Heliyon 10 (2024) e26465

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

A new AI-based approach for automatic identification of tea leaf disease using deep neural network based on hybrid pooling

Qidong Heng [a,**], Sibo Yu [b], Yandong Zhang [c,*]

[a] School of Public Administration, Beijing City University, Beijing, 100083, China
[b] School of Information Science and Engineering, Beijing City University, Beijing, 100083, China
[c] Anxi Campus - Anxi College of Tea Science, Fujian Agriculture and Forestry University, Fuzhou, 350000, Fujian, China

A R T I C L E I N F O

Keywords:
Tea leaf disease detection
Deep learning
Hybrid pooling
Machine vision

A B S T R A C T

The degree of production efficiency and the quality of the commodities produced may both be directly impacted by the presence of illnesses in tea leaves. These days, this procedure may be automated with the use of artificial intelligence tools, and a number of approaches have been put out to satisfy these needs. Nonetheless, current research efforts have focused on improving diagnosis accuracy and expanding the variety of illnesses that might affect tea leaves. In this article, a new method is proposed for accurately diagnosing tea leaf diseases using artificial intelligence techniques. In the proposed method, the input images are preprocessed to remove redundant information. Then, a hybrid pooling-based Convolutional Neural Network (CNN) is employed to extract image features. In this method, the pooling layers of the CNN model are randomly adjusted based on either max pooling or average pooling functions. This strategy can enhance the efficiency of the CNN-based feature extraction model. In this method, the pooling layers of the CNN model are randomly adjusted based on either max pooling or average pooling functions. This strategy can enhance the efficiency of the CNN-based feature extraction model. After feature extraction, a weighted Random Forest (WRF) model is used for the detection of tea leaf diseases. The outputs of the decision tree models and their corresponding weights are used to identify tea leaf illnesses in this classification model, where each tree in the random forest is given a weight depending on how well it performs. The Cuckoo Search Optimization (CSO) method is used in the proposed classification model to give a weight to each tree. Tea Sickness Dataset (TSD) has been used as the basis for evaluating the suggested method's effectiveness. The findings show that the suggested approach has an average accuracy of 92.47% in identifying seven different forms of tea leaf illnesses. Additionally, the recall and accuracy metrics indicate results of 92.35 and 92.26, respectively, indicating improvements over earlier techniques.

1. Introduction

Tea is an ancient and popular plant with a history of thousands of years [1]. It was first cultivated in China and currently, tea constitutes a significant share of agricultural exports in countries like China and India [2]. Over time, tea cultivation methods have undergone various changes, and efforts have been made to maximize tea yield by employing modern agricultural techniques. The

* Corresponding author.
** Corresponding author.
E-mail addresses: mmccarty72712@student.napavalley.edu (Q. Heng), yandongzhang2020@outlook.com (Y. Zhang).

https://doi.org/10.1016/j.heliyon.2024.e26465
Received 14 June 2023; Received in revised form 9 February 2024; Accepted 14 February 2024
Available online 18 February 2024

presence of tea leaf diseases is one of the main factors that can negatively affect the quality of the final product and yield [3]. Therefore, rapid and accurate detection of tea leaf diseases is of great importance. Currently, the main operational methods used for detecting tea leaf diseases are microscopic diagnosis and biological diagnosis. Microscopic methods, which involve subjective diagnosis approaches, require experienced specialists. However, this method has low accuracy, and even experts may make mistakes in disease identification [4]. On the other hand, although biological detection methods have higher accuracy, they require relatively expensive equipment and extensive workforce, making them economically impractical [5]. These conditions have led to the need for automated and cost-effective methods for accurate detection of tea leaf diseases.

Machine vision and artificial intelligence techniques are among the methods that can be effective in achieving this goal. By processing images of tea leaves, the presence of diseases in them can be automatically detected. Some study on this subject has been carried out in the last few years [6]. Only a few research in this area have shown how well artificial intelligence methods work to solve this issue. However, there are still challenges in achieving an ideal detection system. Constraints in the number of detectable diseases and low diagnostic accuracy are among the challenges [7] addressed in this research. The suggested challenges in this article are resolved by combining deep learning and ensemble learning approaches. With the use of a hybrid pooling mechanism, the pooling layers of the convolutional neural network (CNN) used in the suggested technique are modified in order to extract features from pictures of tea leaves. Additionally, a Weighted Random Forest (WRF) model is employed to improve the classification quality of the extracted features by the CNN. The structure of the proposed CNN model based on hybrid pooling for the purposes of extracting features related to tea leaf disease is one of the innovations of the current research. On the other hand, the use of a weighted combination of trees in the form of a random forest where the weight values of each tree are optimized using the cuckoo search optimization (CSO) is another innovative aspect of this paper that distinguishes it from previous works. The contribution of the current article can be summarized as follows:

- This article presents a CNN model based on hybrid pooling for extracting disease-related features from tea leaf images. The use of the hybrid pooling strategy in this CNN model enhances its efficiency in feature extraction from the images.
- To classify the characteristics that the CNN model extracted more accurately, a weighted random forest (WRF) is used in this paper. The decision tree models' outputs and corresponding weights are used in this model to identify illness. Each tree in the random forest is given a weight based on how well it performs. When compared to CNN alone, the weighted combination may improve the classification performance for feature classification. These aspects, including the innovative aspects of the current research, distinguish it from previous studies. The continuation of this article is structured as follows:
- Section 2: Related Works - This section provides an overview of previous research conducted in the field.
- Section 3: Proposed Method - In this section, the details of the proposed method are presented.
- Section 4: Results and Discussion - The performance of the proposed method is evaluated and discussed in this section.
- Section 5: Conclusion - This section summarizes the results and presents the conclusions drawn from the study.

2. Related Works

The review of research on the application of machine vision algorithms for tea leaf disease diagnosis is the main objective of this portion of the paper. The majority of approaches put out to tackle this issue make use of frameworks for classifying images. In these methods, the detection of tea leaf diseases involves preprocessing, feature extraction/selection, and classification steps. However, by employing deep learning techniques, the preprocessing and feature extraction steps can be implicitly performed by the deep model; which some of the recent approaches have been reviewed in this section.

Deep learning methods were used in Ref. [8] to identify illnesses in the tea leaf. This method uses a CNN model known as LeNET, which has three convolutional layers and then MaxPool layers. Two fully linked layers and a Softmax layer are included at the end of the model to help with feature categorization. The effectiveness of LeNET in identifying tea leaf illnesses was assessed in this research using only four disease categories. Reference [9] presents methods for tea leaf disease detection through image processing. In the first method, a CNN called LeafNet was utilized, comprising 5 convolutional layers and 3 MaxPool layers. The first two MaxPool layers are placed after the first and second convolutional layers, while the third MaxPool layer is placed after the fifth convolutional layer. Finally, three fully connected layers and a Softmax layer are used for detection. SVM and MLP models were used for illness categorization in the second and third strategies described in this study. Both of these techniques used the dense scale-invariant feature transform (DSIFT) to extract image features, which were then clustered and feature represented using Bag-of-Visual-Words (BOVW) for classification. The findings showed that when it came to identifying tea leaf illnesses, the CNN model performed noticeably better than the other two models.

A method for detecting tea leaf disease from photos that combines machine learning and deep learning approaches is provided in Ref. [10]. This approach involves first extracting the image's texture and color information, and then applying an SVM model to segment a suspected disease location. To supplement the training data, the technique makes use of the conditional deep convolutional generative adversarial network (C-DCGAN) model. Ultimately, a VGG-16 model processes each target area in the picture to identify the precise illness type. Nevertheless, only three disease kinds may be detected using this technology. The research conducted in Ref. [11] focuses on the detection of suspicious disease regions in tea leaf images. A simple clustering-based method using the Simple Linear Iterative Cluster (SLIC) algorithm and an SVM model is proposed. In this approach, pixel blocks are first extracted using the SLIC algorithm, and then salient points are identified using the Harris algorithm. Based on this, a fuzzy contour of the salient region is extracted using the convex hull method. Next, four-dimensional texture features are extracted from the salient and background regions, and a classification map of the regions is obtained using an SVM model.

Reference [12] introduces an improved CNN model for detecting tea leaf diseases. In this method, a multi-scale feature extraction component is added to the deep CNN model CIFAR10-quick to enhance the automatic extraction of features related to different tea leaf diseases. The model also utilizes a depth-wise separable convolution operator to reduce the number of model parameters and speed up computation.

A method for detecting tea leaf disease using image processing methods is described in Ref. [13]. Image segmentation and the identification of disease-containing areas are two applications for the Non-Dominated Sorting Genetic Algorithm (NSGA-II). The Principal Component Analysis (PCA) technique is used in the next phase to extract features from the target area. In order to determine the kind of illness in the photos, an SVM model is used to classify the retrieved characteristics. While the area identification stage performs well, the PCA method is unable to extract features effectively, which results in a reduction in detection accuracy. A CNN is suggested in Ref. [14] for the detection of tea leaf disease. Two successive convolutional and MaxPool layers make up the CNN model in this technique. Two fully connected layers and a SoftMax layer then classify the features that were retrieved from these layers. Machine learning methods for disease detection in plants are not limited to tea leaves alone. In Ref. [15], a method for detecting diseases in apple tree leaves using deep learning techniques is presented. This method utilizes DenseNet neural network architectures for detecting leaf-related diseases in apples. The architecture consists of four dense convolutional blocks with a total of 121 layers. In this study, the architecture is evaluated for regression and classification tasks related to disease types, and the model performs similarly well in both cases. Additionally, in Ref. [16], a method for detecting tomato-related diseases through image processing is proposed. This method addresses the issue of limited training samples by using a DCGAN network to generate artificial samples based on real samples. The combined set of real and artificial samples is then used as input for the GoogLeNet model to perform disease detection.

In [17], a method for detecting leaf lesions in images is presented, which utilizes a new fuzzy-based segmentation technique for identifying the target region. In this method, segmented images are first obtained, and then a set of descriptor features describing the target region based on texture, color, histogram, and lesion area properties are extracted. Nine different classifiers are used in this study for lesion detection in the target region, and the results demonstrate higher accuracy compared to other learning techniques. However, the proposed method has limitations in image processing and is not applicable to images captured in natural environments.

Furthermore [18], introduces an improved CNN-based method called DICNN for disease detection in grape leaves. In this method, input images are enhanced using brightness correction, contrast correction, rotation, and noise removal operators. The DICNN model consists of three components: a pre-network component comprising two convolutional layers and MaxPool, followed by an inception layer for more efficient feature extraction. The feature extraction component in DICNN consists of four consecutive inception layers with the same dimensions. At the end of the DICNN model, a classification component is included, which identifies seven types of diseases related to grape leaves using a Softmax layer.

Lastly [19], employs a combination of deep learning models and SVM for detecting rice-related diseases. In this method, the input image features are extracted using a ResNet50 model, and these features are then classified using an SVM model with a linear kernel function. This approach is capable of classifying four types of rice diseases. In general, the applications of deep learning are not limited to the diagnosis of diseases in agricultural products, and much wider applications can be imagined for these models. Applications such as packaging products [20], counting agricultural products [21], detecting defects [22], producing artificial images [23] are examples of these applications that can be effective in agricultural intelligence.

3. Proposed method

This section, a new method for diagnosing tea leaf disease through processing color image is presented. The proposed method for accurate diagnosis of tea leaf disease through image processing involves appropriate preprocessing of images, extraction of suitable features, and efficient classification of these features. The proposed method includes the following main steps:

1. Preprocessing
2. Feature extraction using a hybrid pooling-based CNN
3. Detection using WRF

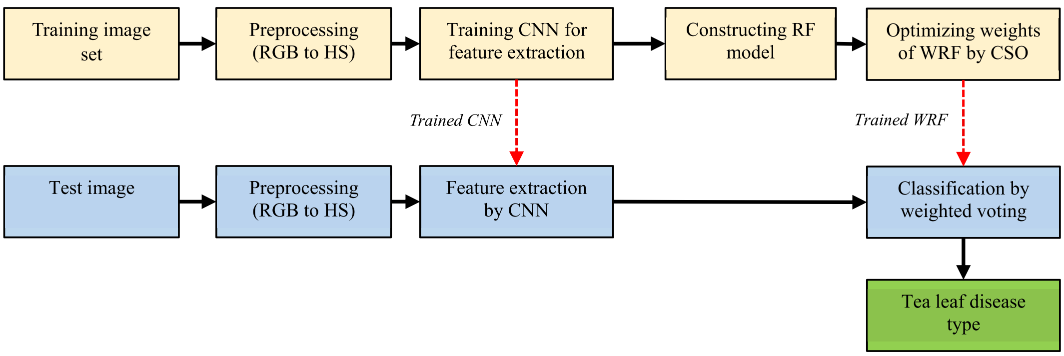

Fig. 1. Steps of the proposed method.

The stages of the suggested strategy are diagrammed in Fig. 1. The diagram's top and bottom parts, respectively, show the essential processes for each training and testing phase. The supplied photos first go through preprocessing. This step's objectives are to eliminate unnecessary details and arrange the photos to optimize the detection process's effectiveness. The characteristics of the pictures are extracted using a hybrid pooling-based CNN in the second stage. The pooling layers in this CNN model are randomly modified according to the average or maximum pooling functions. The last layer of this CNN model is a fully linked layer that depicts the characteristics taken out of the input picture; it is devoid of the layers required for classification. Lastly, a WRF ensemble model is used to classify the features that were retrieved using this CNN model. To reduce the training error in this classification model, a weight value is given using the CSO algorithm to every tree in the random forest. A weighted voting technique among the tree outputs is used in the identification phase of tea leaf disease. Below is a description of each step in the suggested procedure.

3.1. Preprocessing images

One of the most crucial stages in developing a precise system for identifying illnesses of tea leaves is preprocessing. As a result, the suggested method starts with preprocessing pictures of tea leaves. Eliminating superfluous information from the picture that might cause mistakes during the detection process is the aim of the preprocessing stage. On the other hand, additional processes should be avoided in the preprocessing step, as they increase computational costs and may result in the loss of image details. Intensity is one of the redundant characteristics that is not related to the type of tea leaf disease in the images. Considering it may increase the detection error. Additionally, the images in the database are recorded in the Red-Green-Blue (RGB) color system, where the intensity is described through the assigned values to the color layers of each pixel. Thus, the color scheme of the photos is first changed from RGB to Hue-Saturation-Intensity (HIS), where the intensity is expressed in a distinct layer, in the preprocessing stage of the suggested technique. Subsequently, the HSI pictures are characterized in two layers: HS after the intensity information is removed. The final pictures, which are represented by the HS color system, are sent into the second stage of the suggested procedure.

3.2. Feature extraction using a hybrid pooling-based CNN

The suggested method's second stage involves extracting picture characteristics using a CNN model that uses hybrid pooling. CNN models are popular deep learning methods for resolving a variety of issues. Applications like feature extraction, classification, regression, and clustering may benefit from the use of CNN-based models. However, CNN models require a large training dataset for learning patterns. If the number of training samples is small, large-scale CNN models may encounter overfitting issues [24]. This problem leads to a decrease in the generalization of the trained CNN model for new data. In most cases, CNN overfitting arises from the pooling functions used in the pooling layers, which allow obtaining feature maps based on the activation outputs of the convolutional layers [25]. Currently, two pooling functions are commonly used in CNN architectures. The first one is max pooling, which extracts the stronger activations of each region and eliminates background regions. This function can cause overfitting in practical applications [26]. The second function is average pooling, which considers the activations equally for each region. When combined with the ReLU activation function, it reduces the impact of strong activations and leads to the presence of numerous zero values in the feature map. On the other hand, its combination with other activation functions such as hyperbolic tangent neglects both positive and negative activations, resulting in information loss [26]. Considering these limitations, various research efforts have been made to control overfitting in CNN-based models by introducing new pooling strategies. Hybrid pooling [27] is one of the effective approaches that can enhance the generalization of CNN models.

The proposed method utilizes a hybrid pooling solution presented in Ref. [27] to improve the performance of the CNN model. In this approach, an attempt has been made to increase the generalization of the CNN model by training the network under different pooling techniques and averaging them during the testing phase. To achieve this, during the training phase, for each convolutional feature map, the average pooling function with probability p and the max pooling function with probability $1-p$ are used for all pooling regions. Therefore, the pooling process for a convolutional feature map can be formulated by equation (1) [27]:

$$S = \begin{cases} S_{avg} & \text{with probability } p \\ S_{max} & \text{with probability } 1-p \end{cases} \tag{1}$$

In the above equation, S_{avg} represents the result of average pooling for different regions and is described as the $S_{avg} = \{s^1_{avg}, \dots, s^J_{avg}\}$. Each element in this set, can be calculated using equation (2):

$$s^j_{avg} = \frac{1}{|R_j|} \sum_{i \in R_j} a_i \tag{2}$$

In the above equation, R_j represents the j-th pooling region, which contains a set of activation values denoted as $\{a_1, \dots a_{|R_j|}\}$. On the other hand, in equation (1), S_{max} represents the result of max pooling for different regions and is described as the set $S_{max} = \{s^1_{max}, \dots, s^J_{max}\}$. Each element in this set, can be calculated using equation (3):

$$s^j_{max} = \max_{i \in j} a_i \tag{3}$$

During the testing phase, the output of each pooling region is calculated by equation (4) as the average of the following two values [27]:

$$S = S_{hybrid} = p \times S_{avg} + (1 - p) \times S_{max} \tag{4}$$

This approach combines two pooling methods randomly for different feature maps to create diverse CNN models that consider the role of each convolution layer and characteristics of the processed data. Based on the described mechanism, the proposed CNN model will have a structure as shown in Fig. 2. The input of the proposed CNN model is provided through preprocessed HS images. The dimensions of each image are resized to 150 × 150 pixels, resulting in input samples with dimensions of 150 × 150 × 2. This CNN model consists of two convolution layers with 64 filters, and the dimensions of these two layers are set as 7 × 7 and 5 × 5, respectively. The parameters of these layers are adjusted empirically to prevent feature loss (in case of using fewer filters) or overfitting (in case of using more filters). The output of each convolution layer is activated using the ReLU layer, and feature maps are extracted from each region using hybrid pooling. After feature map extraction in the second convolution block, three consecutive fully connected layers are used for feature extraction. Gradual feature reduction in this model prevents the loss of useful features. Finally, the 100 features obtained from the last fully connected layer of the proposed CNN model are considered as the extracted features for classification. This vector is used as the input to the classification model.

3.3. *Classification by weighted random forest (WRF)*

The CNN model's retrieved features are then categorized using a WRF model as the last stage of the suggested procedure. With weights assigned to each tree component, the WRF model functions as a kind of weighted random forest. A weighted voting approach is used to decide the model's output. The CSO method is used in this model to determine the tree weights in the random forest. By optimizing the tree weights, this approach aims to reduce the WRF's training error. Thus, the WRF-based detection process consists of three partial steps:

1. Constructing tree models based on data partitioning (bootstrap).
2. Weighting tree models based on CSO.
3. Detection based on weighted voting (aggregation).

In the proposed method, the first two steps are performed during the training phase, while the third step is used during the testing phase to determine the detection results for new samples of tea leaf diseases.

3.3.1. *Constructing tree models based on data partitioning*

The first step in constructing the WRF model is to build multiple tree models based on various data partitions. The proposed WRF model uses a set of CART trees [28] as its learning components. Initially, the extracted feature set is divided into N subsets, where each subset randomly selects a number of features. Additionally, each data subset formed in this step includes all training samples. The minimum number of selected features in each subset is 3. Then, based on each data subset, a CART model is constructed. The result of this step is N CART models, each formed based on one of the data subsets. The construction algorithm of the CART tree is described in detail in Ref. [28], so it is not discussed in this section.

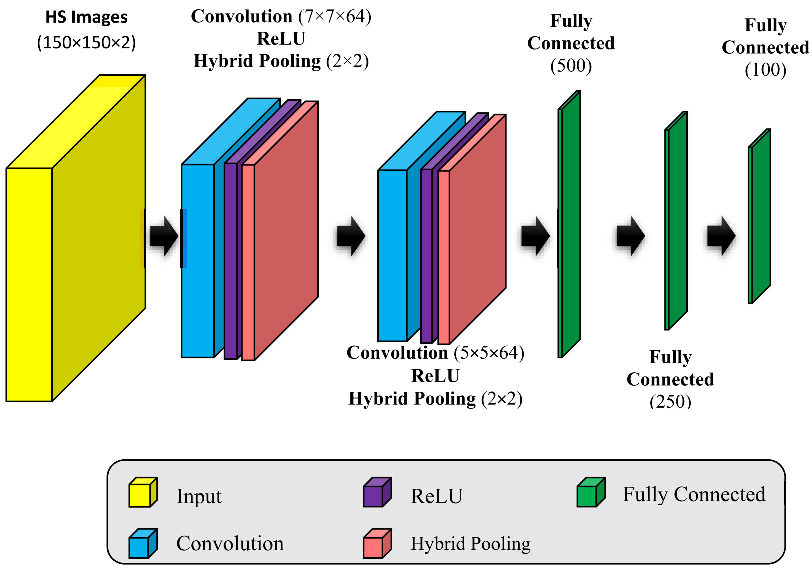

Fig. 2. Proposed CNN model for feature extraction.

3.3.2. Weighting tree models based on CSO

The CSO method is used to weight the CART models produced in the preceding phase of the WRF development process. In order to optimize detection accuracy, this stage aims to ascertain how each CART model affects the final detection outcome. The final output may be significantly impacted by giving a weight to each CART model in order to optimize its influence, since each model varies in performance and produces varied detection results. The suggested solution uses the CSO algorithm, which tries to increase the weights of effective CART models and decrease the weights of models with larger mistakes, hence improving the detection system's effectiveness. The structure of the solution vector and the fitness function are first shown in the following, followed by an explanation of the optimization procedures for utilizing the CSO method to weight the CART model components. The ideal weights for the CART components in the WRF model are the optimization issue that is covered in this stage of the suggested strategy. Every WRF weight, which corresponds to an integer value between 0 and 6, should be allocated to every CART component. With this description, if the WRF model includes N CART components, then each solution vector in the optimization problem will have a length of N, where each element of this vector corresponds to the weight assigned to one of the CART components and can take one of the values {0, 1, 2, 3, 4, 5, 6}. In a solution vector such as $sol = \{w_1,\ldots,w_N\}$, the value of w_i represents the weight assigned to CART component i and indicates the number of times the output of this component is considered in the voting phase. For example, $w_i = 0$ indicates that the output of CART component i is ignored in the voting phase, while $w_i = 3$ indicates that the output of CART component i is calculated three times in the voting phase.

The training error criteria is used to each solution vector in order to assess its fitness. All CART components are subjected to the provided weight values in the solution vector in order to achieve this. Next, for each training sample, the output of each CART component is computed. In this process, the output of each component is repeated according to the assigned weight value, and finally, the detection result for each sample is determined by voting among the repeated outputs of all components. By comparing the actual labels of the training samples with the calculated outputs through the weighted voting process, the fitness of the solution vector can be calculated as equation (5):

$$Fitness = \frac{F}{T} \tag{5}$$

The training samples with labels that vary from their real labels are denoted by F in the equation above, and the number of training samples is indicated by T. We go on to outline the procedures for optimizing weight values using the CSO algorithm based on the solution vector and fitness function structures that have been discussed. The CSO algorithm is a meta-heuristic algorithm inspired by the natural behavior of a bird species for laying eggs in other birds' nests [29]. The optimization steps for weighting CART components of the proposed WRF model are as follows:

Step 1. Creating an initial population of n nests, each containing k eggs. This population is randomly generated within the specified bounds for weight values [0, 6].

Step 2. If the fitness has reached zero or the number of algorithm iterations equals the threshold T, proceed to step 3; otherwise, repeat the following steps:

Step 2-1. Select a solution (cuckoo) like x randomly based on the Levy flight algorithm.

Step 2-2. Calculate the fitness of the solution using equation (5).

Step 2-3. Select a random nest like y from the population n.

Step 2-4. If the fitness of solutions in y dominate x, replace x with the solutions available in y.

Step 2-5. Remove the worst P_a ratio from the worst nests and replace them with the new random nests.

Step 2-6. Transfer the set of best nests (solutions) to the next iteration.

Step 2-7. Increment the iteration counter by one unit and repeat step 2.

Step 3. Return the solution with the least fitness as the optimal solution.

The result of executing the above steps will be an optimal weight vector for the learning components, based on which voting and weighting can be performed in the proposed WRF model.

3.3.3. Detection based on weighted voting

After determining the optimal weight values for CART components in the proposed WRF model, this structure can be used to diagnose tea leaf diseases in new test samples. To achieve this, each new test sample is simultaneously processed by all CART models (those with weights greater than zero) to obtain the labels determined by all components. Then, the output of each CART model is repeated according to its assigned weight value, and the final output of the WRF model is determined by majority voting among the collected labels. If a test sample belongs to two or more target classes with the same number of votes, the sample will be assigned to the class which is determined by CART component with lowest training error.

4. Implementation and evaluation

The outcomes of using the suggested strategy are shown and discussed in this section. To do this, the database and assessment standards have been provided first, followed by a discussion of the experiment findings. Software called MATLAB 2019a was used to accomplish the suggested approach. The evaluation dataset was the Tea Sickness Dataset (TSD) [26]. Accuracy, precision, recall, and F-measure metrics were utilized in the trials to assess the suggested method's efficacy in detecting tea leaf illness.

4.1. Database and evaluation metrics

The suggested approach was put into practice using the TSD database [30]. There are 885 color photos in this collection, with each sample representing one of the seven major illnesses seen in tea leaves. The recorded diseases in this database are as follows: 1- algal leaf (113 samples), 2- Anthracnose (100 samples), 3- bird eyespot (100 samples), 4- brown blight (113 samples), 5- gray blight (100 samples), 6- red leaf spot (143 samples), and 7- white spot (142 samples). Additionally, there are 74 samples containing images of healthy tea leaves in this database, which were disregarded due to the low number of samples and the lack of disease description. The dimensions, aspect ratios of each image, and the orientation of the target region (tea leaf) vary among the samples in this database. During the experiments, all images were converted to the HS color space with dimensions of 150 × 150 pixels. Fig. 3 shows some samples from the TSD database.

The suggested solution was implemented using a 10-fold cross-validation (CV) methodology. Accuracy, precision, recall, and F-measure were computed by comparing the real labels of the test samples with the projected labels generated by the proposed method for each fold. Since accuracy and recall are used for evaluating performance in binary classification problems, these metrics were calculated separately for each class, considering the current class as positive and other classes as negative. Precision which is calculated using equation (6), represents the algorithm's accuracy in classifying samples of each class separately. On the other hand, recall indicates the proportion of positive samples which were correctly classified and is calculated by equation (7). The F-measure is a harmonic mean of precision and recall and can be calculated based on equation (8):

$$Precision = 100 \times \frac{TP}{TP + FP} \tag{6}$$

$$Recall = 100 \times \frac{TP}{TP + FN} \tag{7}$$

$$F - Measure = 2 \times \frac{Precision \times Recall}{Precison + Recall} \tag{8}$$

The number of properly recognized positive samples is denoted by FP in the equations above. The number of positive samples that are inadvertently placed in other (negative) categories is indicated by the letter FN. Furthermore, FP shows how many negative samples are incorrectly identified as positive.

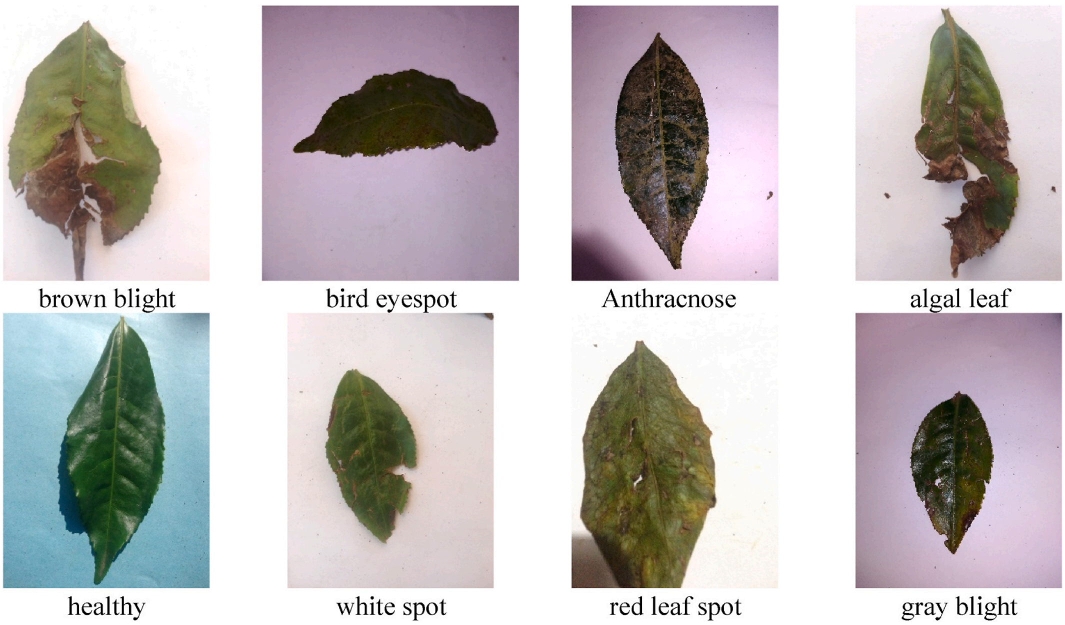

Fig. 3. Samples from the TSD database belonging to different categories.

4.2. Results and discussion

Software called MATLAB 2019a was used to accomplish the suggested approach. Every experiment was conducted using an Intel Core i7 CPU with a 3.2 GHz clock frequency and 16 GB RAM on a personal computer running Windows 10. Based on the Nvidia GeForce GTX 1080 graphics adapter's CUDA capabilities, the CNN model was implemented. To examine the effectiveness of each technique employed in the proposed method (HybPool + WRF), its performance was compared with the following configurations:

- Proposed (MaxPool + WRF): In this configuration, all pooling layers in the CNN model used in the proposed method were of the max pooling type.
- Proposed (AvgPool + WRF): In this configuration, all pooling layers in the CNN model used in the proposed method were of the average pooling type.
- Proposed (HybPool + RF): In this configuration, feature classification was performed by a basic random forest model.

The suggested technique's performance was compared not only with the three configurations mentioned above, but also with the LeafNet model in Ref. [9] and the Mukhopadhyay et al. method in Ref. [13]. During the implementation, the number of CART classifiers in the WRF model was set to 100. The population size and number of iterations parameters in the CSO algorithm for weight vector tuning of the WRF model were set to 170 and 300, respectively. The probability of nest abandonment in the optimization algorithm was set to $P_a = \frac{1}{4}$.

The variations in the CSO algorithm's fitness for weight vector optimization of the WRF model are seen in Fig. 4. This chart shows the average fitness of the population throughout the optimization search in addition to the differences in the best fitness.

After 236 iterations, the CSO algorithm can identify the ideal weight vector for CART components in the suggested WRF model, according to an inspection of the presented graph in Fig. 4. In addition, the graph's mean fitness displays a downward trend, suggesting that the varied population is always heading in the direction of the global optimum. Fig. 5 compares the suggested method's average accuracy with other approaches for recognizing different tea leaf illnesses in the TSD database. Please take notice that the data shown in this figure and the other graphs in this part represent the total results of ten experiment repeats. Based on Fig. 5, the proposed method can achieve an accuracy of 92.47% in identifying tea leaf diseases in the database samples, and it outperforms the compared methods. On the other hand, if the proposed WRF model is replaced with a basic random forest, the detection accuracy will decrease to 89.21%. Similarly, if the hybrid pooling strategy is ignored and feature extraction is done by the Max Pooling-based CNN model, the detection of tea leaf diseases can be performed with an accuracy of 90.12%. In contrast, after replacing the hybrid pooling strategy with Average Pooling, the detection accuracy decreases to 89.01%. These comparisons indicate that: Firstly, the use of the proposed WRF model can perform better compared to the basic RF model and increase the detection accuracy by at least 3.08%. Secondly, comparing the performance of hybrid pooling mechanism with CNN models using static pooling functions demonstrates that this mechanism is capable of extracting more accurate disease-related features, and based on the extracted features, the detection accuracy can be increased by at least 2.35%. These findings show how well the methods included in the suggested strategy work to increase the precision of tea leaf disease identification.

The LeafNet model, on the other hand, achieves an accuracy of 90.12%, which is the closest technique to the suggested method in terms of accuracy. The confusion matrix may give more thorough insights into classification algorithms' efficacy in recognizing different tea leaf illnesses. Fig. 6 depicts the confusion matrix of the proposed technique and competing methods for identifying database samples. In these confusion matrices, each column represents the actual labels of the test samples, and the rows represent the labels assigned by each classification method. For example, in Fig. 6(a), out of 113 samples belonging to the "algal leaf" class (the sum of values in the first column of the matrix), the proposed method correctly classified 101 samples, and only 12 samples were

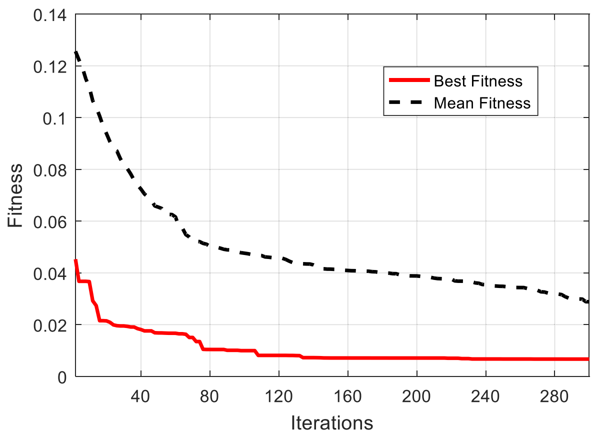

Fig. 4. Variations in the mean and best fitness discovered for weight vector tuning of the learning components in the WRF model by the CSO algorithm.

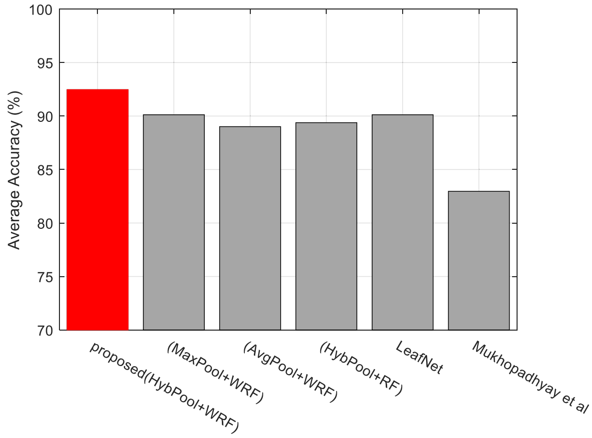

Fig. 5. Average accuracy of the proposed method and other methods in detecting tea leaf diseases in the TSD database.

misclassified into other categories. The LeafNet method showed similar performance in classifying samples of this class. The interpretation of classification results for other existing categories is similarly possible. Overall, the comparison of these confusion matrices shows that the proposed method has superiority over other methods in classifying samples of most categories and has managed to increase the accuracy of disease detection by at least 2.35%. Fig. 6(b) and (c), represent the results of combining WRF with CNNs using max pooling and average pooling layers, respectively. These results show that utilizing hybrid pooling layers in CNN model can improve accuracy at least 2.35%. On the other hand, Fig. 6(d) shows the confusion matrix resulting from replacing WRF with conventional RF in proposed method. This figure, proves the superiority of WRF in classifying instances. Also, using Fig. 6(e) and (f), the proposed method can be compared with models presented in Refs. [9,13], respectively.

Fig. 7(a) compares the performance of different methods in detecting various tea leaf diseases based on precision criteria in the TSD database. Furthermore, Fig. 7(b) and (c) provide a comparative analysis of the performance of these methods in terms of recall and F-measure, respectively.

The first dimension in each of the presented graphs in Fig. 7 indicates the classes linked to tea leaf illnesses, while the second dimension relates to the contrasted procedures. By viewing these graphs, it is clear that the suggested approach, when combined with hybrid pooling and WRF, can categorize distinct categories more efficiently than previous methods. Fig. 8 depicts the average accuracy, recall, and F-measure metrics. These graphs show how various techniques fare in terms of categorization quality overall. Table 1 also displays the numerical data gained from the tests done in this section.

As seen by Table 1 and Fig. 8's comparison of accuracy, precision, recall, and F-measure metrics, the suggested technique is superior to the other approaches in its ability to detect different tea leaf illnesses. These findings suggest that the suggested approach may raise the accuracy, recall, and F-measure metrics. Greater accuracy confirms that, in comparison to other approaches, the outputs produced by the suggested technique have a higher likelihood of being accurate for each kind of illness. Furthermore, increased recall shows that a greater percentage of samples accurately identified as belonging to distinct illnesses using the suggested strategy. The ROC curve derived from the categorization of different tea leaf diseases in the TSD database is shown in Fig. 9. This graph shows that the suggested technique outperforms the other methods in terms of area under the ROC curve, true positive rates, and false positive rates. Therefore, based on the photos provided, it can be said that the approach suggested in this research has a greater average accuracy in properly diagnosing different types of tea leaf illnesses.

The studies in this part demonstrated that the suggested technique outperforms the comparative works in terms of accuracy in identifying tea leaf illnesses. This approach may be more successful in detecting each kind of illness than prior studies, demonstrating the efficacy of the strategies provided in this study. However, compared to the LeafNet [9], the proposed method has some limitations in terms of model training time. Using two separate CNN and WRF models for feature extraction and classification, although it is effective in increasing the accuracy by at least 2.35%, but it also increases the training time of the recognition model by at least 3.11%, which is due to the step of optimizing the weights of the WRF model. However, it should be highlighted that this processing time is only evident during the training phase and has no effect on the suggested method's speed during the diagnostic (test) phase.

5. Conclusion

Tea leaf disease detection that is automatic and accurate may help to improve product quality and production efficiency. Using artificial intelligence approaches, a novel method for increasing the accuracy of tea leaf disease detection was presented in this research. A hybrid pooling-based CNN was used in the suggested technique to extract disease-related characteristics. In this approach, the pooling layers of the CNN model were randomly adjusted using either max pooling or average pooling functions. Performance comparison of this hybrid model with CNN models using static pooling functions demonstrated that this mechanism can extract more accurate disease-related features, resulting in a minimum 2.35% increase in diagnostic accuracy. Additionally, the proposed method utilized a new WRF model for classifying the extracted features. This model employed the CSO algorithm to determine optimal weight

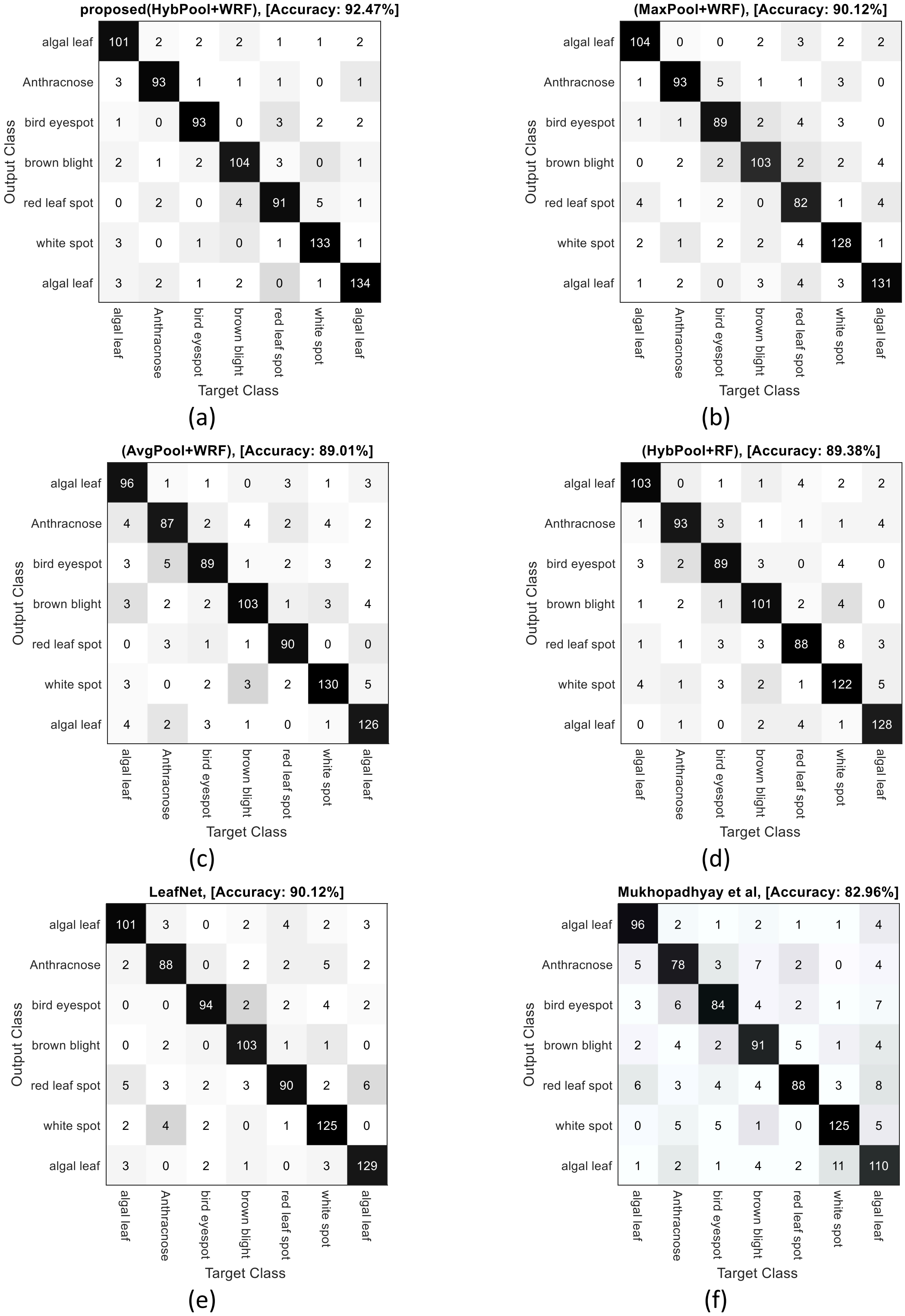

Fig. 6. Confusion matrices of (a) proposed method, (b) combination of CNN with max pooling layers and WRF, (c) combination of CNN with average pooling layers and WRF, (d) combination of CNN with hybrid pooling layers and RF, (e) LeafNet [9] and (f) Mukhopadhyay et al. [13].

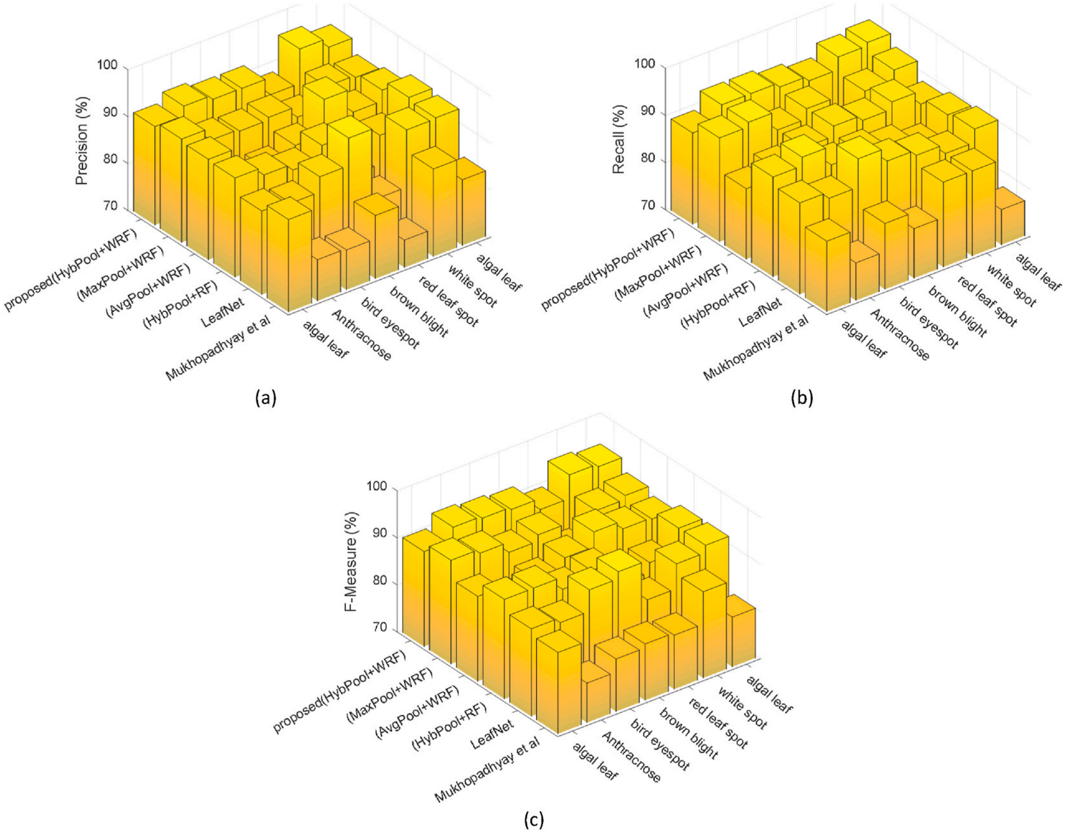

Fig. 7. Comparison of the performance of different methods based on (a) precision, (b) recall, and (c) F-measure.

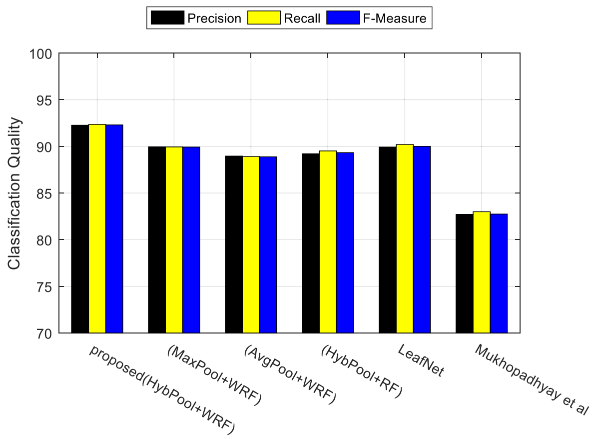

Fig. 8. Average values of precision, recall, and F-measure.

values for each tree in the forest and adjusted the weight vectors of the trees to minimize training errors. Performance comparison of this model with the basic random forest algorithm showed that the proposed classification model can increase the accuracy of tea leaf disease diagnosis by 3.08%. The suggested method's efficacy was assessed using the TSD dataset, and the findings were analyzed. According to the study results, the suggested approach properly classified 7 kinds of tea leaf illnesses with an average accuracy of 92.47%. Furthermore, the suggested method's precision and recall metrics for tea leaf disease diagnosis were 92.26% and 92.35%, respectively, suggesting its superiority over previous approaches. Existing databases do not cover a considerable number of tea leaf disease types. Therefore, one of the obstacles in achieving an operational automatic tea leaf disease diagnosis system is the limited number of training samples for different disease types, which has affected this research and similar studies. Hence, in future research, this limitation can be addressed by adding samples of new disease types to the existing databases. It indicates that the suggested

Table 1
Comparison of the performance of the proposed method with other methods.

Method	Accuracy	F-measure	Recall	Precision
Proposed (HybPool + WRF)	92.4691	92.3005	92.3492	92.2636
MaxPool + WRF	90.1235	89.9268	89.9400	89.9512
AvgPool + WRF	89.0123	88.8830	88.9126	88.9569
HybPool + RF	89.3827	89.3309	89.5125	89.2071
LeafNet [9]	90.1235	89.9975	90.2006	89.9206
Mukhopadhyay et al [13]	82.9630	82.7447	82.9971	82.7118

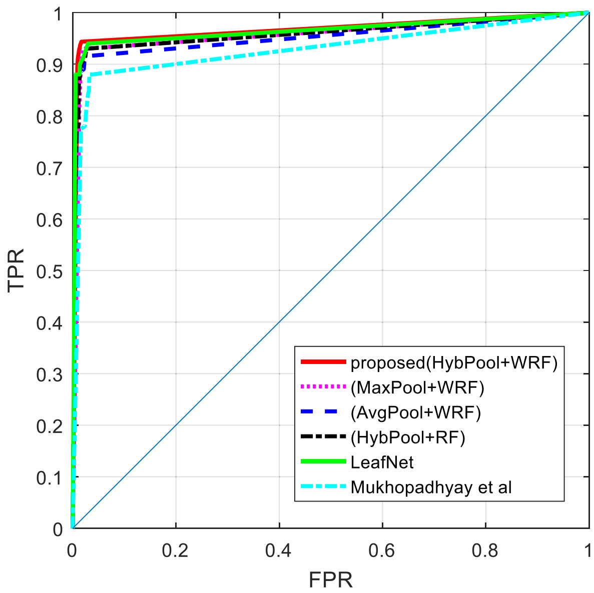

Fig. 9. ROC curve resulting from the classification of various tea leaf diseases in the TSD database.

method's use is not restricted to the identification of tea leaf problems. As a result, in future research, the efficiency of the suggested approach in identifying illnesses associated with different agricultural products may be studied.

Data availability

All data generated or analyzed during this study are included in this published article.

CRediT authorship contribution statement

Qidong Heng: Software, Resources, Investigation, Formal analysis, Data curation. **Sibo Yu:** Writing – review & editing, Visualization, Software, Investigation, Formal analysis, Data curation, Conceptualization. **Yandong Zhang:** Project administration, Investigation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] G. Yashodha, D. Shalini, An integrated approach for predicting and broadcasting tea leaf disease at early stage using IoT with machine learning–a review, Mater. Today: Proc. 37 (2021) 484–488.
[2] B.C. Karmokar, M.S. Ullah, M.K. Siddiquee, K.M.R. Alam, Tea leaf diseases recognition using neural network ensemble, Int. J. Comput. Appl. 114 (17) (2015).
[3] M.J.A. Soeb, M.F. Jubayer, T.A. Tarin, M.R. Al Mamun, F.M. Ruhad, A. Parven, I.M. Meftaul, Tea leaf disease detection and identification based on YOLOv7 (YOLO-T), Sci. Rep. 13 (1) (2023) 6078.
[4] X. Hao, W. Zhang, F. Zhao, Y. Liu, W. Qian, Y. Wang, X. Wang, Discovery of plant viruses from tea plant by metagenomic sequencing, Front. Microbiol. 9 (2018) 2175.
[5] M.S. Segal, A. Bihorac, M. Koç, Circulating endothelial cells: tea leaves for renal disease, Am. J. Physiol. Ren. Physiol. 283 (1) (2002) 11–19.

[6] L. Li, S. Zhang, B. Wang, Plant disease detection and classification by deep learning—a review, IEEE Access 9 (2021) 56683–56698.
[7] V.S. Dhaka, S.V. Meena, G. Rani, D. Sinwar, M.F. Ijaz, M. Woźniak, A survey of deep convolutional neural networks applied for prediction of plant leaf diseases, Sensors 21 (14) (2021) 4749.
[8] S. Gayathri, D.J.W. Wise, P.B. Shamini, N. Muthukumaran, Image analysis and detection of tea leaf disease using deep learning, in: IEEE 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), 2020, July, pp. 398–403.
[9] J. Chen, Q. Liu, L. Gao, Visual tea leaf disease recognition using a convolutional neural network model, Symmetry 11 (3) (2019) 343.
[10] G. Hu, H. Wu, Y. Zhang, M. Wan, A low shot learning method for tea leaf's disease identification, Comput. Electron. Agric. 163 (2019) 104852.
[11] Y. Sun, Z. Jiang, L. Zhang, W. Dong, Y. Rao, SLIC_SVM based leaf diseases saliency map extraction of tea plant, Comput. Electron. Agric. 157 (2019) 102–109.
[12] G. Hu, X. Yang, Y. Zhang, M. Wan, Identification of tea leaf diseases by using an improved deep convolutional neural network, Sustain. Comput.: Inform. Syst. 24 (2019) 100353.
[13] S. Mukhopadhyay, M. Paul, R. Pal, D. De, Tea leaf disease detection using multi-objective image segmentation, Multimed. Tool. Appl. 80 (2021) 753–771.
[14] X. Sun, S. Mu, Y. Xu, Z. Cao, T. Su, Image Recognition of Tea Leaf Diseases Based on Convolutional Neural Network, 2019 arXiv preprint arXiv:1901.02694.
[15] Y. Zhong, M. Zhao, Research on deep learning in apple leaf disease recognition, Comput. Electron. Agric. 168 (2020) 105146.
[16] Q. Wu, Y. Chen, J. Meng, DCGAN-based data augmentation for tomato leaf disease identification, IEEE Access 8 (2020) 98716–98728.
[17] G. Dhingra, V. Kumar, H.D. Joshi, A novel computer vision based neutrosophic approach for leaf disease identification and classification, Measurement 135 (2019) 782–794.
[18] B. Liu, Z. Ding, L. Tian, D. He, S. Li, H. Wang, Grape leaf disease identification using improved deep convolutional neural networks, Front. Plant Sci. 11 (2020) 1082.
[19] P.K. Sethy, N.K. Barpanda, A.K. Rath, S.K. Behera, Deep feature based rice leaf disease identification using support vector machine, Comput. Electron. Agric. 175 (2020) 105527.
[20] R. Zhang, X. Chen, Z. Wan, M. Wang, X. Xiao, Deep learning-based oyster packaging system, Appl. Sci. 13 (24) (2023), https://doi.org/10.3390/app132413105.
[21] H. Zheng, X. Fan, W. Bo, X. Yang, T. Tjahjadi, S. Jin, A multiscale point-supervised network for counting maize tassels in the wild, Plant Phenomics 5 (2023) 100, https://doi.org/10.34133/plantphenomics.0100.
[22] Y. Tao, J. Shi, W. Guo, J. Zheng, Convolutional neural network based defect recognition model for phased array ultrasonic testing images of electrofusion joints, J. Pressure Vessel Technol. 145 (2) (2023), https://doi.org/10.1115/1.4056836.
[23] H. Liu, Y. Xu, F. Chen, Sketch2Photo: synthesizing photo-realistic images from sketches via global contexts, Eng. Appl. Artif. Intell. 117 (2023) 105608, https://doi.org/10.1016/j.engappai.2022.105608.
[24] P. Thanapol, K. Lavangnananda, P. Bouvry, F. Pinel, F. Leprévost, Reducing overfitting and improving generalization in training convolutional neural network (CNN) under limited sample sizes in image recognition, in: IEEE 2020-5th International Conference on Information Technology (InCIT), 2020, October, pp. 300–305.
[25] Y.L. Boureau, J. Ponce, Y. LeCun, A theoretical analysis of feature pooling in visual recognition, in: Proceedings of the 27th International Conference on Machine Learning, ICML-10, 2010, pp. 111–118.
[26] M.D. Zeiler, R. Fergus, Stochastic Pooling for Regularization of Deep Convolutional Neural Networks, 2013 arXiv preprint arXiv:1301.3557.
[27] Z. Tong, G. Tanaka, Hybrid pooling for enhancement of generalization ability in deep convolutional neural networks, Neurocomputing 333 (2019) 76–85.
[28] W.Y. Loh, Classification and regression trees, Wiley Interdiscip. Rev.: Data Min. Knowl. Discov. 1 (1) (2011) 14–23.
[29] X.S. Yang, S. Deb, Cuckoo search: recent advances and applications, Neural Comput. Appl. 24 (2014) 169–174.
[30] Gibson Kimutai, Anna Förster, Tea sickness dataset, Mendeley Data, 2022, https://doi.org/10.17632/j32xdt2ff5.2. V2.