.000	0.988	0.988	0.988	0.988	0.986
NN-MLP (Relu Adam)	1.000	0.995	0.995	0.995	0.995	0.994
NN-MLP (Logistic SGD)	0.597	0.172	0.126	0.171	0.172	0.059
NN-MLP (Logistic ADAM)	1.000	0.994	0.994	0.994	0.994	0.993

Source: (Research Results, 2025)

Figure 10 Comparison of Hyperparameter Tuning on Mango Leaf Disease Prediction Model

Figure 10 shows a comparison comparative results show that the combination of ReLU and Adam produces the best performance. ReLU is effective in processing non-linear features of mango leaf images, while Adam is adaptive in adjusting the learning rate, accelerating convergence, and avoiding local minima traps.

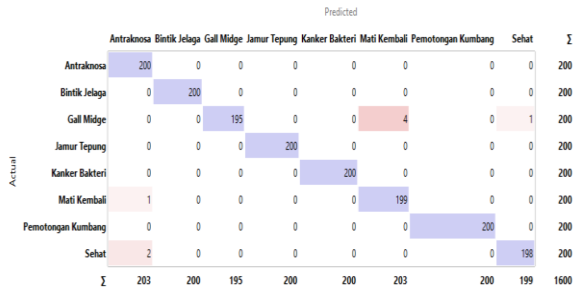


Source: (Research Results, 2025)

Figure 11 Comparison of Hyperparameter Tuning

JITK (JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI KOMPUTER)
VOL. 11. NO. 3 FEBRUARI 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i3.7031

The analysis shows that the combination of ReLU activation and the Adam optimizer provides the best balance of accuracy, training speed, and stability for the MLP-NN mango leaf disease classification model. While Tanh + Adam is stable, it converges more slowly, and ReLU + SGD is effective but requires more careful tuning. Logistic + Adam or SGD is not recommended due to slow convergence and susceptibility to vanishing gradients.



Source: (Research Results, 2025)
Figure 12 Confusion matrix of MLP-NN Model with Relu Activation parameter tuning and Adam Optimization

Figure 12 presents the confusion matrix of the MLP-NN model using ReLU + Adam, showing an overall low error rate. Out of 1,600 test samples, only 7 predictions were incorrect. A closer inspection reveals that the misclassifications are concentrated in specific class pairs. For example, errors between *Kanker Bakteri* and *Mati Kembali* suggest that these classes share highly similar visual features, such as lesion color, texture, or shape, which likely confused the model. Similarly, misclassifications between *Bintik Jelaga* and *Gall Midge* indicate overlapping leaf damage patterns, possibly making these classes difficult to distinguish.

Additionally, the confusion matrix indicates minor misclassifications in other categories, such as a single *Mati Kembali* sample predicted as *Bintik Jelaga*. These errors may arise from subtle intra-class variations, limited data per class, or feature representations that do not fully capture distinguishing patterns. Improving the dataset diversity, augmenting features, or incorporating attention-based mechanisms could help the model better discriminate between visually similar classes.

Overall, these insights emphasize that while ReLU + Adam is optimal for overall performance, targeted interventions such as additional feature engineering or fine-tuning for specific class pairs may further enhance classification accuracy and generalization.

Source: (Research Results, 2025)
Figure 13 Confusion Matrix of MLP-NN Model with Logistic Activation (sigmoid) parameter tuning and SGD Optimization

Figure 13 shows the confusion matrix of MLP-NN Logistic+SGD with the highest error rate. Out of 1600 test data, 1,343 prediction errors were recorded, the rest were predicted correctly.

CONCLUSION

This study demonstrates that the Neural Networks algorithm with the Multi-Layer Perceptron (MLP-NN) architecture is effective in building a mango leaf disease prediction model. Through hyperparameter tuning, it was found that the combination of the ReLU activation function and the Adam optimizer provided the best results in terms of accuracy, stability, and training speed, highlighting the potential of the optimized MLP-NN for predicting plant diseases based on visual symptoms. However, this study has several limitations, the dataset used is relatively small, consisting of [insert number] images, which may limit generalization to unseen data; environmental factors such as lighting, background, and leaf orientation were not considered; image noise and varying image quality were not explicitly addressed and only the MLP-NN architecture was evaluated without comparison to more complex models such as Convolutional Neural Networks (CNN) or ensemble-based approaches. From a practical standpoint, the results indicate potential for integrating the model into a user-friendly disease detection system for farmers to support better crop management practices. For future research, it is recommended to expand the dataset with more diverse images, employ data augmentation techniques to enhance model robustness, explore more complex deep learning architectures like CNNs or hybrid models, automate hyperparameter tuning for more efficient optimization, and conduct comparative studies between MLP-NN and other algorithms such as Random Forest, SVM, and CNN to assess their relative accuracy and efficiency in plant disease prediction. By explicitly addressing these limitations and providing targeted recommendations, this study establishes a foundation for developing more accurate, generalizable, and practical models for plant disease detection.

ACKNOWLEDGEMENTS

The author would like to thank Universitas Potensi Utama for the support and facilities provided during this research process, so that the research could be carried out properly.

REFERENCE

[1] A. Jafar, N. Bibi, R. A. Naqvi, A. Sadeghi-Niaraki, and D. Jeong, “Revolutionizing agriculture with artificial intelligence: plant disease detection methods, applications, and their limitations,” 2024. doi: 10.3389/fpls.2024.1356260.

[2] B. Sambana *et al.*, “An efficient plant disease detection using transfer learning approach,” *Sci. Rep.*, vol. 15, no. 1, 2025, doi: 10.1038/s41598-025-02271-w.

[3] Y. Hu, R. Sandt, and R. Spatschek, “Practical feature filter strategy to machine learning for small datasets in chemistry,” *Sci. Rep.*, vol. 14, no. 1, pp. 1–10, 2024, doi: 10.1038/s41598-024-71342-1.

[4] J. Seetha, R. Ramanathan, V. Goyal, M. Tholkapiyan, C. Karthikeyan, and R. Kumar, “Mango leaf disease classification using hybrid Coyote-Grey Wolf optimization tuned neural network model,” *Multimed. Tools Appl.*, vol. 83, no. 6, 2024, doi: 10.1007/s11042-023-16964-9.

[5] B. U. Mahmud, A. Al Mamun, M. J. Hossen, G. Y. Hong, and B. Jahan, “Light-Weight Deep Learning Model for Accelerating the Classification of Mango-Leaf Disease,” *Emerg. Sci. J.*, vol. 8, no. 1, 2024, doi: 10.28991/ESJ-2024-08-01-03.

[6] M. Hajji *et al.*, “Reducing neural network complexity via optimization algorithms for fault diagnosis in renewable energy systems,” *Ain Shams Eng. J.*, vol. 15, no. 12, p. 103086, 2024, doi: 10.1016/j.asej.2024.103086.

[7] Z. Lun, X. Wu, J. Dong, and B. Wu, *Deep Learning-Enhanced Spectroscopic Technologies for Food Quality Assessment: Convergence and Emerging Frontiers*, vol. 14, no. 13. 2025. doi: 10.3390/foods14132350.

[8] S. Thaseentaj and S. Sudhakar Ilango, “Deep Convolutional Neural Networks for South Indian Mango Leaf Disease Detection and Classification,” *Comput. Mater. Contin.*, vol. 77, no. 3, 2023, doi: 10.32604/cmc.2023.042496.

[9] D. J. Richter, M. Ilias Bappi, S. Sanjay Kolekar, and K. Kim, “A Systematic Review of the Current State of Transfer Learning Accelerated CNN-Based Plant Leaf Disease Classification,” *IEEE Access*, vol. 13, pp. 116262–116303, 2025, doi: 10.1109/ACCESS.2025.3584404.

[10] A. A. M. Faizal, R. G. Jangampally, N. Kaliyan, K. N. Lakshman, D. Karthikeyan, and K. Barathi, “Plant Disease Prediction Using Feature Selection Based Multilayered ANN with Genetic Algorithm,” *Proc. IEEE Int. Conf. Signal Process. Control*, pp. 93–98, 2025, doi: 10.1109/ISPCC66872.2025.11039596.

[11] D. Leite, A. Brito, and G. Faccioli, “Advancements and outlooks in utilizing Convolutional Neural Networks for plant disease severity assessment: A comprehensive review,” 2024. doi: 10.1016/j.atech.2024.100573.

[12] R. A. Rizvee *et al.*, “LeafNet: A proficient convolutional neural network for detecting seven prominent mango leaf diseases,” *J. Agric. Food Res.*, vol. 14, 2023, doi: 10.1016/j.jafr.2023.100787.

[13] D. Guile and J. Popov, “Machine learning and human learning: a socio-cultural and -material perspective on their relationship and the implications for researching working and learning,” *AI Soc.*, vol. 40, no. 2, 2025, doi: 10.1007/s00146-024-01891-6.

[14] L. Kamelia, T. K. A. Rahman, R. R. Nurmalasari, and K. K. Hamdani, “Citrus Tree Nutrient Deficiency Classification: A Comparative Study of ANN and SVM Using Colour-Texture Features in Leaf Images,” *Int. J. Comput. Digit. Syst.*, vol. 15, no. 1, 2024, doi: 10.12785/ijcds/150113.

[15] M. Jamjoom, A. Elhadad, H. Abulkasim, and S. Abbas, “Plant Leaf Diseases Classification Using Improved K-Means Clustering and SVM Algorithm for Segmentation,” *Comput. Mater. Contin.*, vol. 76, no. 1, 2023, doi: 10.32604/cmc.2023.037310.

[16] W. B. Demilie, “Plant disease detection and classification techniques: a comparative study of the performances,” *J. Big Data*, vol. 11, no. 1, 2024, doi: 10.1186/s40537-023-

VOL. 11. NO. 3 FEBRUARI 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i3.7031

00863-9.

[17] S. Kanakala and S. Ningappa, "Detection and Classification of Diseases in Multi-Crop Leaves using LSTM and CNN Models," *J. Innov. Image Process.*, vol. 7, no. 1, 2025, doi: 10.36548/jiip.2025.1.008.

[18] N. Nafi'iyah and J. Maknun, "CNN Architecture for Classifying Types of Mango Based on Leaf Images," *Telematika*, vol. 14, no. 2, 2021, doi: 10.35671/telematika.v14i2.1262.

[19] E. Carrera and V. V. Zozulya, "Carrera Unified Formulation (CUF) for the composite shells of revolution. Equivalent single layer models," *Mech. Adv. Mater. Struct.*, vol. 31, no. 1, 2024, doi: 10.1080/15376494.2023.2218380.

[20] M. A. Hanif, M. Khadimul Islam Zim, and H. Kaur, "ResNet vs Inception-v3 vs SVM: A Comparative Study of Deep Learning Models for Image Classification of Plant Disease Detection," *2024 IEEE Int. Conf. Interdiscip. Approaches Technol. Manag. Soc. Innov. IATMSI 2024*, 2024, doi: 10.1109/IATMSI60426.2024.10502832.

[21] W. Habibi and I. Yuadi, "Evaluating Logistic Regression and SVM for Image Analysis Using VGG-16, VGG-19, and Inception V3 Features," *J. Ilm. Teknol. dan Rekayasa*, vol. 30, no. 2, pp. 136–155, Aug. 2025, doi: 10.35760/TR.2025.V30I2.14056.

[22] M. E. Qureshi, M. H. Ashraf, M. W. Arshad, A. Khan, H. Ali, and Z. -Ul-Abdeen, "Hierarchical Feature Fusion With Inception V3 for Multiclass Plant Disease Classification," *Informatica*, vol. 49, no. 27, Jul. 2025, doi: 10.31449/INF.V49I27.8208.

[23] Y. Arti and A. M. Arymurthy, "Face Spoofing Detection using Inception-v3 on RGB Modal and Depth Modal," *J. Ilmu Komput. dan Inf.*, vol. 16, no. 1, 2023, doi: 10.21609/jiki.v16i1.1100.

[24] P. Kumar, S. Ashtekar, S. S. Jayakrishna, K. P. Bharath, P. T. Vanathi, and M. Rajesh Kumar, "Classification of Mango Leaves Infected by Fungal Disease Anthracnose Using Deep Learning," in *Proceedings - 5th International Conference on Computing Methodologies and Communication, ICCMC 2021*, 2021. doi: 10.1109/ICCMC51019.2021.9418383.

[25] S. K. Paul *et al.*, "An Adam based CNN and LSTM approach for sign language recognition in real time for deaf people," *Bull. Electr. Eng. Informatics*, vol. 13, no. 1, 2024, doi: 10.11591/eei.v13i1.6059.

[26] M. Reyad, A. M. Sarhan, and M. Arafa, "A modified Adam algorithm for deep neural network optimization," *Neural Comput. Appl.*, vol. 35, no. 23, 2023, doi: 10.1007/s00521-023-08568-z.

[27] J. Yang and Q. Long, "A MODIFICATION OF ADAPTIVE MOMENT ESTIMATION (ADAM) FOR MACHINE LEARNING," *J. Ind. Manag. Optim.*, vol. 20, no. 7, 2024, doi: 10.3934/jimo.2024014.

[28] Y. Tian, Y. Zhang, and H. Zhang, "Recent Advances in Stochastic Gradient Descent in Deep Learning," 2023. doi: 10.3390/math11030682.

[29] S. Wojtowytsch, "Stochastic Gradient Descent with Noise of Machine Learning Type Part II: Continuous Time Analysis," *J. Nonlinear Sci.*, vol. 34, no. 1, 2024, doi: 10.1007/s00332-023-09992-0.

[30] S. Aboelenin, F. A. Elbasheer, M. M. Eltoukhy, W. M. El-Hady, and K. M. Hosny, "A hybrid Framework for plant leaf disease detection and classification using convolutional neural networks and vision transformer," *Complex Intell. Syst.*, vol. 11, no. 2, 2025, doi: 10.1007/s40747-024-01764-x.

[31] M. Fahim-Ul-Islam, A. Chakrabarty, R. Rahman, H. Moon, and M. J. Piran, "Advancing mango leaf variant identification with a robust multi-layer perceptron model," *Sci. Rep.*, vol. 14, no. 1, pp. 1–25, Dec. 2024, doi: 10.1038/S41598-024-74612-0;SUBJMETA.