An optimal automated disease detection and classification of crop species using hybrid machine learning techniques

Shravankumar Arjunagi
Dept of cse, Appa Institute of Engineering and Technology, Kalaburgi

Dr Nagaraj B Patil
Government college of Engineering, Gangavati

Abstract - Agriculture is dependent on the quality and quantity of plant growth worldwide. Many factors, such as weather, pest attack or disease, can cause plant-specific infections. The process of voluntary diagnosis is a long and laborious process, and the farmer cannot determine the cause and effect of the disease. This study proposes the optimal autocorrelation and classification of crop species (OADD-CS) using hybrid machine learning techniques. The first contribution is to propose a non-linear deep neural network (NL-DNN) for processing before removing abnormal image angles from the input image. Next, the edge target detection (ETD) algorithm is used to calculate the top and bottom leaf edges and extract the feature. Finally, hybrid crow-optimization-based convolutional neural network (HCO-CNN) classifiers are used to diagnose other diseases. Here we consider a dataset of over a million images taken with a mobile phone in real field conditions. This data set includes 17 diseases for 5 crops and 5 equally distributed diseases, with multiple diseases present in the same image. Specific work can be performed on MATLAB tools, and performance is compared to traditional complex work.

Keywords: Spiking Neural Networks, Shape Features, Color Features, Tooth Features, Gabor Features

1. Introduction

The agricultural environment is a food source in the world today. The Indian economy is based on agriculture and is a major source of rural livelihood. The Indian economy is heavily dependent on agricultural productivity [1] [2]. Therefore, in agriculture, diseases play an important role in plants. All living things depend on agriculture for food. However, for better yields, crops must be healthy, so routine monitoring requires advanced technology [3]. Plant disease is an important factor that significantly reduces the quality and quantity of agricultural products. Due to the exponential slope of the population, climate also contributes to plant diseases. Plants suffer from diseases that affect the quality of the crop. The identification and identification of leaf diseases is usually carried out by farmers using the naked eye. This leads to false diagnosis as the farmer solves the symptoms through his experience [4]. This makes unnecessary and excessive use of expensive pesticides. Therefore, it is important to automatically detect the disease, which helps in the early diagnosis of leaf disease. One of the major challenges of sustainable development is to improve quality by reducing pesticide use and reducing environmental advertising costs [5] [6].

Machine learning is becoming increasingly popular due to the technology and practical applications offered in many fields [7]. In identification / discovery, mechanical means are used to identify objects or certain essential parts of an image or object. For example, it identifies areas affected by plants or detects water bodies from satellite images. When measured, the machine learning algorithm is used to measure the amount of material found. Classification works by categorizing images or objects into different categories according to the characteristics and characteristics of the data set [8][9]. The final forecast is commonly used in research to predict future results based on current data sets and collected facts. However, size and computation are more important than classification and identification, and very little has been done in previous studies. Machine learning prediction systems can be useful for early outbreaks based on current habitat and crop characteristics. Many machine learning methods are used to determine plants, fruits and vegetables. These include logistic regression (LR), naïve bayes (NP), support vector machine (SVM), ensemble technologies, clustering algorithms, KN (K-Neighbor) and EDC [10].

Various diagnostic and classification methods have recently been used for the classification and diagnosis of leaf disease. The K-adjacent Neighbor (K-NN) algorithm is integrated to classify plant leaves with two-dimensional shape characteristics, [11] called distribution hierarchical graphical neurons (DHGN). Euclidean Distance (ED) and machining measures (MM) classifiers [12] are used to identify leaves. This module was tested in dynamic conditions using YOUNG-leaf-detection efficiency (hYLD) and leaf discrimination efficiency (hLD). Multi-class SVM (K-SVM) [13] is used to create a good number. The classifier has 1 VS1 SVM, which can be used to determine class pairs. Classes from 1vs1 SVM are used in the classification process to determine all class pairs.

The final classification is based on the number of binary votes available in each class. To reject all leaf sizes, automatic processes of form, texture information or color are used. It is commonly used to detect changes in leaf morphology [14]. Mulberry leaves are used in Taiwan based on plant characteristics and cold phenotype classification system [15]. Phenotype classification systems require numerical classification analysis. Another method is based on leaf thickness [16]. In this way, the correlation between leaf length and leaf width is taken into account. Thick leaves are described as beneficial for increasing the area of a single leaf. In Brazil, it is used to identify high economic conditions. A black-sounding automated detection method is used to identify leaves [17]. Leaf proteins were isolated from leaves infected with Tarsica. This activity is performed after 72h of vaccination and is differentially expressed using two-dimensional electrophoresis and mass spectrometry [18]-[20].

2. Related works

Thenmozhi et al. [21] have proposed an efficient deep CNN model for classifying insects from three common insect data sets. We evaluated this model and compared it with pre-trained in-depth study frameworks for pest classification (e.g., Alexanet, Resnet, Coculinet, and VGNET). Transfer training was used to optimize the pre-training model. Data enrichment techniques such as mirroring, scaling, rotation, and transformation are used to prevent network overload. We analyzed the performance of superconductors in the proposed model to improve accuracy. Classification accuracy of 96.75, 97.47 and 95.97% was achieved in standard CNN models for NBIR insect database (40 species), X1 (24 species) insect database and X2 (40 species) insect database.

Rangarajan et al. [22] have proposed tomato crop disease classification using images from the Plant Village Dataset using previously trained deep learning frameworks, AlexaNet and VGG16NET. The classification accuracy used for the 13,262 images is 97.29% for the VGG16 net and 97.49% for the Alexa net. The performance of the model was evaluated by changing the number of images, various mini-patch size settings, weights, and relative learning rate. The number of images had a major impact on model performance. I get maximum accuracy when the number of images is 373. There is no clear correlation with classification accuracy to fine-tune the mini-patch size of the Alaskanet, but the accuracy of the mini-patch size in the VGG16 net decreases.

Picon et al. [23] have presented new development of automated multi-disease identification for field acquisition conditions. These improvements can help support early diagnosis while maintaining high features. The system is validated against three diseases: septoria (septoria dricti) in wheat film, tan spots (Drexlera tridenti-rebendis) and rust (Puccinia striiformis & Pucinia recondita), which have been tested and used in real smartphone applications. This means that the average PAC of the original specification was reduced to 0.78, especially in early patients, so the specification is reduced, which does not guarantee greater reliability in detection.

KC et al. [24] presented two versions of depth separation, which includes two types of building blocks. Samples were trained and tested on a subset of the publicly available Plant Village Dataset, 82,161 images of 55 healthy and diseased plants. These deep split coils achieved lower accuracy and higher gains in combination speed. A number of models have been trained and tested with 29% fewer parameters and 62.6% less classification accuracy than the mobile NET compared with the mobile NTG reduction. However, when testing the model in a set of images taken under different conditions than those used for training, MobileNet outperformed the existing model with an accuracy of 36.03%.

Ozguven et al. [25] have proposed updated R-CNN structure is improved by changing the fast R-CNN structure and the modified parameters of the CNN model for the automatic detection of sugar beet leaf spot disease. 155 images were trained to determine the severity of the disease by an imaging-based visualization system, and the overall correct classification rate was 95.48% according to the test results. This method shows that changing the CNN parameters depending on the image and location of the image will increase the success of the fast R-CNN architecture. This method yielded better results for the relevant parameters than the most recent method reported in previous literature. Therefore, we hope that this method will reduce the time it takes to determine beet spots in large productive areas and reduce human error and time to detect the severity and incidence of the disease.

Toseef et al. [26] have proposed an intelligent approach to crop disease diagnosis has been proposed, which is a key decision-making mechanism in the backend, allowing you to work on Android mobile devices using ambiguous inference methods. This system is sufficient for Pakistani farmers to communicate in their native language, Urdu and to diagnose crop diseases. Government experts in agriculture will benefit equally from the detection and prevention of crop diseases. Use virtual machines to provide crop symptoms with ambiguous inputs and to generate output in the form of diagnostic diseases. This system provides the major crops in Pakistan such as cotton and wheat and can detect major ailments. The system was tested in a pool of 100 real crop problems, and the imaging machine performed well in predicting 99% accuracy and accuracy.

Roldán-Serrato et al. [27] have presented automated insect detection systems using artificial neural networks. The system automatically detects two rotting insects from the Mexican bean beetle and the Colorado potato beetle from potato and soybean crops. RSC and LIRA are the neural classifiers used for beetle detection. The MBB image used as input to the classifier was obtained from a Mexican farm. CBP images were collected from various internet sources. Compare the results obtained with the two classifiers in the image database. The RSC classifier shows the best results for certification (89%), but Lira provides an 88% recognition rate. Its purpose is to contribute to the development of automated detection applications based on images of potato and soybean farms. Solving the pest problem in agriculture is very important in Mexico and other countries.

ArnalBarbedo et al. [28] have discussed the classification of plant diseases using digital images is difficult. Deep learning technology, especially CNN, can properly address most of the technical challenges associated with plant disease classification. On the other hand, the limitations of the data, which take into account the number and variety of models, preclude the emergence of a comprehensive system for classification of plant diseases. Attempts have been made to create a more representative database, and data sharing is becoming more common, but available data is still limited. The solution proposed in this article can not only significantly increase the size of the image data set, but also increase the diversity of the data. This is because the natural change in each image is divided into smaller parts and treated indirectly.

Singh et al. [29] have presented various disease classification techniques used to detect plant leaf diseases and image segmentation techniques that can be used for automatic detection and subsequent classification of plant leaf diseases. Bananas, beans, jackfruit, lemon, mango, potatoes, tomatoes and sapota are just a few of the 10 that have been tested using the proposed method. The optimal results were obtained through computational effort to specify the effectiveness of the proposed method in identifying and classifying leaf diseases. Another advantage of using this method is that it can detect plant diseases early or early. Artificial neural networks, bias classifiers, fuzzy logic and hybrid algorithms can be used to improve the identification rate of the classification process.

3. Problem methodology and system model

3.1 Problem methodology

Pantazi et al. [30] have presented classifier has been proven to detect four health conditions, including healthy and fluffy mold, powdery mildew, and black rot. The model was trained on the leaves of the vine to identify four healthy conditions. The present application has a high generalization ability, as demonstrated by testing on different leaf samples of different plants. The results showed that the model was efficient in most cases. More specifically, 44 of the 46 plant diseases examined were classified as successful, with an overall success rate of 95%. In order to accurately classify 50% of cases, conflict resolution has been shown to be important in obtaining 100% identification. From [21]-[30], we review different problems are (1) generating smaller models for specific crop or, (2) to generate a unique multi-crop model in a much more complex task but with the benefit of the entire multiple crop image dataset variability to enrich image feature description learning. Moreover, effective system of crop detection requires adaptation to demanding conditions such as different weather conditions and image capture conditions. In this paper, we propose an optimal automated disease detection and classification of crop species (OADD-CS) using hybrid machine learning techniques.

- The first contribution is to propose non-linear deep neural network (NL-DNN) for preprocessing that remove the abnormal image corners from the input image.
- Next, top and bottom leaf edges are compute by the edge target detection (ETD) algorithm and perform the feature extractions.
- Finally, a hybrid crow optimization based convolutional neural network (HCO-CNN) based classifier is used to detect different diseases. Here, we consider dataset of more than one hundred-thousand images taken by cell phone in real field wild conditions.
- This dataset contains almost equally distributed disease stages of seventeen diseases and five crops (wheat, barley, corn, rice and rape-seed) where several diseases can be present on the same picture. The proposed work can implement in MATLAB tool and the performance is compare with existing state-of-art works.

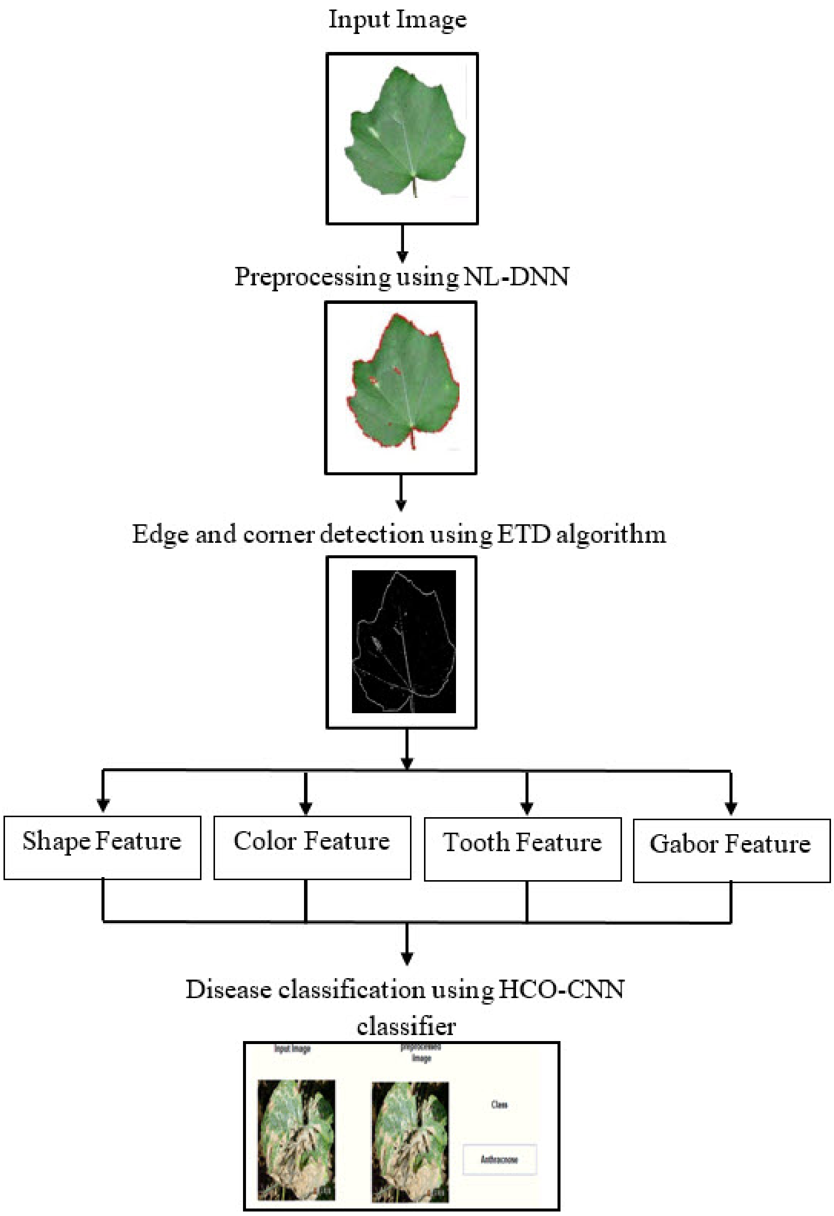


Fig. 1 System model of proposed optimal automated disease detection and classification of crop species

3.2 System model of proposed OADD-CS

Input images are taken for crop analysis. For the input image, the first pre-processing is done by a linear deep neural network (NL-DNN). Plant margins are identified using the edge target detection (ETD) method. Hybrid crow optimization based convolutional neural network (HCO-CNN) classifiers are used to diagnose various diseases. Tried wheat, barley, corn, rice, and rapeseed seeds.

4. Proposed OADD-CS method

In section, we first introduce the preprocessing task with proposed NL-DNN technique. Then, the edge and corner detection is achieved by ETD algorithm, which provides feature of the plant leaf. Finally, we illustrates HCO-CNN classifier for classifying the diseases in the given leaf plant. This dataset contains almost equally distributed disease stages of seventeen diseases and five crops are wheat, barley, corn, rice and rape-seed, where several diseases can be present on the same picture.

4.1 Preprocessing using non-linear deep neural network (NL-DNN) technique

Preprocessing is an important step in the image processing system. Pre-treatment process can be done to minimize processing problems by removing the contaminants or serious contents of the image and improving the image quality. Each pixel is addressed and replaced with the nearest statistic n. The surrounding value is computed from the surrounding pixels, so it is more stable for foreigners and does not create new realistic pixel values, preventing edge blurring and loss of image information. It also protects clear high-frequency details. Increasing the window size mathematically increases the effect of the average filter on noise removal:

$$\hat{Z}(i,j) = \underset{(s,t)\in S_{ij}}{median}\{g(s,t)\} \tag{1}$$

where, $Z(i,j)$ = median filter at a given coordinate and S_{ij} = coordinate of sub-image window of size $m \times n$.

Here, we introduce a nonlinear deep neural network (NL-DN) technique for early treatment of disease classification. Clustering operation with similar properties until the given algorithm reaches the reference limit. The existing KNN for the clustering process looks like this: NL-DNN is an important sequence computation with various functions, which is widely used in many fields. The basic point of the NL-DNA ordering system is to find the system status using class labels. Categorized by:

$$q_x' = \underset{L\in\{L_1, L_2\}}{\arg\max} \sum_{p_k\in\alpha(p_x)} F(q_x = L) \tag{2}$$

$$q_x' = \max\left\{ \sum_{p_k\in(p_x)} F(q_x = L_1) \sum_{p_k\in(p_x)} F(q_x = L_2) \right\} \tag{3}$$

where, L_1 and L_2 are the classification labels; q_x' is the predicted label; $F(\cdot)$ is represents the function values (1 or 0) depend on the true/ false condition; and $\alpha(p_x)$ denotes the nearest neighbors. The correspondence among the examples is resolute using Euclidean distance computation given as:

$$dist(p,q) = \sqrt{\sum_{k=1}^{K} (p_k - q_k)^2} \tag{4}$$

where, p, q are the two sample values and 'K' is represents the number of feature qualities. In this way, the NL-DNN grouping should be possible dependent on the nearest neighbor through the accompanying portrayal:

$$q_x' = \underset{L\in\{L_1, L_2\}}{\arg\max} \sum_{p_k\in\alpha(p_x)} F(q_x = L) \frac{1}{dist(p_x, p_k)} \tag{5}$$

In the above equation $dist(p_x, p_k)$ is the distance among the testing with training samples. The nearby adaptive distance among p_x and the training example p_k is defined as:

$$d_{new}(p_x, p_k) = d(p_x, p_k)/q_x^{\lambda} \ (\lambda > 0) \tag{6}$$

4.2 Edge detection algorithm for edge target detection (ETD) algorithm

Cane Edge Detection Algorithm is a multi-level edge detection algorithm designed to detect sharp edges in an image. Although more complex than other edge detection algorithms, Kani is the best edge detection algorithm. Kane's algorithm is based on mathematical representation of three main objectives. Low error rate; the identified edge is as close as possible to the original edge. Contour points can be well defined. The minimum distance is compute between the specified point and the actual edge center. Single edge point response; the sensor has only one point on the right edge and the smallest local maximum on the right edge.

This is the most effective way of finding the edges of an image. A softening filter is used before the film is received. The canopy edge detection algorithm creates a single pixel thick edge and merges the dashed lines. The steps of the Kanine algorithm are: Transferring the Gaussian to the kernel reduces the noise in the image. The Average or Average filter can be used outside the Gaussian Cyan filter. In this way, the size and direction of the slope are calculated. With maximum non-compression, the edges are lightweight. Binary thresholds are used, which results in undesirable partitioning. After weak separation, the shape is used and the main image is given an important edge [11]. In the next subsection, the steps of the improved canopy algorithm are described. The average filter is a low-pass filter and can process signals at specific frequencies. Gaussian filtering is used to blur, dim, and reduce image noise. Gaussian Xian filtering is mainly used for digital image processing to simplify the image and reduce the noise, which includes the average pixel value of the image. To find the edges, you need to smooth the image.

$$G(x,y)=\frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{7}$$

The Gaussian function is an operator used for smoothing. Here, x, y gives the standard deviation of the image coordinate information σ, associated with the probability distribution. Three different kernel operators are used to determine the slope used in the steps of the Kani algorithm. After the Gaussian filtering phase of the canopy edge detection algorithm, the Sobel, Robert, and Brewitt kernels were used separately in the gradient operator phase. The Sobel operator is a unique differential operator used to approximate the slope of the image intensity function for edge detection. For each pixel in the image, the Sobel operator normalizes to the corresponding gradient vector or vector. The kernel divides the input image and computes the slope size and orientation. It uses the following 3x3 two kernels.

$$D_i=\begin{bmatrix}1 & 2 & 1\\ 0 & 0 & 0\\ -1 & -2 & -1\end{bmatrix} \tag{8}$$

$$D_j=\begin{bmatrix}1 & 0 & -1\\ 2 & 0 & -2\\ 1 & 0 & -1\end{bmatrix} \tag{9}$$

It has slower computational efficiency than the Robert operator, but the larger kernel makes the sound more sensitive than the Robert operator. The larger the mask, the lower the error due to the influence of the local average noise nears the mask. First we calculate the shape of the square of the differences between adjacent pixels, and then estimate the approximate slope of the image. The input image is connected to the operator's default kernel, and the slope size and orientation are calculated. The following 2 x2 uses two kernels.

$$D_x=\begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix} \tag{10}$$

$$D_y=\begin{bmatrix}0 & 1\\ -1 & 0\end{bmatrix} \tag{11}$$

The plus factor of this operator is its simplicity but having small kernel it is highly sensitive to noise not and not much compatible with today's technology.

4.2.1 Edge target detection (ETD) algorithm

ETD algorithm is based on the use of a first order derivative, or can say gradient based. If I (i , j) be the input image, then image gradient is given by following formula:

$$\nabla I(i,j)=\hat{i}\frac{\partial I(i,j)}{\partial i}+\hat{j}\frac{\partial I(i,j)}{\partial j} \tag{12}$$

Where $\frac{\partial I(i,j)}{\partial i}$ is the gradient in the i direction and $\frac{\partial I(i,j)}{\partial j}$ is the gradient in the j direction. The gradient magnitude can be computed by the formula:

$$|G|=\sqrt{\left(\frac{\partial I}{\partial i}\right)^2+\left(\frac{\partial I}{\partial j}\right)^2} \tag{13}$$

$$|G|=\sqrt{G_i^{\,2}+G_j^{\,2}} \tag{14}$$

The gradient magnitude can be computed by the formula:

$$\theta=\arctan\left(\frac{G_j}{G_i}\right) \tag{15}$$

The magnitude of gradient computed above gives edge strength and the gradient direction is always perpendicular to the direction of edge. The algorithm two gives the edge detection algorithm.

Algorithm 1

Input	Preprocessed input image
Output	edge detection and feature extractions
1	Read the input image.
2	Convolve the resultant image with chosen operator's gradient mask in i axis and j axis.
3	Set a threshold value, T.
4	For a pixel M (i, j)
5	Compute the gradient magnitude say G .
6	If(G > T)
7	Mark pixel as an "edge".
8	else
8	Consider the next neighbor pixel.
9	end
Return edges, features	

4.3 Hybrid crow optimization based convolutional neural network (HCO-CNN) classifier

Convloutional neural network (CNN) is the best deep learning (DL) model [26]. CNN's mission is to transfer data across multiple transformation layers to represent data in a complex, concise, and hierarchical way. The structure of a CNN usually consists of three parts: the input layer, the hidden layer, the layer output layer, and each layer has interconnected processing units. The general structure of the CNN is shown in Figure 2. In CNN, each layer uses a linear shift in the input and represents the output. CNN is the vector of the input network and ve is the output vector.

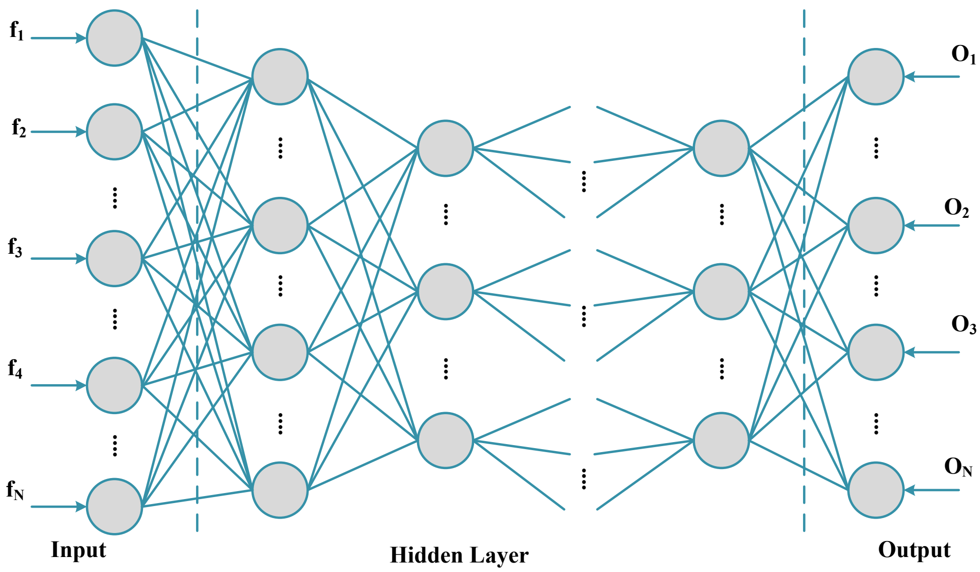


Fig. 2 Typical structure of the CNN

Consider $[f_m]$ be the input features where $1 \leq m \leq N$ and 'O' denotes the output data sets. The summed up model of the neural network can be given as 'O' for a yield of the whole network and 'O_H' for a yield of hidden layer.

Like CNNs, there are more hidden layers, and the weight of the main hidden layer exacerbates individual hidden information sources. Similarly, individually hidden element output multiplies by another set of weights in the second hidden layer. In the hidden first layer, the weight value of the information increases in the condition along the slope of the neuron (15).

$$O_{H\_1}(x=1,2..,K)=\left(\sum_{m=1}^{M} w_{xm}\ f_m\right)+B_x \tag{16}$$

where, B_x represents the constant value known as bias, w_{xm} is the interconnection weight between the input feature and first hidden layer with M and K denote the quantity of input and hidden nodes in the main hidden layer.

The activation function of the first hidden layer output is given as,

$$F\left(O_{H\_1}(x)\right)=\frac{1}{\left(1+e^{-O_{H\_1}(x)}\right)} \tag{17}$$

where, $F(\cdot)$ is the sigmoid activation function. Therefore, the operation of n^{th} hidden layer can be represented as,

$$O_{H\_n}(q)=\left(\sum_{z=1}^{K} w_{qx} F\left(O_{H\_(n-1)}(x)\right)\right)+B_q \tag{18}$$

where B_q represents the bias of q^{th} hidden node, w_{qx} is the interconnection weight between the $(n-1)^{th}$ hidden layer and $(n)^{th}$ hidden layer with K hidden nodes. The actuation work which is the yield of the n^{th} hidden layer is defined as,

$$F\left(O_{H\_y}(q)\right)=\frac{1}{\left(1+e^{-O_{H\_n}(q)}\right)} \tag{19}$$

At the output layer, the output of n^{th} hidden layer is again duplicated with the interconnection weights (i.e. weight between the n^{th} hidden layer and output layer) and afterward summarized with the bias B_p as

$$O(p)=F\left(\sum_{p=1}^{K} w_{pq}\ F\left(O_{H\_n}\ (q)\right)+B_p\right) \tag{20}$$

where w_{pq} represents the interconnection weight at the n^{th} hidden layer and output layer having q^{th} and p^{th} nodes individually. The initiation work at the yield layer goes about as the yield of the entire model. Presently the output of the model is differed from the target and the error is attained to enhance the output of the model. The calculation of error is defined as follows,

$$Error=\frac{1}{M}\sum_{m=1}^{M}\left(Actual(O_m)-T\arg et(O_T)\right)^2 \tag{21}$$

Where, $T\arg et(O_T)$ denotes the target output and $Actual(O_m)$ is the real output. The error must be minimized to attain the improved DFNN. Consequently, the weight values must be balanced until the error gets diminished at each iteration.

4.3.1 Feature selection using crow search optimization algorithm

The behavior of running after targets of cat is applied to tracing mode. Therefore, it is very clear that classifier should be a tiny value in order to guarantee that the crows spend most of the time in seeking mode. The process of crow search optimization can be described in the following four steps.

Initialization:

i) Create M cats in the process. Here, the cat represents the solution set.
ii) Randomly position the cats into the N-dimensional solution space and randomly select values, which are in-range of the maximum velocity, to the velocities of each cat.

$$A = \{V_1, V_2, ..., V_N\} \tag{22}$$

$$V_i \in \{x_i, y_i\}, \quad where\ 0 < i < N \tag{23}$$

The optimal features are selected using the crow search optimization algorithm once the feature points are extracted. With a random set of solutions that are assigned with the binary equivalent values, the algorithm is initialized. The feature points corresponding to the binary value 1 from every speech signal are selected by the algorithm. Compute opposite position of the cats using equation (7).

$$V_i' = (x_i' + y_i' - V')_i \tag{24}$$

Here, *A* is the population, *V* is the solution, *x* and *y* represents the velocity and position respectively. Then haphazardly pick number of cats and set them into tracing mode according to MR and the others set into seeking mode.

Fitness evaluation:

To evaluate the fitness value of each cat by applying the positions of cats into the fitness function, it represents the criteria of the goal. Here the cat represents the selected features. The fitness function can be evaluated as follows:

$$fit(V) = Min(EF) \tag{25}$$

In this work, the error function is considered as the fitness function. Here, *EF* represents the error function.

Updating the solution:

To update the best cat into memory using eqn. (25), note that it is only needed to remember the position of the best cat because it represents the best solution so far. Move the cats according to their flags, if cat_l is in seeking mode, apply the cat to the seeking mode process, otherwise apply it to the tracing mode process. Re-pick number of cats and set them into tracing mode according to MR, then set the other cats into seeking mode.

Termination:

Check the termination condition, if satisfied, terminate the program, otherwise repeat update process to stop the work.

5. Results and Discussion

In this section, we analyze the results of investigating the proposed method for identifying plant leaf images. Evaluate performance based on metrics such as sensitivity, specificity, accuracy, and rate of recognition. The total number of leaves used in the study was 80, where the training picture included 50 leaves of each type. The dataset includes 17 diseases and nearly distributed morbidity of 5 crops (wheat, barley, corn, rice, and rapeseed seeds), with multiple diseases occurring in the same area. Specific work can be performed on MATLAB tools, and performance is compared to traditional complex work. To arrive at a conclusion about the position of the leaf, the collision is achieved based on the proximity between the support vector and the new vector feature vector. The model accurately identifies the position of the model. A similar discrepancy solution was repeated, including the combination of the given samples (Anthracnose, Dow Mildew, and Gremolt). This is always the right conclusion about plant health.

5.1 Performance metrics

Accuracy, Sensitivity, and Uniqueness: To analyze accuracy, sensitivity, and specificity, TP (True Positive) detects many plant leaf images accurately, and Positive Positivity (FP) is the wrong number. Correctly reject plant leaf images and correct plant negative (FN) LA plant leaf rejection abuse pictures. The goal of the proposed FRVM method is to predict the type of leaf from the images of plants. In the kernel-based BSO method, the leaves have the same systematic structure, and the leaves are difficult to classify into similar shapes. Accuracy is defined as:

$$Accuracy = \frac{Number\ of TP + Number\ of TN}{Total\ Value} \tag{20}$$

Sensitivity rate is described as the probability that a test result will be positive when the leaf category is present. It is evaluated as follows

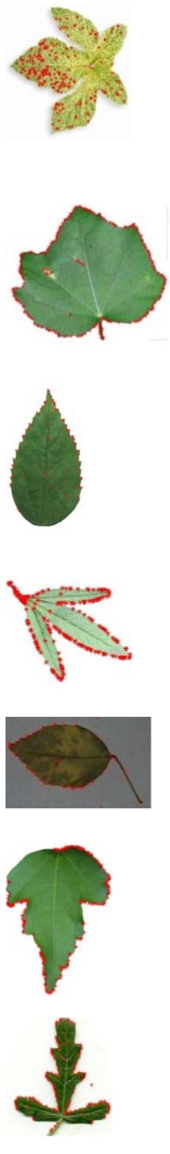
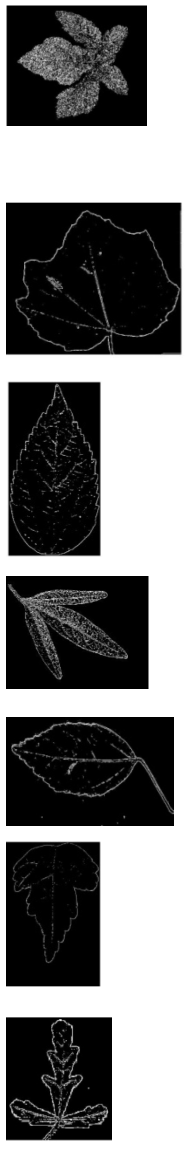
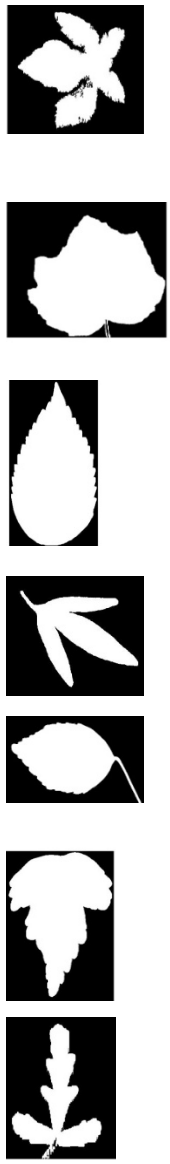
$$Sensitivity = \frac{Number\ of\ TP}{Number\ of\ TP + Number\ of\ FN} \tag{21}$$

Specificity rate is termed as the probability that a test result will be negative once the leaf category is not present and is determined as:

$$Specificity = \frac{Number\ of\ TP}{Number\ of\ FP + Number\ of\ TN} \tag{22}$$

The accuracy, sensitivity, and specificity of the proposed SRVM technology are higher than the existing SRVM technology. The precision, sensitivity, and specificity values of the existing SVM technology are 94.9, 95.3 and 97.02, respectively, and the specifications are 98.06, 99.83 and 97.73. From the graph it can be seen that the proposed FRVM classification system has higher accuracy, sensitivity and specificity than the existing SVM classification technology. The classifier function is used to estimate the values of TP, TN, FP and FN when there are 60 types of leaves with non-classification problems (Xie, 2011). This classifier is created and refreshes the classifier performance object (CB) that collects the classifier results.

Table 1 Analysis of the plant leafs

Original image	Corner	Edge	Segmentation
			

5.2 Comparison with state-of-art

Use these calculated TP, TN, FP, and FN values to measure performance metrics such as accuracy, sensitivity, and specificity. The leaf samples are taken from the ICL data set and processed using the proposed method. Figure 2 and 3 show the three sample leaf types of the original image in the third row, the cellular automata results in the fourth row, the enhanced images in the fifth row, the ROI function extraction in the sixth column, and the background normalized image shown in Figure 4. Give it. The feature output of the feature extraction is displayed in the eighth column and the 30 predicted angles in the last column. The angle of the leaf image is represented by 30 rough lines, the edges of the rows are green, and all the green dots are shown in red. Tables 2 and 3 show the confusion matrix of existing and proposed technologies, respectively, and the understanding of the experimental set is 13-tuple. The sensitivity, specificity, and accuracy values corresponding to the rotation of the edge geometry are 93.89%, 99.49% and 94.36%, respectively. The same discussion is summarized graphically in Figures 5 and 6.

(a) (b) (c)

Fig. 2 Disease detection using proposed classifier Input images with (a) Anthracnose type 1 (b) Anthracnose type 2 (c) Anthracnose type 3 diseases

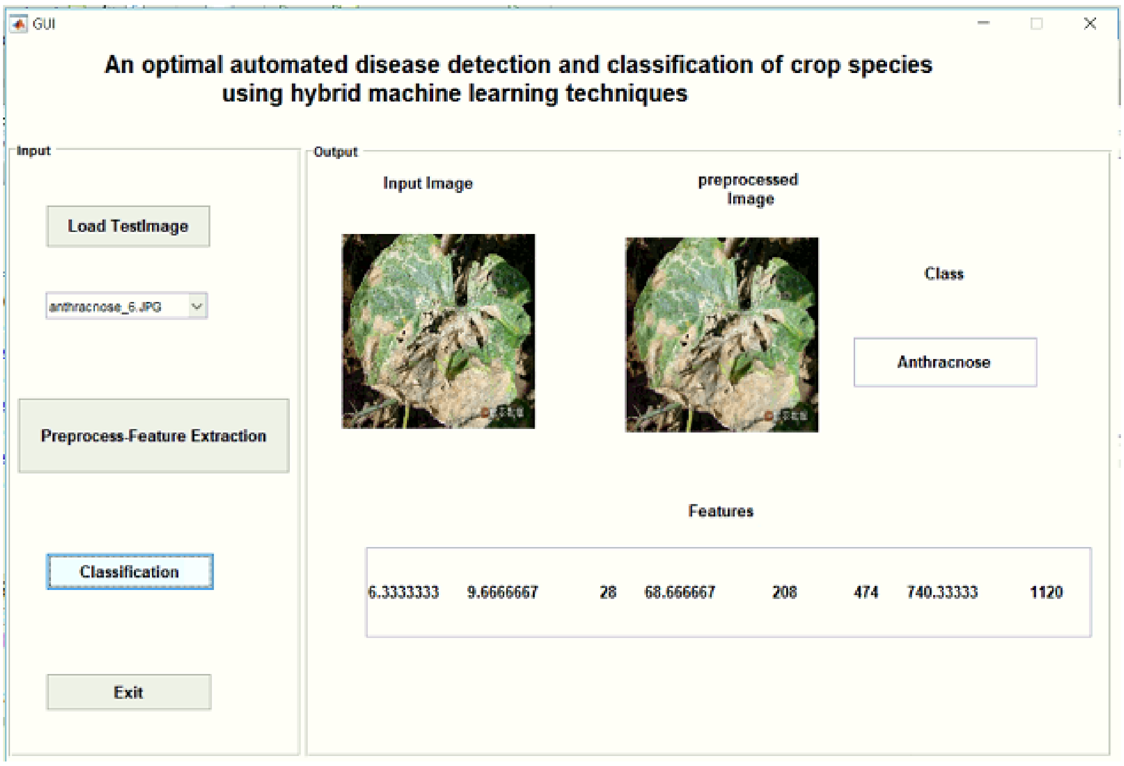


Fig. 3 Output GUI screenshot of proposed classifier

Table 2 Confusion Matrix of Existing Technique

Confusion matrix		Predicted label							
		1	2	3	4	5	6	7	8
Actual label	1	12	3	0	0	0	0	0	1
	2	0	15	0	0	0	0	0	0
	3	0	0	14	0	0	0	0	0
	4	0	2	0	15	0	0	0	0
	5	0	0	0	0	15	0	0	0
	6	0	0	0	0	0	15	0	0
	7	0	0	0	0	0	0	15	0
	8	0	0	0	0	0	0	0	14
Sensitivity	96.69%								
Specificity	99.72%								
Accuracy	97.44%								

Table 3 Confusion Matrix of Proposed Technique

Confusion matrix		Predicted label							
		1	2	3	4	5	6	7	8
Actual label	1	14	0	0	0	0	0	0	1
	2	0	15	0	0	0	0	0	0
	3	0	0	15	0	0	0	0	0
	4	0	2	0	13	0	0	0	0
	5	0	0	0	0	15	0	0	0
	6	0	0	0	0	0	15	0	0
	7	0	0	0	0	0	0	15	0
	8	0	0	0	0	0	0	0	15
Sensitivity	97.73								
Specificity	99.83								
Accuracy	98.06								

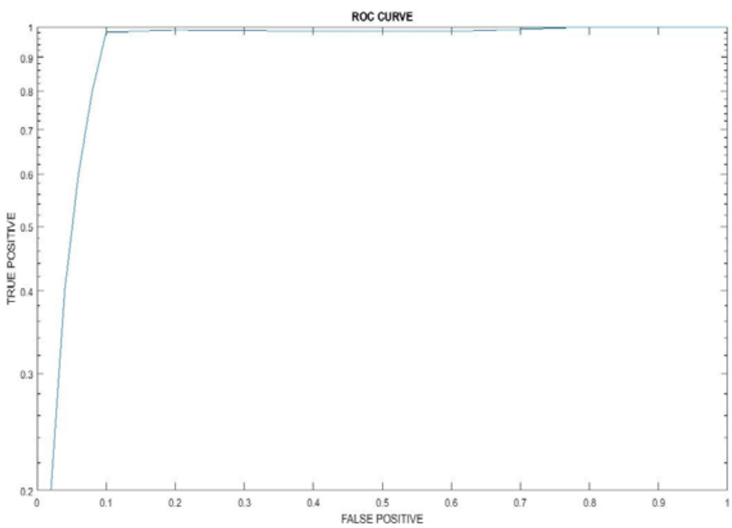


Fig. 4 ROC Curve

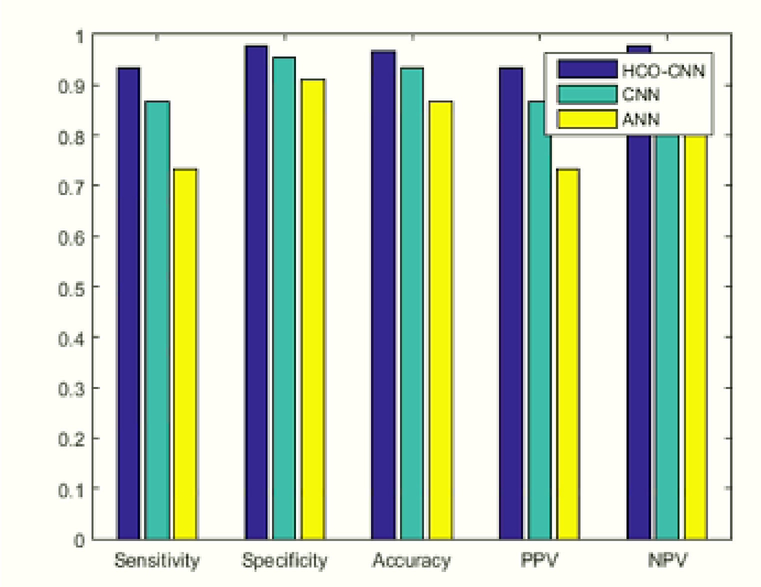


Fig. 5 performance comparison of proposed and existing classifiers

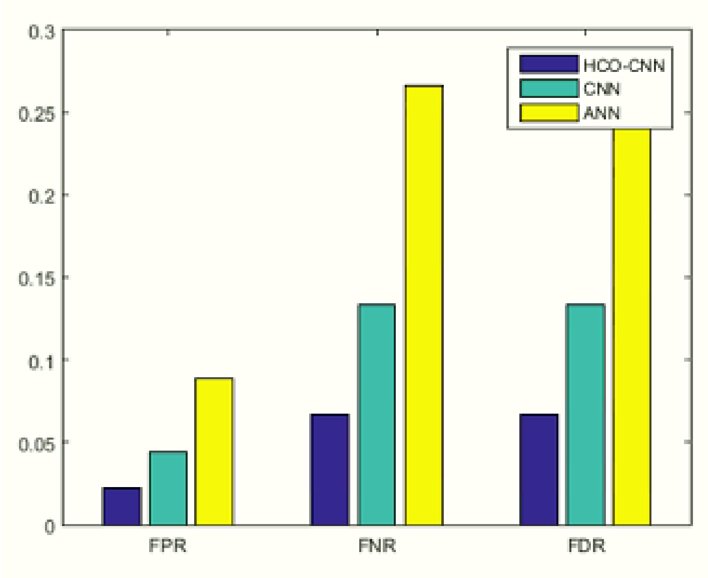


Fig. 6 FPR, FNR, FDR of proposed and existing classifiers

6. Conclusion

Using hybrid machine learning technology, we have proposed the optimal automatic detection and classification of crop species (OADD-CS). Nonlinear deep neural network (NL-DNN) is used for processing before removing abnormal image angles from the input image. Next, the top and bottom leaf edges are calculated using the edge target detection (ETD) method and feature extraction is performed. Hybrid crow optimization based convolutional neural network (HCO-CNN) classifiers are used to diagnose various diseases. Finally, the simulation results showed the effectiveness of the proposed classifier in terms of accuracy, sensitivity and specificity compared to conventional complex techniques.

References

[1] Saidi, M.N., Ladouce, N., Hadhri, R., Grima-Pettenati, J., Drira, N. and Gargouri-Bouzid, R., 2010. Identification and characterization of differentially expressed ESTs in date palm leaves affected by brittle leaf disease. Plant science, 179(4), pp.325-332.
[2] Rumpf, T., Mahlein, A.K., Steiner, U., Oerke, E.C., Dehne, H.W. and Plümer, L., 2010. Early detection and classification of plant diseases with Support Vector Machines based on hyperspectral reflectance. Computers and Electronics in Agriculture, 74(1), pp.91-99.
[3] Rizos, D., Feltrin, G. and Motavalli, M., 2011. Structural identification of a prototype pre-stressable leaf-spring based adaptive tuned mass damper: Nonlinear characterization and classification. Mechanical Systems and Signal Processing, 25(1), pp.205-221.
[4] Da, X., Yu, K., Shen, S., Zhang, Y., Wu, J. and Yi, H., 2012. Identification of differentially expressed genes in a spontaneous altered leaf shape mutant of the navel orange [Citrus sinensis (L.) Osbeck]. Plant Physiology and Biochemistry, 56, pp.97-103.
[5] Cope, J.S., Corney, D., Clark, J.Y., Remagnino, P. and Wilkin, P., 2012. Plant species identification using digital morphometrics: A review. Expert Systems with Applications, 39(8), pp.7562-7573.

[6] Diago, M.P., Fernandes, A.M., Millan, B., Tardáguila, J. and Melo-Pinto, P., 2013. Identification of grapevine varieties using leaf spectroscopy and partial least squares. Computers and electronics in agriculture, 99, pp.7-13.
[7] Du, J.X., Zhai, C.M. and Wang, Q.P., 2013. Recognition of plant leaf image based on fractal dimension features. Neurocomputing, 116, pp.150-156.
[8] Xia, C., Lee, J.M., Li, Y., Song, Y.H., Chung, B.K. and Chon, T.S., 2013. Plant leaf detection using modified active shape models. Biosystems engineering, 116(1), pp.23-35.
[9] Silva, M., Santana, A.S., Pimentel, R.M., Silva, F.C., Randau, K.P. and Soares, L.A., 2013. Anatomy of leaf and stem of Erythrinavelutina. RevistaBrasileira de Farmacognosia, 23(2), pp.200-206.
[10] Podda, A., Checcucci, G., Mouhaya, W., Centeno, D., Rofidal, V., Del Carratore, R., Luro, F., Morillon, R., Ollitrault, P. and Maserti, B.E., 2013. Salt-stress induced changes in the leaf proteome of diploid and tetraploid mandarins with contrasting Na+ and Cl− accumulation behaviour. Journal of plant physiology, 170(12), pp.1101-1112.
[11] Amin, A.H.M. and Khan, A.I., 2013. One-shot classification of 2-D leaf shapes using distributed hierarchical graph neuron (DHGN) scheme with k-NN classifier. Procedia Computer Science, 24, pp.84-96.
[12] Larbi, P.A., Ehsani, R., Salyani, M., Maja, J.M., Mishra, A. and Neto, J.C., 2013. Multispectral-based leaf detection system for spot sprayer application to control citrus psyllids. Biosystems engineering, 116(4), pp.509-517.
[13] Cerutti, G., Tougne, L., Coquin, D. and Vacavant, A., 2014. Leaf margins as sequences: A structural approach to leaf identification. Pattern Recognition Letters, 49, pp.177-184.
[14] Larese, M.G., Namías, R., Craviotto, R.M., Arango, M.R., Gallo, C. and Granitto, P.M., 2014. Automatic classification of legumes using leaf vein image features. Pattern Recognition, 47(1), pp.158-168.
[15] Chang, L.Y., Li, K.T., Yang, W.J., Chang, J.C. and Chang, M.W., 2014. Phenotypic classification of mulberry (Morus) species in Taiwan using numerical taxonomic analysis through the characterization of vegetative traits and chilling requirements. ScientiaHorticulturae, 176, pp.208-217.
[16] LIU, C.G., ZHOU, X.Q., CHEN, D.G., LI, L.J., LI, J.C. and CHEN, Y.D., 2014. Natural variation of leaf thickness and its association to yield traits in indica rice. Journal of integrative agriculture, 13(2), pp.316-325.
[17] Faria, F.A., Perre, P., Zucchi, R.A., Jorge, L.R., Lewinsohn, T.M., Rocha, A. and Torres, R.D.S., 2014. Automatic identification of fruit flies (Diptera: Tephritidae). Journal of Visual Communication and Image Representation, 25(7), pp.1516-1527.
[18] ZHANG, X.L., SI, B.W., FAN, C.M., Li, H.J. and Wang, X.M., 2014. Proteomics identification of differentially expressed leaf proteins in response to Setosphaeriaturcica infection in resistant maize. Journal of Integrative Agriculture, 13(4), pp.789-803.
[19] Joly, A., Goëau, H., Bonnet, P., Bakić, V., Barbe, J., Selmi, S., Yahiaoui, I., Carré, J., Mouysset, E., Molino, J.F. and Boujemaa, N., 2014. Interactive plant identification based on social image data. Ecological Informatics, 23, pp.22-34.
[20] Xing, J.J., Liu, Y.F., Li, Y.Q., Gong, H. and Zhou, Y.P., 2014. QSAR classification model for diverse series of antimicrobial agents using classification tree configured by modified particle swarm optimization. Chemometrics and Intelligent Laboratory Systems, 137, pp.82-90.
[21] Thenmozhi, K. and Reddy, U.S., 2019. Crop pest classification based on deep convolutional neural network and transfer learning. Computers and Electronics in Agriculture, 164, p.104906.
[22] Rangarajan, A.K., Purushothaman, R. and Ramesh, A., 2018. Tomato crop disease classification using pre-trained deep learning algorithm. Procedia computer science, 133, pp.1040-1047.
[23] Picon, A., Alvarez-Gila, A., Seitz, M., Ortiz-Barredo, A., Echazarra, J. and Johannes, A., 2019. Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild. Computers and Electronics in Agriculture, 161, pp.280-290.
[24] Kamal, K.C., Yin, Z., Wu, M. and Wu, Z., 2019. Depthwise separable convolution architectures for plant disease classification. Computers and Electronics in Agriculture, 165, p.104948.
[25] Ozguven, M.M. and Adem, K., 2019. Automatic detection and classification of leaf spot disease in sugar beet using deep learning algorithms. Physica A: Statistical Mechanics and its Applications, 535, p.122537.
[26] Toseef, M. and Khan, M.J., 2018. An intelligent mobile application for diagnosis of crop diseases in Pakistan using fuzzy inference system. Computers and Electronics in Agriculture, 153, pp.1-11.
[27] Roldán-Serrato, K.L., Escalante-Estrada, J.A.S. and Rodríguez-González, M.T., 2018. Automatic pest detection on bean and potato crops by applying neural classifiers. Engineering in Agriculture, Environment and Food, 11(4), pp.245-255.
[28] Barbedo, J.G.A., 2019. Plant disease identification from individual lesions and spots using deep learning. Biosystems Engineering, 180, pp.96-107.
[29] Singh, V. and Misra, A.K., 2017. Detection of plant leaf diseases using image segmentation and soft computing techniques. Information processing in Agriculture, 4(1), pp.41-49.
[30] Pantazi, X.E., Moshou, D. and Tamouridou, A.A., 2019. Automated leaf disease detection in different crop species through image features analysis and One Class Classifiers. Computers and electronics in agriculture, 156, pp.96-104.