J. Integr. Sci. Technol. 2024, 12(5), 804

Journal of Integrated
SCIENCE & TECHNOLOGY

DV-PSO-Net: A novel deep mutual learning model with Heuristic search using Particle Swarm optimization for Mango leaf disease detection

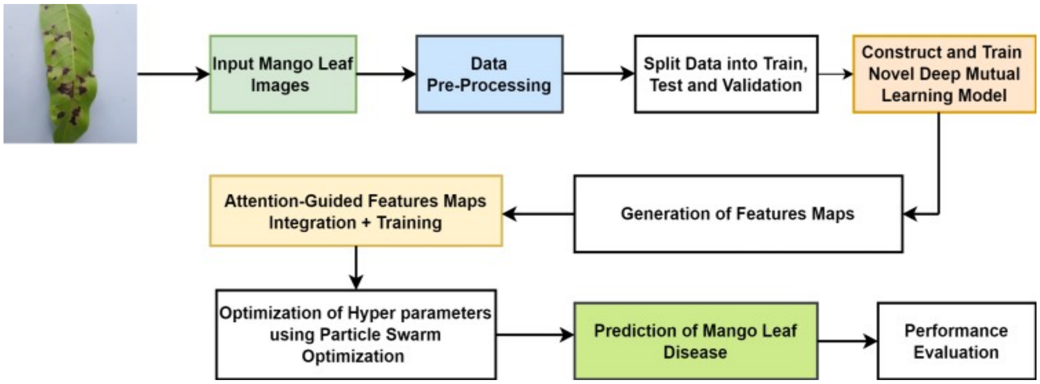
C.P. Vijay,[1*] K. Pushpalatha[2]

[1]*Department of Computer Science and Engineering (AI&ML), Vidyavardhaka College of Engineering, Mysuru, India.*
[2]*Department of Computer Science and Engineering (AI&ML), Sahaydri College of Engineering & Management, Mangalore. India.*

Received on: 08-Oct-2023, Accepted and Published on: 06-Feb-2024

Article

ABSTRACT



Efficient identification of diseases in mango leaves is crucial for maintaining the health and productivity of mango crops. Existing approaches, relying on manual inspection or image processing, are time-consuming, error-prone, and face challenges with various image conditions. This study addresses these issues by proposing a robust machine learning model capable of classifying mango leaf diseases across diverse conditions, including different resolutions, structural complexities, and varying blur levels. The solution involves exploring and optimizing machine learning algorithms, tuning hyperparameters, and developing a predictive model for accurate disease identification based on visual features extracted from leaf images. To overcome these challenges, a novel deep mutual learning model, DVNet, is introduced, leveraging the strengths of Densenet 121 and VGG19 neural networks. Hyperparameter optimization, a systematic procedure for identifying optimal values, is incorporated using Particle Swarm Optimization (PSO). The proposed framework achieves an impressive accuracy of 94.72% in detecting eight distinct disease categories and healthy mango leaves, surpassing existing works in mango leaf disease detection.

Keywords: Mango Leaf disease Detection, Deep mutual learning, DenseNet-121, VGG-19, Particle Swarm optimization (PSO).

INTRODUCTION

Plant diseases pose a significant threat to agricultural production and can lead to substantial economic losses. Early detection and accurate diagnosis of plant diseases are crucial for implementing timely control measures and preventing the spread of infections. In recent years, there has been a growing interest in utilizing deep learning techniques for the detection and diagnosis of plant diseases. Deep learning models, such as convolutional neural networks, have shown promising results in various domains, including computer vision and image recognition. The mango industry, in particular, faces challenges with detecting diseases that affect mango leaves. These diseases, if left untreated, can lead to reduced yield and poor fruit quality. The mango, a fruit that is extensively produced on a global scale, is prone to a range of diseases that have the potential to greatly affect its overall productivity. Plant diseases, particularly those affecting the leaves, are significant variables that have a substantial impact on crop productivity. These illnesses are characterized by the presence of certain symptoms on the leaves, including the formation of lesions, changes in pigmentation, and the development of deformities. The timely identification of these signs is of utmost importance in facilitating efficient disease control and mitigating the dissemination of illnesses inside the orchard. The foresight of outbreaks of diseases would facilitate and streamline the diagnostic procedure, hence promoting the cultivation of high-quality crops.[1]

Conventional approaches to disease identification frequently depend on visual examination, a process that is labor-intensive and potentially prone to inaccuracies. Small farms have a better chance of spotting the infections early and taking preventative action.

---

[*] Corresponding Author: Vijay C.P, Department of CSE(AI&ML)
VVCE, Mysuru
Email: vijay.cp@vvce.ac.in

Cite as: *J. Integr. Sci. Technol., 2024, 12(5), 804.*
URN:NBN:sciencein.jist.2024.v12.804

However, when it comes to large-scale agricultural operations, the process becomes both labor-intensive and financially burdensome. Hence, the pursuit of an automated, precise, efficient, and cost-effective technique for the identification of plant diseases holds significant significance.[2] In recent times, the utilization of deep learning techniques has become increasingly prevalent in the field of image analysis. These techniques have proven to be highly effective in addressing the challenge of automated identification of diseases affecting mango leaves, so presenting a hopeful prospect for this particular application.[3] Recently, there has been a shift in the utilization of deep learning models for the purpose of detecting diseases in mango leaves, as these models are increasingly being implemented in real-world scenarios rather than solely inside research settings. By utilizing smartphones and affordable imaging gear, farmers have the capability to record photographs of mango leaves, which can then be analyzed by deep learning models.[4] This process enables the prompt and precise diagnosis of potential issues. The utilization of real-time evaluation enables farmers to promptly and specifically respond, thereby reducing the adverse effects of diseases on crop productivity. Several automated methods have been published for detecting disease-affected regions, including segmentation and classification methods such as K-means,[5] Support Vector Machine (SVM),[6] and Convolutional Neural Networks (CNNs).[7-9]

Problem Statement: The accurate identification and diagnosis of mango leaf diseases from images present a significant challenge due to inherent complexities such as blurriness, varying orientations of leaves, disease pattern variations and size variations in the leaves. Traditional methods of disease detection struggle to handle these diverse and intricate characteristics, leading to suboptimal results and hindering the effectiveness of agricultural disease management practices. To address these problems, a novel approach is required that integrates advanced deep learning models. Additionally, the optimization of model parameters is crucial for achieving robust performance. This paper aims to provide a solution to the challenges posed by the complex nature of mango leaf images, offering an innovative and efficient method for disease detection that overcomes the limitations of existing approaches

This research's primary findings and contributions are outlined in the subsequent sections.

- This study utilized a unique dataset comprising 4873 pictures of mango leaves exhibiting healthy and disease-affected conditions. Through careful analysis, eight distinct categories of leaf diseases were identified: Anthracnose, Bacterial Canker, Cutting Weevil, Die Back, Gall Midge, Powdery Mildew, Red Rust, and Sooty Mould.
- The utilization of deep learning methods provides a more automated and accurate approach to identifying and diagnosing mango leaf diseases. By leveraging the Densenet-121 and VGG19 deep learning models, demonstrated the potential to revolutionize disease detection in mango plants. These models have the ability to analyze large datasets of images of mango leaves, effectively distinguishing between healthy leaves and those affected by diseases. Furthermore, the integration of a deep mutual learning model enhances the accuracy and robustness of disease detection by leveraging the collaborative learning from multiple models
- Introduced a heuristic search mechanism based on Particle Swarm Optimization. This optimization technique is employed to fine-tune the model parameters, ensuring that the integrated Densenet 121 and VGG19 architecture is optimized for mango leaf disease detection. This contributes to achieving better convergence and performance.
- Combined effect of deep mutual learning and heuristic search using PSO results in a more accurate and efficient model for mango leaf disease detection. The proposed approach aims to outperform existing methods by providing a robust and reliable solution for accurate identification and classification of various diseases affecting mango leaves.
- The deep mutual learning model described in the paper demonstrates a prediction accuracy rate of 94.72% in recognizing eight separate diseases affecting mango leaves. This finding indicates the model's potential for practical implementation in real-time applications.

LITERATURE SURVEY

In recent years, a multitude of methodologies have been devised for the detection of diseases affecting mango leaves. The methods can be broadly classified into illness detection and classification techniques. The majority of strategies employ a deep convolutional network for the purposes of segmentation, obtaining attributes, and categorization.[10] CNNs have demonstrated remarkable efficacy in the domain of image identification, rendering them particularly suitable for the detection of diseases affecting mango leaves. These models represent a class of deep neural network structures that have been specifically engineered to acquire and adjust hierarchical features from visual data, such as photographs. In the domain of mango leaf disease identification, CNNs demonstrate exceptional proficiency in discerning complex patterns and subtle deviations within leaf pictures, hence facilitating precise categorization of leaves as either healthy or diseased. Mishra et al. introduced a CNN framework aimed at the identification of plant diseases.[11] Prasad et al. proposed a CNN-based model which was constructed to detect plant disease and abnormalities at their early stages. The model achieved an accuracy of 90.36% in accurately identifying three unique disease classes.[12] Pham et al. employed an artificial neural network (ANN) based architecture to identify the presence of early-stage disease on plant leaves marked by tiny pathogenic blobs.[13] In their study, Prabhu et al. introduced a CNN structure that employed the levy flight distribution for the purpose of feature selection, while utilizing MobileNetV2 for classification. The model successfully identified three distinct diseases affecting mango leaves.[14]

In their study, Rao et al. employed the AlexNet[15] model to perform automated feature extraction and classification in the context of detecting diseases in mango and grape leaves. The research findings yielded a success rate of 89% in the treatment of mango leaf cases.[16] Rizvee et al. developed LeafNet, a novel detection system, for identifying seven prevalent leaf diseases in mango plants.[3] This framework adheres to the convention established by AlexNet, wherein the number of filters or channels is initially expanded and subsequently reduced during the process of feature map extraction. The aforementioned model achieved an average accuracy rate of 99.3%.[3] In their study, Swaminathan et al. devised a plant leaf disease detection model based on Densenet-121 architecture, which was applied to identify diseases in seven distinct plant species.[17] Nandhini et al. introduced an improved and hyperparameter tuned Densenet-121 framework with the aim of enhancing the accuracy of plant leaf disease identification. The model achieved an average precision score of 98.60%.[18] Seetha et al. proposed the hybrid Coyote Grey Wolf optimization (CO-GWO) method to classify mango leaves as healthy or unhealthy. Classification involves extracting important elements from the split image. The Neural network (NN) classifier classifies, with the weights updated using the proposed algorithm to improve classification accuracy. Precision, recall, and F1 measure, accuracy show that the proposed model is 96.7111%, 97.5712%, 97.1504%, and 96.4792% effective.[28] Performed the comparison of CNN architectures AlexNet, VGG-16, and the ResNet-50 model for mango plant disease classification. These models are trained on the Mendeley dataset and validated with and without transfer learning models. AlexNet, for short (25 layers, 6.2 million individuals parameters) has 94.54% testing accuracy and reduced training time. ResNet-50 (117 layers, twenty-three million parameters) and VGG-16 (16 layers, a total of 138 million features) have 98.56% and 98.26% testing accuracies.[29]

DESIGNED METHODOLOGY

The designed methodology shown in figure 1, for mango leaf disease prediction involves a series of steps designed to enhance the performance and robustness of the models, focusing on image preprocessing, model construction, and optimization. Initially, the mango leaf images are resized to a 256x256 standardized resolution to facilitate seamless integration with the models. This resizing step ensures that the images, regardless of their original dimensions, conform to a consistent format suitable for both DenseNet-121 and VGG-19.

Following normalization, data augmentation techniques are employed to diversify the training dataset. Augmentation methods include random rotations, flips, and shifts, enhancing the model's ability to generalize to varying conditions in mango leaf images. Data pre-processing step is a crucial stage that ensures the input images are appropriately prepared for the subsequent fusion model. Once the pre-processing steps are completed, a deep mutual learning model is constructed by combining DenseNet-121 and VGG-19 architectures. Feature maps are extracted from intermediate layers of both models, and a novel feature integration technique is introduced. This integration incorporates a fusion mechanism that merges complementary information captured by each model, enhancing the model's overall understanding of intricate disease patterns in mango leaves. Combining DenseNet-121 and VGG-19 models in mango disease prediction involves a two-step fusion process: feature-level concatenation and attention-guided feature integration. In the feature-level concatenation step, feature maps extracted from intermediate layers of both DenseNet-121 and VGG-19 are concatenated along the channel axis. This process capitalizes on the dense connectivity of DenseNet-121 and the deep hierarchical feature extraction of VGG-19. The concatenated features form a fused representation that encapsulates both intricate and high-level patterns related to mango diseases. The second step involves attention-guided feature integration. A lightweight attention mechanism is introduced to dynamically modulate the importance of features from DenseNet-121 and VGG-19. This attention mechanism is trained to weigh the contribution of features based on their relevance to disease patterns in the training dataset. By emphasizing the most informative features from each model, the attention-guided integration refines the fused representation, ensuring that the combined features focus on crucial disease-related characteristics.

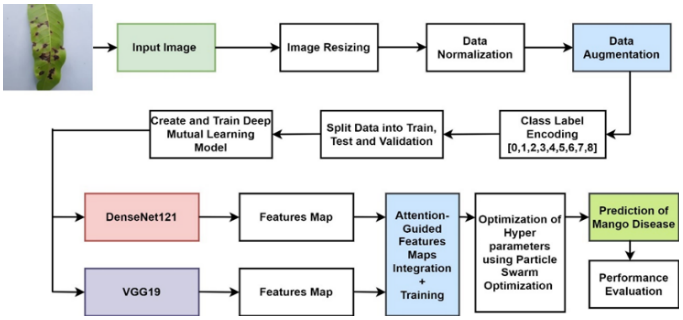


Figure 1. Proposed Methodology for Mango Leaf Disease Prediction.

The methodology for the deep mutual learning model for combining DenseNet-121 and VGG19 to predict of mango leaf diseases involved several key steps. First, the pre-trained DenseNet-121 and VGG19 models, which are well-established convolutional neural networks (CNNs) in computer vision, were selected as the base architectures. These architectures were then fused at an intermediate layer to create a deep mutual learning model. The decision to combine DenseNet-121 and VGG19 was based on their complementary strengths in feature extraction, as DenseNet-121 is known for its dense connectivity and feature reuse, while VGG19 is recognized for its simplicity and effectiveness. To optimize the performance of the deep mutual learning model, hyperparameter tuning was conducted using Particle Swarm Optimization (PSO), a heuristic search algorithm inspired by the social behavior of birds and fish.

The hyperparameters considered for optimization included learning rates, batch sizes, and training epochs. The PSO algorithm efficiently explored the hyperparameter space to find optimal configurations that maximize the accuracy of the model in predicting mango leaf diseases. The selected hyperparameters were then used to train the deep mutual learning model on a dataset of mango leaf images, aiming to achieve superior disease classification performance through the synergistic utilization of DenseNet-121 and VGG19. In detail of the individual model is explained in below subsections

DENSELY CONNECTED CONVOLUTIONAL NETWORKS

DenseNet, also known as "Densely Connected Convolutional Network," is a convolutional neural network (CNN) architecture that has been specifically developed for the purpose of picture categorization and other related computer vision applications [19].

The fundamental concept underlying DenseNet is the notion of dense connectedness.[20] In conventional CNN topologies, the feature maps present at a certain layer are exclusively coupled to the following layers. On the other hand, DenseNet facilitates inter-layer connections in a dense and compact fashion. In a DenseNet, each layer is intricately related to every successive layer.[21] The high level of connection in the network enables the reuse of features and promotes the smooth propagation of gradients during the training process.

DenseNet is constructed with dense blocks, which consist of many convolutional layers within each block. Inside a compact block, the resultant feature maps of each layer are combined with the feature maps of all preceding layers inside the same block by concatenation. Transition layers are employed in the context of dense blocks to effectively decrease the spatial dimensions of feature maps. The transition layers commonly consist of a blend of 1x1 convolution layers, which are employed for dimension reduction and pooling layers. Global average pooling is a commonly used technique in DenseNet for classification, replacing the use of fully connected layers. The process of global average pooling involves the reduction of spatial dimensions in feature maps to a singular value per feature map. This is subsequently followed by the implementation of a softmax layer for the purpose of classification. In order to enhance computing performance, DenseNet has bottleneck layers within its dense blocks. The bottleneck layer is composed of a 1x1 convolution layer, which is subsequently followed by a 3x3 convolution layer. The use of a 1x1 convolutional layer aids in the reduction of the number of feature maps prior to the subsequent 3x3 convolutional layer, which then serves to enlarge them. DenseNet-121 represents a particular iteration of the DenseNet architectural framework. The term "121" in DenseNet-121 denotes the cumulative count of layers within the network architecture

DEEP MUTUAL LEARNING MODEL

In this work, created a deep mutual learning model using DenseNet-121 and VGG-19 models. The details of this structure are discussed in this section. DenseNet is renowned for its characteristic dense connectedness, wherein every layer is provided with input from all preceding levels. The architectural structure is characterized by the presence of compact blocks and transitional levels. The DenseNet-121 architecture shown in figure 2, consists of a total of 120 convolutional layers and 4 average pooling layers. The weights of all layers, even those within the same dense block and transition layers, are distributed across different inputs. This distribution enables deeper layers to utilize features that were extracted in earlier stages. Due to the presence of numerous repetitive features in the transition layers, the layers inside the second and third dense blocks have a tendency to assign lower weights to the output generated by these transition layers.

Let X be the input of layer l in the dense block operation, and the output is H_i. A concatenation takes place here inside the operation, which is represented in the equation below.

$$H_l = \mathrm{Re}LU(Batch\_normalization([X_0, X_1, ..., X_{l-1}]) \qquad (1)$$

Transition layers are employed in order to decrease the spatial dimensions, specifically the width and height while maintaining the same number of channels. If θ is a compression factor, then the transition layer can be presented as in Equation 2.

$$X_{transition} = Conv2D(H, growth\_rate \times \theta, \text{kernel\_size=1}) \qquad (2)$$

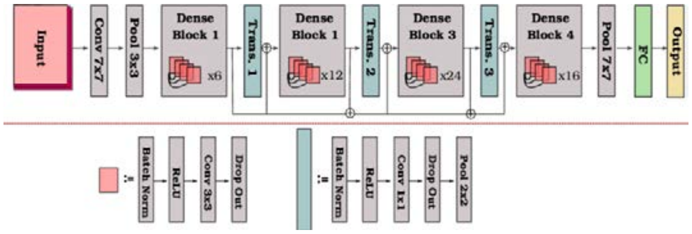


Figure 2. DenseNet-121 architecture

VGG19 is a variant of the VGG architecture as shown in figure 3, characterized by its simplicity with many convolutional layers. Let X_l be the input to the l-th convolutional layer, W_l be the convolutional filter, b_l be the bias, H_l and be the output. Then, the output can be written as Equation 3.

$$H_l = \mathrm{Re}LU(Conv2D(X_l, W_l, padding, stride) + b_l) \qquad (3)$$

Then, the fully connected layer can be presented as below.

$$H_{fully\_connected} = \mathrm{Re}LU(X_{fully\_connected} . W_{fully\_connected} + b_{fully\_conected}) \qquad (4)$$

In a classification task, the output is subjected to a 'softmax' activation function to get class probabilities.

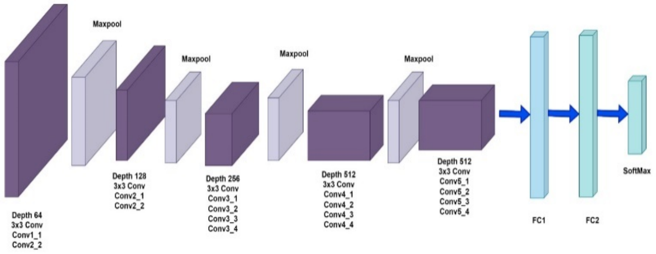


Figure 3. VGG19 framework

PARTICLE SWARM OPTIMIZATION

Deep learning models have the capability to automatically classify various disease categories by analyzing photos of mango leaves. Nevertheless, the efficacy of these models is contingent upon the selection of hyperparameters, including but not limited to the learning rate, number of layers, and activation function. The hyperparameter optimization process is an essential component in searching for the optimal set of hyperparameters that may effectively enhance the accuracy and generalizability of models. Particle Swarm Optimization (PSO) is a metaheuristic algorithm that emulates the collective behavior observed in swarms of avian or aquatic organisms. PSO is a technique that may be utilized to explore and identify the optimal hyperparameters in machine learning models.[22] This is achieved by an iterative process of updating the positions and velocities of a group of potential solutions, known as candidates, based on their fitness and the best solution discovered thus far. PSO has demonstrated efficacy and efficiency in addressing many optimization challenges, encompassing the optimization of hyperparameters for machine learning models.[23]

The coordinates of particle i in hyperparameter space can be denoted by X_i, where i is an integer between 1 and the total number of points in search space. Each X_i is a vector encoding some combination of hyperparameters. Each point's velocity V_i is also a vector, and it's this vector that determines the point's positional evolution over time. Both the particle's individual best-known location (Q_{best}) and the swarm's global best-known position (G_{best}) are used to calculate the most up-to-date velocity. Then, the new velocity of that point is as follows.

$$V_i^{t+1} = w.V_i^t + k_1.r_1.(Q_{best_i} - X_i^t) + k_2.r_2.(G_{best} - X_i^t) \qquad (4)$$

The current iteration is denoted by t, here. Particle momentum is determined by the inertia weight, denoted using w. The motion of the point is affected by two acceleration constants k_1 and k_2, which are defined by the point's local and global optimal locations, respectively. The numbers r_1 and r_2 are completely arbitrary between 0 and 1.

The fitness function ($f(X_i)$)assesses the model's performance by considering the hyperparameters (X_i). The position of the particle that is most widely recognized is revised when the fitness of the current position surpasses the fitness of the previously acknowledged best-known position (Q_{best}). The global best-known position (G_{best}) is updated whenever a particle within the swarm discovers a superior position. The PSO technique employs an iterative process to continually update the positions and velocities of particles. The objective is to identify the optimal hyperparameter configuration that either reduces or maximizes the fitness function.

ALGORITHM

The methods utilized in this research are presented in the visual depiction depicted in Figure 4. The primary objective of our study is to forecast the occurrence of several disease classes in mango leaves, encompassing both healthy and eight distinct diseases. The picture data was gathered from several sources in order to create a more comprehensive dataset with a substantial quantity of photographs. A complete statistical analysis was performed on the dataset through the use of exploratory data analysis techniques. Following this, the examination is presented using a visual representation of data. Subsequently, the dataset was divided into separate subsets to facilitate the training and testing procedures. Following that, the dataset was subjected to data augmentation in order to address the problem of data imbalance. Subsequently, the DV-PSO-Net model was constructed to predict the occurrence of mango leaf disease. The methodology provided in this study involves the extraction of relevant features through the utilization of a concatenated model that is built on the DenseNet-121 and VGG-19 frameworks. The PSO technique was employed in this work as the hyperparameter optimizer for the suggested model. The hyperparameters encompassed in this set consist of swarm size, inertia weight, and acceleration coefficients. The utilization of the PSO approach has been observed to enhance the performance metrics in the identification of mango leaf diseases.

PSO is employed as a powerful heuristic search algorithm to optimize the hyperparameters crucial for the performance of a deep mutual learning model combining DenseNet 121 and VGG19. The objective is to enhance the predictive accuracy of the model in the context of identifying mango leaf diseases. Hyperparameters such as learning rate, epochs, batch size, and network weights play a pivotal role in determining the model's efficacy. PSO is utilized as an optimization technique to automatically search and discover an optimal set of hyperparameter values that maximizes the model's predictive performance. The application of PSO involves representing potential solutions, or particles, within a multidimensional search space defined by the hyperparameters. Each particle in the swarm corresponds to a unique combination of hyperparameter values. The fitness of a particle is determined by evaluating the performance of the associated deep mutual learning model on a training dataset, where the model is configured with the specific hyperparameters represented by that particle. The fitness function typically involves a performance metric such as accuracy or F1 score, reflecting the model's ability to accurately classify instances of mango leaf diseases.

In PSO selecting the specific values for parameters such as swarm size, inertia weight, cognitive weight, and social weight significantly impact the convergence and exploration-exploitation trade-off during the optimization process. The swarm size determines the number of particles in the search space, affecting the diversity of the search. Common values range from 10 to 100, and the optimal value depends on the problem. Inertia weight balances the particle's current velocity with its historical velocity, determining the balance between exploration and exploitation. Values typically range between 0 and 1. Lower values promote local exploration, while higher values favor global exploration. The cognitive and social weights govern the influence of a particle's personal best and the global best solutions on its movement. Common values are between 1 and 2, with higher values emphasizing individual learning. The careful selection of these parameters is crucial for achieving an optimal balance between exploration, ensuring broad coverage of the search space, and exploitation, focusing on promising regions.

Throughout the optimization process, particles in the swarm iteratively update their positions and velocities based on both their individual experiences (local best-known position) and the collective knowledge of the entire swarm (global best-known position). This dynamic interaction allows the swarm to explore the hyperparameter space efficiently, converging towards configurations that yield superior model performance. The PSO algorithm fine-tunes hyperparameters, adjusting learning rates, epochs, batch sizes, and network weights to discover an optimized configuration for the deep mutual learning model. The final result is a set of hyperparameters that significantly improves the accuracy and reliability of the combined DenseNet 121 and VGG19 model for predicting mango leaf diseases. This approach not only enhances model performance but also demonstrates the effectiveness of PSO as a tool for automated hyperparameter tuning in complex deep learning architectures.

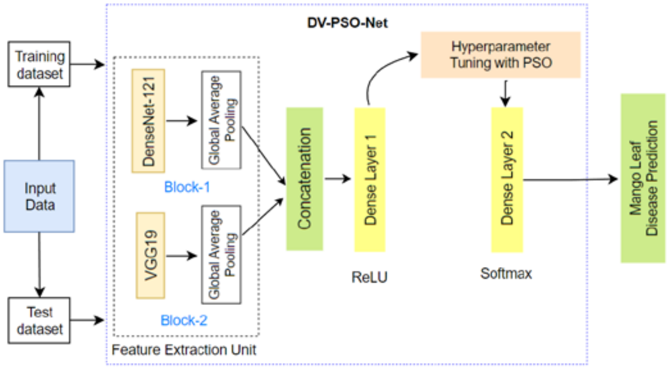


Figure 4. The designed Fused DV-PSO-Net Model for Mango Disease Prediction.

The fusion of DenseNet-121 and VGG-19 models for mango disease prediction brings forth a synergistic approach that leverages the unique strengths of both architectures. DenseNet-121, known for its dense connectivity patterns, enables efficient information flow between layers by concatenating feature maps from preceding layers. This characteristic facilitates the model's ability to capture intricate patterns and hierarchical representations within the input images, particularly beneficial when dealing with complex structures such as diseased regions in mangoes. On the other hand, VGG-19, renowned for its simplicity and deep convolutional layers, excels in capturing low to high-level features across the image, providing a robust foundation for recognizing disease-related patterns. By combining these models, harnessed the detailed feature extraction capability of DenseNet-121 and the depth of VGG-19, creating a comprehensive framework that excels in discerning nuanced patterns indicative of various mango diseases. The necessity for combining models arises from the diverse nature of mango disease patterns, wherein certain diseases may manifest with intricate textures and structures, while others may present more straightforward visual cues. The fusion of DenseNet-121 and VGG-19 enables a more holistic understanding of mango disease scenarios, allowing the model to exploit the strengths of each architecture based on the specific characteristics of the input image. This fusion enhances the model's capacity for generalization across a broad spectrum of disease presentations, thereby improving the overall accuracy and robustness of mango disease prediction. In essence, the combined architecture harnesses the complementary features of DenseNet-121 and VGG-19, offering a versatile and powerful solution for accurate mango disease diagnosis in agricultural settings.

In this study, the combination of DenseNet121 and VGG19 for mango leaf disease prediction from mango leaf images, selection of hyperparameters such as learning rate, batch size, epochs, dense drop outs, number of neurons in dense layer and optimization of these hyperparameters plays pivotal role in building the robust model. The general outline of the algorithm of this proposed methodology is presented below.

Algorithm: Mango Disease Classification using DV-PSO-Net

```
Input:
   - Dataset D containing mango images with corresponding disease labels
   - PSO parameters: population size, maximum iterations, inertia weight, cognitive factor, social factor
Output: Trained model for mango disease classification
Step 1: Data Preprocessing(D): Perform data preprocessing steps, including resizing, normalization, and augmentation
Step 2: Initialize PSO Parameters(): Initialize PSO parameters such as population size, maximum iterations, inertia weight, cognitive factor, and social factor
Step 3: Initialize Population(): Initialize the population of particles with random values for hyperparameters
Step 4: while (termination criteria not met) do
Step 5:     for each particle in the population, do
Step 6:         Update Particle Velocity($V_i$)
Step 7:         Update Particle Position($X_i$)
Step 8:         Evaluate Fitness()
Step 9:         Update Personal Best($Q_{best}$)
Step 10:        Update Global Best($G_{best}$)
Step 11:    end for
Step 12: end while
Step 13: Extract Global Best($G_{best}$):  Extract the best set of hyperparameters found during the PSO search
Step 14: Model Training(D, Global Best): Train the combined DenseNet-121 and VGG19 model with the hyperparameters obtained from the PSO search
Step 15: return Trained Model
```

The algorithm aims to optimize hyperparameters for mango disease classification using a combination of DenseNet-121 and VGG19 models through the PSO technique. The algorithm begins with data preprocessing, ensuring that the mango image dataset is appropriately resized, normalized, and augmented to enhance the model's generalization capabilities. Subsequently, PSO parameters are initialized, including population size, maximum iterations, inertia weight, cognitive factor, and social factor. The PSO optimization process iteratively refines a population of particles representing different hyperparameter configurations such as learning rate, number of epochs, batch size and network weights. In each iteration, the algorithm updates particle velocities and positions based on the PSO dynamics. The fitness of each particle, representing the performance of a specific hyperparameter configuration, is evaluated using a predefined fitness function. Personal and global best solutions are updated to guide the search towards promising regions in the hyperparameter space. The algorithm terminates when a predefined termination criterion is met. The global best solution, representing the optimal set of hyperparameters, is then extracted. Finally, the DenseNet-121 and VGG19 model is trained using the obtained hyperparameters, resulting in a trained model for mango disease classification. This algorithm offers a systematic and automated approach to hyperparameter tuning, leveraging the collective intelligence of particles in the PSO search space to enhance the performance of the combined model for accurate disease prediction in mango leaves

RESULTS AND DISCUSSION

One of the primary obstacles faced in the development of systems for identifying pests and diseases in mangoes is the lack of sufficiently large and precisely annotated datasets. The effectiveness of deep learning (DL) models is limited by the lack of sufficient training data. To mitigate the issue of overfitting, reduce mistakes, and improve the model's ability to generalize, deep learning models necessitate the utilization of adequately large training datasets. In recent times, scholars have utilized a diverse range of computer-assisted and machine-learning techniques to classify different types of illnesses affecting mango leaves. However, it is important to note that these methodologies have shown limitations in their effectiveness. These limitations can be attributed to various factors such as increased dimensionality of features, overfitting, higher computational complexity, longer time consumption, limited consideration of features, inadequate feature quality, and lower segmentation outcomes.

DATA SETS

Compiled a unique dataset of 4873 images that are categorized into nine distinct classes as presented in figure 5. Despite the dataset including a substantial amount of data, preparation is necessary for the entire dataset due to variations in the shapes and sizes of the photos. Table 1 represents the count of the mango leaf images utilized in each category of diseases and healthy mango leaves images count.

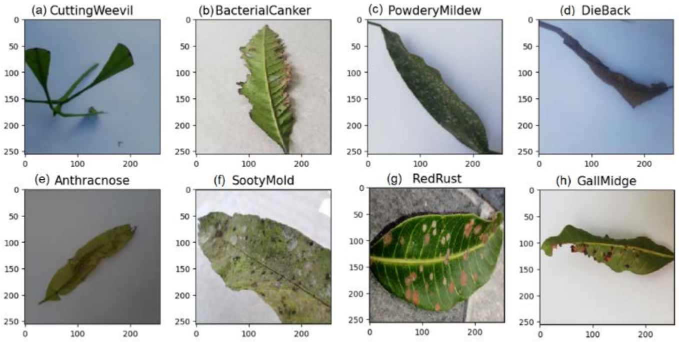


Figure 5. Examples of Eight Distinct Mango Leaf Diseases.

Table 1. The Image data count present in each class of the dataset.

Mango Leaf Condition			Number of Images
Healthy			670
With Disease	1.	Anthracnose	743
	2.	Bacterial Canker	500
	3.	Cutting Weevil	500
	4.	Die Back	500
	5.	Gall Midge	500
	6.	Powdery Mildew	500
	7.	Red rust	163
	8.	Sooty Mould	797

In this experiment, we implemented the models with GPU backing. Therefore, we used a Lenovo ThinkPad E16 40.64cms computer with Windows 11 operating system, which has a 13th Generation Intel® Core™ i5-1335U Processor. The system used a 40 GB memory with NVIDIA® GeForce® MX550 graphics. The PyTorch deep learning platform was chosen in this study to facilitate the quick training and testing of datasets. The implementation of these programs employs the Keras framework, which is a Python toolkit for deep neural networks that is open-source. In order to conduct image classification on a substantial dataset, it is imperative to partition the data into two distinct subsets: one designated for training purposes and the other intended for validation. The training set is utilized for the purpose of training the neural network model, whilst the validation set is employed to assess its performance and mitigate the risk of overfitting. A commonly employed approach involves utilizing a predetermined ratio, such as 80:20, to partition the data into distinct training and validation sets.

A variety of CNN architectures are available for selection in order to train our model. This study aims to do a comparative analysis of the outcomes obtained by employing ResNet, VGG19, and Inceptionv3 models. The ResNet architecture is a type of deep residual network that addresses the issue of disappearing gradients by incorporating skip connections. The VGG19 architecture is characterized by its depth, employing a series of convolutional layers with tiny filters to effectively extract data. The Inception V3 model is a convolutional neural network architecture that incorporates inception modules. These inception modules are sub-networks designed to concatenate the outputs of various convolutional operations. The learning rate, batch size, and number of epochs are just a few of the hyperparameters that must be tweaked before we can begin training our model. The learning rate parameter governs the magnitude of weight updates performed by the model at each iteration. The batch size parameter determines the number of samples that are simultaneously inputted into the model during training or inference. The quantity of epochs determines the number of complete iterations the model undergoes during the training process, encompassing the whole training dataset. Grid search and random search are two commonly used methods for determining the ideal values of hyperparameters that minimize the validation loss and maximize the validation accuracy

A range of criteria were utilized to assess the different models given in the work presented here. The confusion matrix is a matrix that has been expressly created to assess the efficacy of a certain methodology within the realms of pattern recognition and machine learning, particularly in the context of classification problems. The classifier's output yields several parameters. The above-mentioned classes cover both positive and negative consequences, that can be either valid or invalid. Various evaluation measures can be obtained from these. The measures of precision, recall, F1-score, and accuracy are extensively acknowledged and held in high regard within the academic community.[24]

The statistical notion of precision entails calculating the proportion of correct predictions generated by an algorithm in relation to the overall sum of forecasts. The recall rate can be calculated by dividing the sum of true positives and false negatives by the number of true positives alone. The incorporation of precision and recall measurements can offer significant benefits in gaining a more comprehensive comprehension of the efficacy of a certain approach while also ensuring that the results conform to established standards. Nevertheless, the issue arises when depending simply on these measures for comparison, as it involves the selection of the best appropriate strategy for the data while concurrently evaluating numerous approaches proficient on the same information sets. The importance of the F1 score is of utmost significance. The F1 score is a quantitative measure that quantifies the harmonic mean of the precision and recall metrics.

The methodology employed in this research entails the extraction of pertinent features by employing a deep mutual learning model constructed using the concatenated version of DenseNet-121 and VGG-19 frameworks, which is subsequently followed by two fully connected layers. The initial dense layer has a Rectified Linear Unit (ReLU) activation function. At this step of the study, the PSO technique was utilized as the hyperparameter optimizer for the created model. The set of hyperparameters includes swarm size, inertia weight, and acceleration coefficients. The second dense layer is triggered using the 'Softmax' function since the target variable requires a prediction over nine distinct classes. The implementation of the PSO technique has been found to improve the performance metrics in the detection and classification of mango leaf diseases. In accordance with the technique adopted in this work, the normalization step was conducted by dividing each pixel value of the images in both the original and supplemented datasets by 255. Following this, the photos underwent a resizing process to conform to the standard dimensions that were suitable for the corresponding models.

In the conducted experiments, the deep mutual learning model DenseNet-121 and VGG19 was trained using various hyperparameter configurations obtained through Particle Swarm Optimization (PSO) heuristic search. The experiments aimed to explore the impact of different learning rates, batch sizes, and training epochs on the model's performance. The obtained results shown in table 2 gives that the Experiment 5, with a moderate learning rate, PSO iterations, and random initialization of VGG19 weights, achieved a high training accuracy of 94% and a commendable test accuracy of 91%. On the other hand, Experiment 7, involving a lower learning rate, PSO iterations, and random initialization of VGG19 weights, resulted in a lower training accuracy of 90% and a test accuracy of 85%. These findings suggest that a careful balance of hyperparameters, particularly learning rates and weight initializations, is crucial for achieving optimal performance in the deep mutual learning model.

The details of the hyperparameters tuning based on the PSO technique in seven different trials are described in Table 3. Following this, the plot depicted in Figure 6 shows the accuracies in seven different trial runs of the proposed models in different hyperparameter value combinations. Found that the most efficient approach was the 4th trial. In this case, the mode achieved an accuracy of 94.72% when the swarm size is 12, and the inertia weight is set at 0.4. The first and second acceleration coefficients are 2.5 and 1.6, respectively. Figure 7, shows the Training and test loss in case of trial with highest accuracy (4th trial). Table 5 represents the comparisons of various models for mango leaf disease detection. This table also compares the number of leaf disease categories for each study. In this comparative table 4, various methods for mango leaf disease classification are evaluated based on their respective classes, employed models, and average accuracy. Gulavani et al. [25] utilized a CNN-ResNet50 model achieving an accuracy of 91% for 5 Classes. Prabhu et al. [14] applied MobileNetV2 for 4 Classes and achieved an average accuracy of 92.15%. Our Previous work[26] employed EfficientNetB4-CNN, reaching an accuracy of 93.01% for 9 Classes. The proposed method, presented in this paper, surpasses the existing approaches by introducing DV-PSO-Net, achieving the highest accuracy of 94.72% for 9 classes. This suggests that the proposed method outperforms the existing referenced models, providing a more effective solution for mango leaf disease classification, particularly in 9 classes.

Table 2. Hyperparameters Tuning based on PSO technique in various Experiment (Exp) trials.

Learning Rate	Batch Size	Epochs	PSO Iterations	DenseNet 121 Weights	VGG19 Weights	Train Accuracy
0.0010	32	20	50	'imagenet'	'imagenet'	0.92
0.0005	64	15	75	'random'	'random'	0.89
0.0020	128	25	100	'imagenet'	'random'	0.94
0.0015	64	30	50	'random'	'imagenet'	0.91
0.0018	**128**	**18**	**60**	**'imagenet'**	**'random'**	**0.94**
0.0022	32	25	80	'random'	'imagenet'	0.93
0.0013	64	22	70	'imagenet'	'random'	0.90
0.0016	128	20	90	'random'	'imagenet'	0.93
0.019	32	30	60	'imagenet'	'random'	0.92
0.0012	64	15	40	'random'	'imagenet'	0.91

Table 3. PSO parameters obtained from various trials.

Trials	Swarm Size	Inertia Weight	Cognitive Weight	Social Weight	Model Performance (Accuracy)
1	10	0.5	2.0	2.0	0.9400
2	20	0.7	1.5	1.9	0.9232
3	15	0.6	2.0	1.8	0.9351
4	12	0.4	2.0	1.6	0.9472
5	18	0.8	1.8	2.0	0.9211
6	16	0.6	2.0	1.6	0.9139
7	12	0.8	1.9	1.8	0.9301

The confusion matrix shown in figure 8, reveals the model's performance across nine distinct mango disease classes. Notably, the diagonal elements represent true positives, indicating the model's accurate predictions for each class. Classes such as "Powdery Mildew" (Class 6) and "Red Rust" (Class 7) exhibit perfect predictions, with all instances correctly classified. The off-diagonal elements highlight instances of misclassification, with a few false positives and false negatives observed across various classes. For instance, "Anthracnose" (Class 1) shows a slight confusion with "Bacterial Canker" (Class 2) and "Cutting Weevil" (Class 3), as evident from the non-zero values in the respective cells. The overall distribution of values in the confusion matrix indicates a commendable degree of accuracy, emphasizing the model's ability to differentiate between mango diseases.

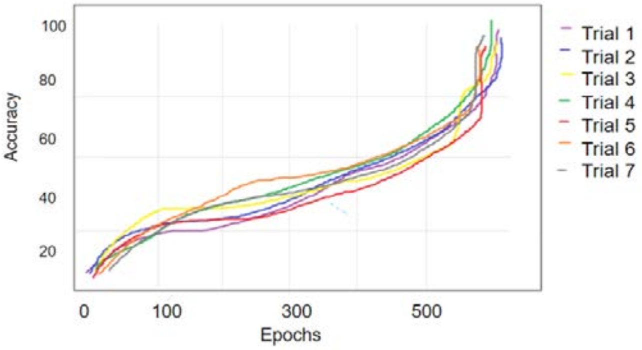


Figure 6. Accuracy of the model in different trials

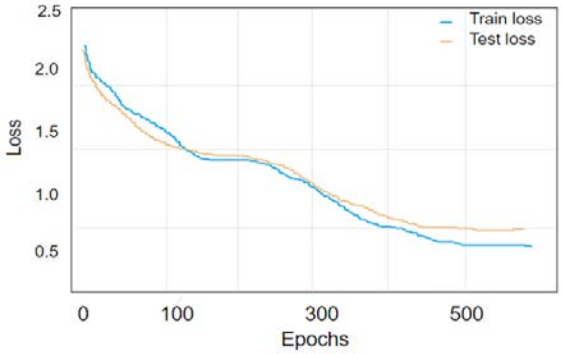
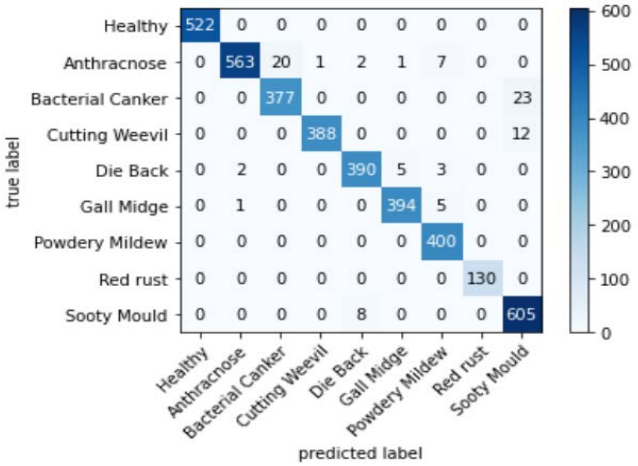


Figure 7. Training and test loss in case of trial with highest accuracy (4th trial).



	Healthy	Anthracnose	Bacterial Canker	Cutting Weevil	Die Back	Gall Midge	Powdery Mildew	Red rust	Sooty Mould
Healthy	522	0	0	0	0	0	0	0	0
Anthracnose	0	563	20	1	2	1	7	0	0
Bacterial Canker	0	0	377	0	0	0	0	0	23
Cutting Weevil	0	0	0	388	0	0	0	0	12
Die Back	0	2	0	0	390	5	3	0	0
Gall Midge	0	1	0	0	0	394	5	0	0
Powdery Mildew	0	0	0	0	0	0	400	0	0
Red rust	0	0	0	0	0	0	0	130	0
Sooty Mould	0	0	0	0	8	0	0	0	605

Figure 8. Confusion Matrix

The performance evaluation results shown in table 4, across the nine classes indicate a generally robust predictive capability of the model for mango disease classification. Notably, classes such as "Healthy" (Class 0), "Powdery Mildew" (Class 6), and "Red Rust" (Class 7) exhibit exceptional precision, recall, and F1-scores, indicating accurate identification and classification of these conditions. Other classes, including "Anthracnose" (Class 1) and "Gall Midge" (Class 5), demonstrate high precision and recall, highlighting the model's proficiency in distinguishing these diseases. The overall F1-scores for all classes underscore the balanced performance in terms of precision and recall, emphasizing the model's ability to provide accurate predictions across a diverse range of mango diseases. Additionally, the model showcases high precision in the detection of healthy mango leaves, critical for avoiding false positives. These results collectively signify a well-performing model that effectively addresses the complexity of mango disease classification.

Table 4. Results of precision, recall and f1-score of each class.

Mango Leaf Disease Classes	**Performance Evaluation Metrix's**		
	Precision	**Recall**	**F1-score**
Class 0 (Healthy)	1.00	0.94	0.97
Class 1 (Anthracnose)	0.96	0.94	0.95
Class 2 (Bacterial Canker)	0.94	0.94	0.94
Class 3 (Cutting Weevil)	0.97	0.97	0.97
Class 4 (Die Back)	0.975	0.97	0.97
Class 5 (Gall Midge)	0.98	0.98	0.98
Class 6 (Powdery Mildew)	1.00	1.00	1.00
Class 7 (Red Rust)	1.00	1.00	1.00
Class 8 (Sooty Mould)	0.97	0.98	0.98

Table 5. Various model comparisons for mango leaf disease detection

Paper	**Class**	**Model**	**Average Accuracy**
Gulavani *et al.* [25]	5	CNN-ResNet50	91%
Prabhu *et al.* [14]	4	MobileNetV2	92.15
Vijay *et al.* [26]	9	EfficientNetB4-CNN	93.01%
Proposed method	**9**	**DV-PSO-Net**	**94.72%**

It was seen that the designed model in this study performed better than CNN-ResNet50,[25] MobileNetV2,[14] and our previously reported model EfficientNetB4-CNN.[26] DV-PSO-Net surpasses EfficientNetB4-CNN due to the PSO-based hyperparameter tuning method.

CONCLUSION

Several networks can share predictions and jointly optimize a loss function to improve their performance using deep mutual learning. In this way, they can improve the generalization capability of each network while still benefiting from the diversity and complementarity of their architectures. The approach taken in this study involves the use of a deep mutual learning model built using a combination of the DenseNet-121 and VGG-19 frameworks, followed by two fully connected layers for feature extraction. The PSO method was used to optimize the model's hyperparameters at this stage of the research. The size of the swarm, mass of the inertia, and acceleration coefficients are all part of the collection of hyperparameters. The model was trained and tested using a dataset consisting of 4873 images of mango leaves that are either healthy or damaged, which has identified the existence of eight unique forms of leaf diseases. To name just a few, we have anthracnose, bacterial cellulase, cutting weevil, die back, gall midge, powdery mildew, red rust, and sooty mold. The study proposes a hybrid model that demonstrates a recognition accuracy rate of 94.72% for recognizing leaf diseases in mango plants. Experiments demonstrate that DV_PSO-Net can greatly boost network performance. It can make the networks more amenable to real applications by decreasing their inference time and memory usage. We think our method offers a novel perspective and potential solution for disease detection in mango leaves and other plant materials. The adoption of the PSO technique has been proven to increase the performance metrics in the early prediction of mango leaf diseases. The complexities associated with the interactions of hyperparameters are well-documented. The PSO operates by treating each hyperparameter in isolation, potentially constraining its capacity to effectively capture complex interdependencies among the parameters.

ACKNOWLEDGMENTS:

Sincere Thanks to Principal, HoD Dept of CSE(AI&ML) and faculties for their constant support in this work.

CONFLICT OF INTEREST STATEMENT

The author has no conflict of interest.

REFERENCES AND NOTES

1. V. Gautam, R.K. Ranjan, P. Dahiya, A. Kumar. ESDNN: A novel ensembled stack deep neural network for mango leaf disease classification and detection. *Multimed. Tools Appl.* **2024**, 83 (4), 10989–11015.
2. S. Arivazhagan, S.V. Ligi. Mango Leaf Diseases Identification Using Convolutional Neural Network. *Int. J. Pure Appl. Math.* **2018**, 120 (6), 11067–11079.
3. R.A. Rizvee, T.H. Orpa, A. Ahnaf, et al. LeafNet: A proficient convolutional neural network for detecting seven prominent mango leaf diseases. *J. Agric. Food Res.* **2023**, 14, 100787.
4. A. Koirala. Precision Agriculture : Exploration of machine learning approaches for assessing mango crop quantity, Doctoral dissertation, CQ University, **2020**.
5. M.R. Mia, A.C. Chhoton, M.H. Mozumder, S.A. Hossain, A. Hossan. An Approach for Mango Disease Recognition using K-Means Clustering and SVM Classifier. In *Proceedings of the 2019 8th International Conference on System Modeling and Advancement in Research Trends, SMART 2019*; IEEE, **2020**; pp 404–409.
6. M.R. Mia, S. Roy, S.K. Das, M.A. Rahman. Mango leaf disease recognition using neural network and support vector machine. *Iran J. Comput. Sci.* **2020**, 3 (3), 185–193.
7. S. Arya, R. Singh. A Comparative Study of CNN and AlexNet for Detection of Disease in Potato and Mango leaf. In *IEEE International Conference on Issues and Challenges in Intelligent Computing Techniques, ICICT 2019*; **2019**; Vol. 1, pp 1–6.
8. A. Rajbongshi, T. Khan, M.M. Rahman, et al. Recognition of mango leaf disease using convolutional neural network models: A transfer learning approach. *Indones. J. Electr. Eng. Comput. Sci.* **2021**, 23 (3), 1681–1688.
9. U.P. Singh, S.S. Chouhan, S. Jain, S. Jain. Multilayer Convolution Neural Network for the Classification of Mango Leaves Infected by Anthracnose Disease. *IEEE Access* **2019**, 7, 43721–43729.
10. R. Saleem, J.H. Shah, M. Sharif, G.J. Ansari. Mango Leaf Disease Identification Using Fully Resolution Convolutional Network. *Comput. Mater. Contin.* **2021**, 69 (3), 3581–3601.
11. S. Mishra, S. Satapathy, G. Dengia, B. Mulatu, F. Tadele. Identification and Classification of Mango Leaf Disease Using Wavelet Transform based Segmentation and Wavelet Neural Network Model. *Ann. Rom. Soc. Cell Biol.* **2021**, 25 (2), 1982–1989.
12. S.R. Prasad, G.S. Thyagaraju. Leaf analysis based early plant disease detection using Internet of Things, Machine Learning and Deep Learning: A comprehensive review. *J. Integr. Sci. Technol.* **2023**, 12 (2), 734.
13. T.N. Pham, L. Van Tran, S.V.T. Dao. Early Disease Classification of Mango Leaves Using Feed-Forward Neural Network and Hybrid Metaheuristic Feature Selection. *IEEE Access* **2020**, 8, 189960–189973.
14. M. Prabu, B.J. Chelliah. Mango leaf disease identification and classification using a CNN architecture optimized by crossover-based levy flight distribution algorithm. *Neural Comput. Appl.* **2022**, 34 (9), 7311–7324.
15. Z.-W. Yuan, J. Zhang. Feature extraction and image retrieval based on AlexNet. In *Eighth International Conference on Digital Image Processing (ICDIP 2016)*; **2016**; Vol. 10033, p 100330E.
16. U. Sanath Rao, R. Swathi, V. Sanjana, et al. Deep Learning Precision Farming: Grapes and Mango Leaf Disease Detection by Transfer Learning. *Glob. Transitions Proc.* **2021**, 2 (2), 535–544.
17. A.S. Vellaichamy, A. Swaminathan, C. Varun, K. S. Multiple Plant Leaf Disease Classification Using Densenet-121 Architecture. *Int. J. Electr. Eng. Technol.* **2021**, 12 (5), 38–57.
18. S. Nandhini, K. Ashokkumar. An automatic plant leaf disease identification using DenseNet-121 architecture with a mutation-based henry gas solubility optimization algorithm. *Neural Comput. Appl.* **2022**, 34 (7), 5513–5534.
19. G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger. Densely connected convolutional networks. In *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*; **2017**; Vol. 2017-January, pp 2261–2269.
20. G. Huang, Z. Liu, G. Pleiss, L. Van Der Maaten, K.Q. Weinberger. Convolutional Networks with Dense Connectivity. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, 44 (12), 8704–8716.
21. Y. Yang, Q.M.J. Wu, X. Feng, T. Akilan. Recomputation of the dense layers for performance improvement of DCNN. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, 42 (11), 2912–2925.
22. Y. Wang, H. Zhang, G. Zhang. cPSO-CNN: An efficient PSO-based algorithm for fine-tuning hyper-parameters of convolutional neural networks. *Swarm Evol. Comput.* **2019**, 49, 114–123.
23. P.R. Lorenzo, J. Nalepa, M. Kawulok, L.S. Ramos, J.R. Pastor. Particle swarm optimization for hyper-parameter selection in deep neural networks. In *GECCO 2017 - Proceedings of the 2017 Genetic and Evolutionary Computation Conference*; **2017**; pp 481–488.
24. M. Heydarian, T.E. Doyle, R. Samavi. MLCM: Multi-Label Confusion Matrix. *IEEE Access* **2022**, 10, 19083–19095.

25. S. Gulavnai, R.J.I.J. R. T. Patil, Engineering. Deep Learning for Image Based Mango Leaf Disease Detection. *Int. J. Recent Technol. Eng.* **2019**, 8 (3S3), 54–56.
26. C.P. Vijay, K. Pushpalatha. Revolutionizing Mango Leaf Disease Detection: Leveraging Segmentation and Hybrid Deep Learning for Enhanced Accuracy and Sustainability. *Int. J. Intell. Syst. Appl. Eng.* **2023**, 11 (4), 121–131.
27. V. Gautam, J. Rani. Mango Leaf Stress Identification Using Deep Neural Network. *Intell. Autom. Soft Comput.* **2022**, 34 (2), 849–864.
28. J. Seetha, R. Ramanathan, V. Goyal, et al. Mango leaf disease classification using hybrid Coyote-Grey Wolf optimization tuned neural network model. *Multimed. Tools Appl.* **2023**, 83 (6), 17699–17725.
29. B. Jayanthi, L.S. Kumar. Comparison of convolutional neural networks architectures for mango leaf classification. *Int. J. Comput. Vis. Robot.* **2023**, 14 (1), 84–98.