Artificial Intelligence Review
https://doi.org/10.1007/s10462-020-09813-w

Deep neural network based Rider-Cuckoo Search Algorithm for plant disease detection

R. Cristin[1] · B. Santhosh Kumar[1] · C. Priya[2] · K. Karthick[3]

Abstract
Agriculture is the main source of wealth, and its contribution is essential to humans. However, several obstacles faced by the farmers are due to different kinds of plant diseases. The determination and anticipation of plant diseases are the major concerns and should be considered for maximizing productivity. This paper proposes an effective image processing method for plant disease identification. In this research, the input image is subjected to the pre-processing phase for removing the noise and artifacts present in the image. After obtaining the pre-processed image, it is subjected to the segmentation phase for obtaining the segments using piecewise fuzzy C-means clustering (piFCM). Each segment undergoes a feature extraction phase in which the texture features are extracted, which involves information gain, histogram of oriented gradients (HOG), and entropy. The obtained texture features are subjected to the classification phase, which uses the deep belief network (DBN). Here, the proposed Rider-CSA is employed for training the DBN. The proposed Rider-CSA is designed by integrating the rider optimization algorithm (ROA) and Cuckoo Search (CS). The experimental results proved that the proposed Rider-CSA-DBN outperformed other existing methods with maximal accuracy of 0.877, sensitivity of 0.862, and the specificity of 0.877, respectively.

Keywords Plant disease · Piecewise fuzzy C-means clustering · Deep belief network · Classification · Texture features

1 Introduction

In the Indian economy, Agriculture is considered as a crucial part, and also, it is considered as the source of income for numerous human beings in several countries (Kumar et al. 2018). Agriculture is a basic need for human survival. Thus, it becomes essential to maximize the production of vegetables, crops, and fruits for the developing countries,

✉ R. Cristin
cristin.r@gmrit.edu.in

[1] Department of CSE, GMR Institute of Technology, Rajam, AP 532127, India

[2] Department of Information Technology, Vels Institute of Science, Technology and Advanced Studies (VISTAS), Chennai 600117, India

[3] Department of EEE, GMR Institute of Technology, Rajam, AP 532127, India

Published online: 20 February 2020

such as India. Moreover, the quality of production should be high not only for productivity but also for attaining improved health. However, the quality and productivity of food get fraught by several factors, like disease spread, which could be avoided with early diagnosis. Most of these diseases are contagious, which can lead to the loss of crop yield (Singh 2018). Numerous varieties of food plants are harvested with the help of land**'s** environmental conditions, but simultaneously many issues are faced by the farmers, which involves water shortage, plant diseases, and natural disasters. However, most of these issues can be minimized by offering technical amenities to the farmers. One such system is automated plant disease identification systems, which helps the farmers to provide the technical facilities. These systems can conquer the problems by the experts using their prior knowledge (Kumar et al. 2018). In agriculture, an automated-based mechanized technique is employed for detecting the disease of plants. Thus, disease detection has become an important research topic in automatically monitoring, detecting, controlling, and mitigating the disease of plants. Plant disease is a significant factor that leads to a major reduction in both quantity and quality of plant production. The major task is to enhance the productivity of plants, which are done by plant disease detection and classification (Patil and Chandavale 2015). The disease occurred in plant leaves cause economic losses in the agricultural industries. Thus, the classification and identification of disease harshness are considered as an important factor for minimizing economic loss by controlling the diseases.

The wealth of the developing countries is generally based on agriculture, but the disease caused in plants badly affects agriculture. Thus, plant disease is controlled by enhancing the quality and quantity of agriculture (Ganesan et al. 2017). Furthermore, the automated systems maximize the productivity of food by exploiting on-time prevention from the plant disease without experts. These systems are cost-effective and require less time for the detection process (Kumar et al. 2018). The disease in the plants is considered as a major worldwide issue in agricultural productivity. The information related to plant virus disease is acknowledged in the developing countries. The plant disease affects agricultural production in underdeveloped and developed countries and is complex to access the research facilities. Overwhelming fatalities caused by the viruses has been observed in several countries of Southeast Asia (Akhter et al. 2019). Moreover, plant disease caused major production losses in the field of agriculture. The management of the disease is a major challenge. Generally, the disease and its indications, like streaks, or colored spots, are noticed on the stem of leaves of the plants. In plants, the majority of the leaf diseases occur due to viruses, and fungi. Moreover, the disease origin is characterized using numerous visual symptoms, which can be noticed in the parts of plants, like leaves or stems, and these symptoms are detected manually (Francis and Anoop 2016). The identification of plant disease is an essential step in the field of agriculture. Most of these cases are identified manually. The major obstacles using the visual assessment is a subjective task, as it becomes probe to cognitive or psychological phenomena, which may lead to errors, bias, and optical illusions (Barbedo 2016). The image processing (Daga et al. 2011; Thomas and Rangachar 2018; Aher and Waykar 2014) is essential for detecting the disease of plants as it offers improved results and minimizes human efforts (Francis and Anoop 2016).

Usually, it is a complicated task to determine the disease of crops using conventional methods. Computerized and automated methods employed for identifying the plant disease is beneficial as it determines the indications of plant diseases and minimizes the workloads in scrutinizing huge fields (Ganesan et al. 2017). Several clustering (Bharill et al. 2019a; Saxena et al. 2017) and classification methods (Kumar et al. 2018; Bharill et al. 2019b; Cheng et al. 2019; Ratre 2019) adapted for the disease identification in plant involves random forest (RF) (Kodovsky et al. 2012), k-nearest neighbor (kNN) (Guettari et al. 2016),

support vector machine (SVM) (Deepa 2017), neural network (Sheikhan et al. 2012; Gangappa and Mai 2019), learning approach (Prasad et al. 2018) and Fisher linear discriminant (FLD) (Ramezani and Ghaemmaghami 2010). Also, optimization algorithms (Menaga and Revathi 2018) have been utilized for plant disease detection. The methods based on image processing are applied for solving the issues of identification and removal of pretentious parts in an effective manner (Ganesan et al. 2017). Thus, the processing of image is considered as an important part for determining and examining the disease in plants. Moreover, the image processing is applied not only for disease discover, but also in several agricultural purposes, like identification of diseased leaf, stem, fruit and to compute the level of diseases (Ganesan et al. 2017). Numerous image processing methods are utilized in agriculture, which involves vegetable and fruits recognition, segmentation of soil and weed, quantification of diseases, and for disease classification and identification. The rural farmers pose the ability to establish the exact information regarding the seeds, insects, disease, and its prevention measures (Kamble 2018). The major technique adapted for determination and inspection of plant disease is done based on digital image processing algorithms. Thus, a variety of data transmission technologies and image acquisition devices are devised in order to capture the agronomical crop images and are transmitted to the information center (Guettari et al. 2016). Receiver operating characteristic (ROC) based performance evaluation has been performed since it is applied mostly in medical image segmentation (Göçeri 2016; Göçeri et al. 2013, 2014), which needs precise operations. ROC analysis can be applied to evaluate results obtained from different image modalities and segmented organs in medicine (Goceri 2011¸2013; Goceri and Goceri 2015).

This paper presents an approach for plant disease detection using deep learning (Rajora et al. 2018) concepts. Initially, the input image is fed to the pre-processing phase for removing the noise and artifacts present in the image. After obtaining the pre-processed image, it is subjected to the segmentation phase for obtaining the segments using piFCM. Each segment undergoes a feature extraction phase in which the texture features are extracted using information gain, HOG, and entropy. The obtained texture features are subjected to the classification phase, which uses DBN for the detection of plant disease. Here, the proposed Rider-CSA is employed for training the DBN. The proposed Rider-CSA is designed by integrating the ROA and CSA for DBN in order to detect the plant disease.

Proposed Rider-*CSA algorithm for plant disease detection* The proposed Rider-CSA is the integration of ROA in CSA in such a way that the developed algorithm aims at plant disease detection.

The paper is organized in the following manner: Sect. 2 discusses the existing methods of plant disease detection with the challenges of the methods, which remains the motivation of the method. The proposed method of plant disease detection is demonstrated in Sects. 3 and 4 reveals the effectiveness of the proposed method by comparing the methods with the conventional methods. Lastly, Sect. 5 provides a summary.

2 Literature survey

This section elaborates the literature survey of the plant disease detection techniques along with the disadvantages of the methods. At the end of the section, the challenges of the conventional methods are deliberated.

The eight existing techniques considered for plant disease detection are illustrated in this section. Zhou et al. (2018) developed a skeleton extraction method for

identifying and counting the maize seedlings for computing the azimuth angle. Here, the RGB images were converted to greyscale images for making the pre-processed images adapted to the threshold segmentation. Here the images were segmented using the threshold optimization, and the efficiency of the technique was verified by comparing automated measurements of the plant shapes. The method utilized the external pressure method (EPM)-skeleton-debarring method based on the saliency theory (SDST) method for overcoming the difficulties with the extraction of plant skeletons for accurate information. The method determined the seedling rate in an accurate manner and provided a valuable analytical tool for agronomic research. However, the method failed to analyze the utility of the extracted using the series of applications. Chouhan et al. (2018) developed a Bacterial foraging optimization based radial basis function neural network (BRBFNN) for identifying and classifying the plant diseases in an automatic manner. Here, a Bacterial foraging optimization (BFO) was used for assigning an optimal weight to the RBFNN classifier. The method increased the network efficiency by searching and grouping the seed points for initiating the feature extraction process. The method considered cedar apple rust, leaf spot, and rusts for the analysis. However, the method did not consider different databases, such as viruses or bacteria. Zhang et al. (2018) developed a deep convolutional neural networks (DCNN) for predicting the disease present in the leaves. The method used two models for training the DCNN by altering the parameters, changing the pooling layers, and adding dropout operations. The method provided improved accuracy and minimized the convergence iterations by enhancing the model training and recognition efficiency. However, the method was unable to make reasonable judgments about disease detection. Also, parameters should be chosen carefully particularly batch size, which has an important role in deep learning-based approaches (Goceri and Gooya 2018). Tetila et al. (2017) developed a simple linear iterative clustering method for tracking soybean foliar diseases in the crop using the images confined by the aerial vehicle model. The method was based on the segmentation method and used the clustering method for detecting the plant leaves of the images for describing the features using the foliar physical properties, like gradient, texture, and color. The method was a failure if the layers of the CNN are increased. Moreover, the method failed to use higher resolution cameras. Barbedo (2019) developed a Data augmentation and deep learning techniques for the plant disease classification. However, the method faced dataset limitations with respect to the sample varieties, which prevented the system for plant disease classification. This technique generated defective results with respect to incomplete data availability. Kumar et al. (2018) developed a Spider monkey optimization approach for fixing the significant features using the huge features set produced by the subtractive pixel adjacency model (SPAM). Here, the selected features were fed to the support vector machine for the classification of the diseased plants and the healthy plants by analyzing the characteristics of leaves. However, the method failed to determine the plant disease categories and created the multi-class problem. Chuanlei et al. (2017) developed a leaf disease recognition method for plant disease prediction. The method used color transformation structure in the Red, Green, and Blue (RGB) model was transformed to Hue, Saturation, and Intensity (HSI) and gray models. Here, the combined genetic algorithm (GA) and correlation-based feature selection (CFS) were used for selecting the features and enhanced the accuracy of leaf disease identification. At last, the diseases were identified using the SVM classifier. The method was unable to process the diseased leaf images. Lu et al. (2017) developed a rice disease identification method using Deep CNN for predicting the diseases. Here, the accuracy was

higher than the conventional machine learning model. However, the method was not applicable for diagnosing the fault in the distributed systems and for the time-varying systems. Shen et al. (2018) proposed a novel technique of trajectory clustering by means of submodular optimization for the enhanced motion segmentation in videos. Initially, a small number of appropriate trajectories are chosen, and then every initial trajectory are divided into fragments with the appropriate trajectories, which are considered as the fragment centers. At last, fragments are combined as clusters using a two-phase technique of bottom-up clustering. Shen et al. (2019) proposed the structure of increasing the energy of quadratic submodular with a knapsack constraint for solving the crisis of computer vision. Image segmentation and motion trajectory clustering algorithm were applied to reveal the effectiveness and maximization of the energy function, and finally, this outperforms the algorithm of classic segmentation for both random walks and graph cuts. Dong et al. (2018) proposed a new technique of hyperparameter optimization, which can determine the best hyperparameters by an action-prediction network controlled on Continuous Deep Q-Learning for a given sequence. Dong et al. (2019) proposed a novel quadruplet deep network to observe the potential connections with the training instances, which are aimed to attain the powerful representation and designed a shared network by means of four branches, and are connected by a novel loss function comprising of pair-loss and triplet-loss. Dong and Shen (2018) proposed a new triplet loss to remove the expressive deep features for object tracking by means of Siamese network structure instead of pairwise loss, and also the theoretical analysis is developed by merging the comparison of both the back-propagation and gradients to verify the effectiveness. Shen et al. (2016) proposed a technique of real-time image superpixel segmentation with 50 frames per second by means of density based spatial clustering of applications with noise (DBSCAN) algorithm, which achieves the better performance than the existing method and computation cost is less. But the compactness of superpixels has to be developed much better, and also, the structure of the current superpixel to the realtime video has to be extended. Shen et al. (2014) presented a new method of image superpixel segmentation by means of he lazy random walk (LRW) algorithm with self-loops, which includes the benefits of complicated regions of the texture, and segmenting the weak boundaries using the novel global probability. Improves the performance of superpixels by relocating the center positions of superpixels and then dividing the large superpixels into small ones. Dong et al. (2016) proposed a new algorithm of subMarkov random walk (subRW) for segmenting the seeded image, which can be interpreted as a traditional random walker on a graph with added auxiliary nodes. This algorithm outperforms the existing RW-based algorithms, which proves that it is realistic to intend a new subRW algorithm by adding new auxiliary nodes. Wang et al. (2019) designed a neural network with two branches for attention box prediction (ABP) and aesthetics assessment (AA). The ABP network is responsible for inferring the initial cropping, whereas the AA network finds out the final cropping. Wang et al. (2018) proposed a novel method of data augmentation which simulates the training data of video from an existing dataset of an annotated image, which enables our network to study the diverse saliency information and prevents overfitting through the limited number of training videos. Wang and Shen (2018) proposed a prediction of deep visual attention for capturing the information of hierarchical saliency from deep, fine layers through the response of local saliency, and coarse layers for shallowing through the information of global saliency. This method is based on the framework of skip-layer network, which calculates human attention from several convolutional layers by means of different reception fields.

2.1 Challenges

The challenges faced by conventional plant disease detection techniques are illustrated as follows:

- In Deng (2008), greenhouse management was introduced on the basis of the wireless sensor network in order to comprehend recent precision agriculture. This method helped to manage the greenhouse equipment and provided numerous services to users using handheld devices. The method was cost-effective and efficient but was complex to understand and requires prior knowledge for the execution.
- In Francis and Anoop (2016), the soft computing model was devised for identifying the evident patterns of a specific plant. The detection of stems, plants, and leaves determines the pest disease, but with poor accuracy.
- Artificial intelligence (AI) algorithm based on cloud image processing was utilized for improving the accuracy of the model. Even though it was highly scalable, but the performance was poor using real-life data (Singh 2018).
- The methods based on fuzzy were devised for identifying and segmenting the disease affected areas in plants leaves. The disease affected area was evaluated using the quality parameters. The method was highly efficient and robust but resulted in poor accuracy (Ganesan et al. 2017).
- The threshold-based segmentation method was introduced in Zhou et al. (2018) for separating the maize seedlings using the soil backgrounds. Here, the external pressure method was adapted for segmenting the results of skeleton extraction. Moreover, it utilized principal component analysis for determining the direction of the principal axis. The method was effective with respect to seedlings, but the utility of extracted images was poor.

3 Proposed DBN-based Rider-CSA method for determining plant disease

The proposed Rider-CSA-based DBN is elaborated for effective image processing based on plant disease identification method is described in this section. The proposed model undergoes four phases for performing the disease discovery in the plants. At first, the input image is fed to the pre-processing step for removing the noise and artifacts present in the image. The obtained pre-processed image is given to the segmentation phase for producing the segments. The segmentation is performed using the piFCM (Wu et al. 2017) to extract the region of the image. The segmented results are fed to the feature extraction phase, where the texture features are extracted using the information gain, HOG, and entropy. The information gain is utilized for ranking features and thus, the high information gain is ranked higher and is powerful in classifying the data. The entropy is utilized for reflecting the amount of information and HOG is used for providing better matching results. Thus, the three features pose the capability to produce improved plant disease detection. Finally, the texture features obtained are fed to the classification phase for detecting the presence of plant diseases. For the classification, the proposed Rider-CSA-based DBN is employed in which the proposed Rider-CSA is adapted for training the DBN (Hinton et al. 2006). The proposed Rider-CSA is designed

by integrating the ROA (Binu and Kariyappa 2018) and CS (Gandomi et al. 2013) algorithm. Thus, the proposed Rider-CSA based DBN is responsible for determining the plant diseases by tuning the weights in order to obtain optimal weights for training the DBN. Thus, the proposed Rider-CSA based DBN is responsible for classification by analyzing the plant images. Figure 1 illustrates the schematic diagram of the proposed Rider-CSA based DBN model for detecting the plant disease. Assume a database D with a images as represented below,

$$D = \left\{D_1, D_2, \ldots, D_b, \ldots, D_a\right\} \tag{1}$$

where D_b specifies the bth input image.

3.1 Pre-processing for removing noise

The importance of pre-processing is to facilitate the smooth processing of the input image, and it is performed using the filtering methods. The pre-processing is carried out for removing the noise and artifacts present in the image. The filters adapt the image and eliminate the noise contained in the image.

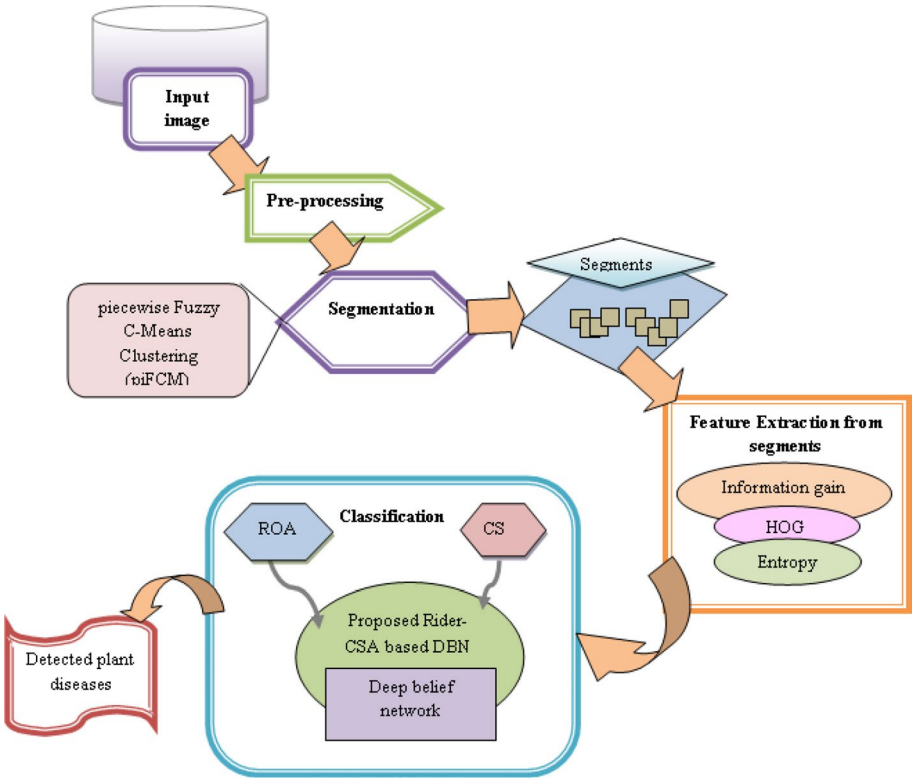

Fig. 1 Architecture of the proposed DBN-based Rider-CSA for plant disease detection

3.2 Segmentation based on piFCM for classification

This section elaborates the segmentation process for partitioning the images into different segments. Segmentation aims to simplify the image representation into something, which is understandable and simpler to evaluate. In this research, the piFCM is adapted for generating the segments from the original image D_b. The piFCM technique (Wu et al. 2017) aims to provide image segments based on particular image features, which are used in the domain of real-world application. The technique of fuzzy classification permits precise information convey, which examines the substructure in data, explores the feasibility for the algorithm of centroid based cluster, depicts the clustering internally through spectral interpretations and external data by means of environmental and soil variables (Heil et al. 2019). The piFCM poses improved global convergence based on the iterative process. Moreover, the piFCM technique helps to boost the performance with flexible utility functions. The piFCM helps to generate segments of the image using image characteristics and intensity level of the pixel. For eliminating the variations caused by the illumination, the piFCM used membership degrees for segmenting the image. In the segmentation phase, each D_b is broken into smaller parts using the piFCM approach by introducing the multi-membership data. Consider $H = \{h_1, \ldots, h_j\}$ be a membership dataset obtained from the set of basic partitions, which is given by,

$$k = \left\{\alpha^{(1)}, \ldots \alpha^{(l)}\right\} \tag{2}$$

with

$$h_m = \left(\alpha_m^{(1)}, \ldots, \alpha_m^{(n)}, \ldots, \alpha_m^{(l)}\right)^T \quad \forall m \tag{3}$$

where h_m corresponds to D_b and aligns the membership degrees of D_b to basic partition. The piFCM was applied for the fuzzy clustering of H. Consider $s = \{s_1, \ldots, s_p\}$ with

$$s_p = \left(s_p^{(1)}, s_p^{(n)}, \ldots, s_p^{(l)}\right)^T \tag{4}$$

The weights are set to $\left\{W_n\right\}_{n=1}^{l}$ and derive,

$$t = \left(\alpha_m^{(n)}, s_p^{(n)}\right) = \theta\left(\alpha_m^{(n)}\right) - \theta\left(s_p^{(n)}\right) - \left(\alpha_m^{(n)} - s_p^{(n)}\right)^T \nabla\theta\left(\alpha_m^{(n)}\right) \tag{5}$$

The distance from the α_m to s_p is given by,

$$d\left(\alpha_m, s_p\right) = \sum_{n=1}^{l} w_n t\left(\alpha_m^n s_p^n\right) \quad \forall m, p \tag{6}$$

$$\alpha_{mp} = \frac{d\left(\alpha_m s_p\right)^{\frac{-1}{(C-1)}}}{\sum_{p=1}^{J} d(\alpha_m, s_p)^{-1/C-1}} \quad \forall m, p \tag{7}$$

$$s_p^{(n)} = \sum_{m=1}^{j} \frac{\left(\alpha_{m,p}\right)^C}{||\alpha_p^c||} \mu_m^{(n)} \quad \forall p, n \tag{8}$$

Then, the obtained equation is given as,

$$\alpha = \left[\alpha_{m,p}\right]_{j\times J} = K(s) \quad \text{and} \quad s = \left[s_p\right]_{p=1}^{J} = L(\alpha) \tag{9}$$

where K and L are the two mapping functions defined in the above equation. The obtained segments are given as a result of piFCM process and are represented as,

$$S_b = \left\{ S_b^1, S_b^2, \ldots, S_b^\gamma, \ldots S_g^b \right\} \tag{10}$$

where S_b^γ represents γth segment of the bth image, and 'g' denote the total segments generated from bth image. The segments generated from the piFCM are determined using D_b and are adapted for extracting features.

3.3 Feature extraction for the classification

The section illustrates the noteworthy features extracted from the segmented images and the significance of feature extraction is to generate the highly relevant features that enable the better detection of the plant diseases. In contrast, the complications while analyzing the image are minimized due to the reduced features set. Moreover, the accuracy associated with the detection is assured through the effective feature extraction. The features employed for the identification of plant diseases are information gain, entropy, and HOG.

(i) Entropy

Entropy (Wang et al. 2018) is a significant factor that can be utilized for measuring the information content of the images and poses the ability to distinguish the information and noise. The entropy is given as,

$$F_1 = -\sum_{r=0}^{N-1} P_i \log P_i \tag{11}$$

where N is total gray levels of the image, P_i is the ratio of number of pixels to ith gray values and is given by,

$$P_i = \frac{Q_i}{Q} \quad \text{and} \quad Q = \sum_{i=1}^{N-1} Q_i \tag{12}$$

where $P = \{P_0, P_1, \ldots, P_{N-1}\}$.

(ii) Information gain

The information gain (Roobaert et al. 2006; Sui 2013) is defined as a measure, which is used to evaluate the amount of obtained information by analyzing the attribute value, which is based on the entropy. The information gain is formulated as,

$$F_2 = T(S) - \sum_b \frac{S_b}{S} T(S_b) \tag{13}$$

where $T(S)$ is the entropy of the given segment, and $T(S_b)$ is the entropy of the bth subset produced by partitioning S based on the feature.

(iii) HOG

The HOG (Damer and Führer 2012) is a kind of feature descriptor, which is used in the image processing for determining the objects present in the image. This technique utilizes gradients for localizing the image. The HOG features are extracted using the magnitude and orientation and are given by $X(u, v)$ and $Y(u, v)$. The computation of magnitude using a Sobel filter is given as,

$$X(u, v) = \sqrt{du(u, v)^2 + dv(u, v)^2} \tag{14}$$

where $du(u, v)$ represents the x directional gradients and $dv(u, v)$ indicates the y directional gradients. The computation of orientation using Sobel filter is given as,

$$Y(u, v) = \tan \begin{cases} \tan^{-1}\left(\frac{dv(u,v)}{du(u,v)}\right) - \pi & \text{if } du(u, v) < 0 \text{ and } dv(u, v) < 0 \\ \tan^{-1}\left(\frac{dv(u,v)}{du(u,v)}\right) + \pi & \text{if } du(u, v) < 0 \text{ and } dv(u, v) > 0 \\ \tan^{-1}\left(\frac{dv(u,v)}{du(u,v)}\right) & \text{Otherwise} \end{cases} \tag{15}$$

Thus, the HOG features are computed using magnitude and orientation and its is formulated as,

$$F_3 = X(u, v) + Y(u, v) \tag{16}$$

3.3.1 Computation of the feature vector

The features of the fth segment, like information gain, entropy, and HOG features are associated to symbolize the features of the fth segment. The feature vector of the jth segment is expressed as,

$$F\left[S_b^f\right] = \left\{F_1, F_2, F_3\right\} \tag{17}$$

3.4 Classification using proposed DBN classifier

This section elaborates the proposed Rider-CSA based DBN classifier to classify the images for disease detection. Here, the Rider-CSA based DBN is proposed by incorporating Rider-CSA in the DBN model, for selecting the optimal weights in the Rider-CSA based DBN. The proposed Rider-CSA helps in modifying the performance of

DBN by selecting the optimal weights. The architecture of Rider-CSA-based DBN and its training process are illustrated in the subsections.

3.4.1 Architecture of DBN classifier

The DBN (Hinton et al. 2006) is a part of Deep Neural Network (DNN) and contains different layers of restricted Boltzmann machines (RBMs) and multilayer perceptrons (MLPs). RBMs contain hidden and visible units, which are linked based on weighted connections. The MLPs are considered as the feed-forward networks that contain input, hidden, and output layers. The network with multiple layers can solve any complicated tasks and thereby make the classification of images more effective for evaluating the plant disease. Figure 2 shows the architecture of DBN.

The input given to the visible layer is the attribute information obtained by evaluating the features and the hidden layer of the first RBM is expressed as,

$$F^1 = \left\{F_1^1, F_2^1, \ldots, F_x^1, \ldots, F_3^1\right\}; \quad 1 \le x \le 3 \tag{18}$$

where F_x^1 denotes the xth visible neuron in the first RBM. The hidden layers in the first RBM is given as,

$$A^1 = \left\{A_1^1, A_2^1, \ldots, A_o^1, \ldots, A_e^1\right\}; \quad 1 \le o \le e \tag{19}$$

where A_o^1 denote the oth hidden neuron and e indicates total hidden neurons. Each neuron contains a bias in its visible layer and hidden layer. Let y and z represents the biases of visible layer and hidden layer, and these biases are expressed as,

$$y^1 = \left\{y_1^1, y_2^1, \ldots, y_x^1, \ldots, y_3^1\right\} \tag{20}$$

where y_x^1 denote the bias corresponding to xth visible neuron.

$$z^1 = \left\{z_1^1, z_2^1, \ldots, z_o^1, \ldots, z_e^1\right\} \tag{21}$$

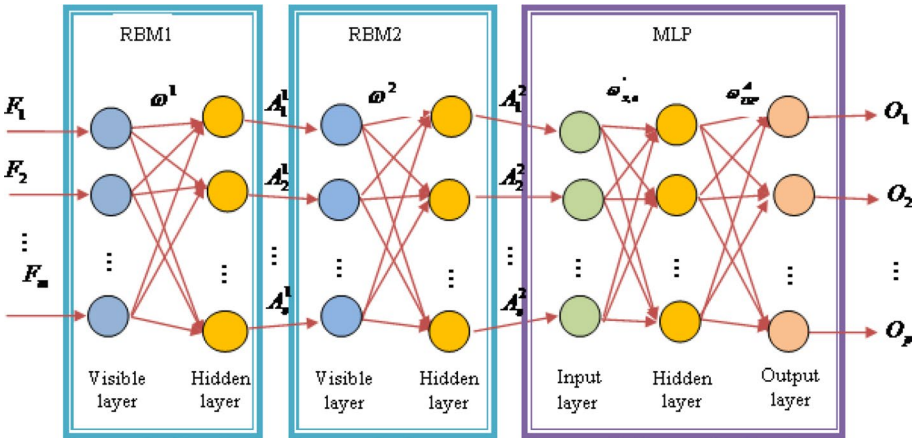

Fig. 2 Structural design of DBN classifier

where and z_o^1 represents the bias corresponding to oth hidden neuron. The weights used in the first RBM is given as,

$$\omega^1 = \left\{ \omega_{x,o}^1 \right\}; \quad 1 \le x \le 3; 1 \le o \le e \tag{22}$$

where $\omega_{x,o}^1$ represents the weight between xth visible neuron and o^{th} hidden neuron. Thus, the output of the hidden layer from the first RBM is computed using bias, and the weights linked with each visible neuron and is given as,

$$A_o^1 = \beta \left[z_o^1 + \sum_x (F_x)^1 \omega_{x,o}^1 \right] \tag{23}$$

where β is the activation function. Thus, the output computed from the first RBM is represented as,

$$A^1 = \left\{ A_o^1 \right\}; \quad 1 \le o \le e \tag{24}$$

Thus, the output of the first RBM is given as the input to the visible layer of the second RBM as input. Hence, the input of second RBM layer is denoted as, F^2. Likewise, the hidden layer of the second RBM is represented as,A^2. The bias in the visible layer and the hidden layer is indicated as, y^2 and z^2. The second RBM weight vector is given as, ω^2 the output of the oth hidden neuron is given as A_o^2. Thus, the output generated from the hidden layer is given by A^2. The output generated from the hidden layers of the second RBM is subjected to MLP as an input and is represented as,

$$I = \left\{ I_1, I_2, \ldots, I_o, \ldots, I_e \right\} = \left\{ A_o^2 \right\}; \quad 1 \le o \le e \tag{25}$$

where e denote the total neurons present in the input layer, which is given by the output of hidden layer of second RBM $\left\{ A_o^2 \right\}$. The hidden layer of the MLP is given as,

$$Z = \left\{ Z_1, Z_2, \ldots, Z_U, \ldots, Z_V \right\}; \quad 1 \le U \le V \tag{26}$$

where V represents total hidden neurons. The output layer of the MLP is given as,

$$O = \left\{ O_1, O_2, \ldots, O_P, \ldots, O_R \right\}; \quad 1 \le P \le R \tag{27}$$

where R represents the total neurons present in the output layer. MLP considers two weight vectors, one between input and hidden layer, and the other between hidden and output layer. Consider ω' represents the weight vector between input and hidden layers and is given as,

$$\omega' = \left\{ \omega'_{ov} \right\}; \quad 1 \le o \le e; 1 \le U \le V \tag{28}$$

where ω'_{ov} denote the weight between oth neuron and vth hidden neuron. The output of hidden layer is represented as,

$$Z_U = \left[\sum_{o=1}^{e} \omega'_{oV} * I_o \right] G_U \forall I_o = A_o^2 \tag{29}$$

where I_o denote the k^{th} input layer of MLP. The weights between hidden and output layer are given by ω^A and are represented as,

$$\omega^A = \left\{\omega_{oV}^A\right\}; \quad 1 \le U \le V; 1 \le P \le R \tag{30}$$

where ω_{oV}^A represents the weight between oth neuron and vth hidden neuron

Thus, the output vector is computed using the weights ω^A and output of hidden layer and is represented as,

$$O_P = \sum_{U=1}^{V} \omega_{UP}^A + Z_U \tag{31}$$

where ω_{UP}^A represents the weight between the Uth hidden neuron and Pth output neuron and Z_U denotes hidden layer output.

3.4.2 Training of Rider-CSA-based DBN

This section elaborates the training procedure of the proposed Rider-CSA-based DBN classifier. Here, the DBN is composed of RBM layers and MLP layers, which are trained using the proposed Rider-CSA algorithm. Here, the training of Rider-CSA based DBN is done using Rider-CSA so that the suitable weights are chosen optimally. The training process of the proposed Rider-CSA based DBN classifier is elaborated in three layers that are described as follows,

3.4.3 Training of MLP using the proposed Rider-CSA

The training process of MLP is based on Rider-CSA algorithm by giving the training data, which is the output of hidden layer of the second RBM layer, over the network. Examining the data, the network is accustomed repeatedly till the optimal weights are chosen. Moreover, Rider-CSA is employed for computing the optimal weights, which are evaluated using an error function. The proposed Rider-CSA is designed by integrating ROA and CSA. ROA (Binu and Kariyappa 2018) is inspired on the basis of a group of riders, which race towards a target location. Moreover, ROA uses fictional computing and follows the procedure of fictional computing for solving the problems of optimization based on imaginary ideas and thoughts. This algorithm can improve the convergence and avoid the convergence to the local optimum with better performance. CS algorithm (Gandomi et al. 2013) is introduced based on the breed behavior of cuckoos. CS can solve real-world issues having difficult unknown search spaces. Thus, incorporating ROA in CSA yields a better solution with improved performance and controls the rate of convergence. The steps involved in Rider-CSA are explained below:

(i) Initialization

Initialize the weights in a random manner and is represented as,

$$\omega = \{\omega^1, \omega^2, \ldots, \omega', \ldots, \omega^A\} \tag{32}$$

where ω' represents the weight between input and hidden layers, and ω^A represents the weight between hidden and output layer.

(ii) Compute the error

The error is calculated on the basis of difference generated between the desired output and the obtained output and is represented as,

$$E_f = \frac{1}{u}\sum_{q=1}^{u}\left(S_c^q - t_c^q\right)^2 \tag{33}$$

where S_c^q denotes the obtained output and t_c^q represents the desired output.

(iii) Determination of new weights using proposed Rider-CSA

The weight update follows the update procedure of ROA that is based on the group of riders. Thus, the new weights computed using ROA algorithm are derived based on bypass, follower, attacker, and overtaker and the equation for these are given below. The bypass riders follow a common path without tracking the leading rider. The equation of the bypass rider is given as,

$$\omega_{q+1}^B(c,w) = \eta\left[\omega_q(l,w) * \lambda(w) + \omega_q(\vartheta,w) * [1-\lambda(w)]\right] \tag{34}$$

where η represents a random number, l is a random number between 1 and R, ϑ represents a random number ranging from 1 and R, and λ denote a random number ranging between [0, 1]. The follower tends to update the weights based on the location of the leading rider in order to reach the target in a quicker manner, and the equation of the follower is given by,

$$\omega_{q+1}^F(c,v) = \omega^E(E,v) + \left[\mathrm{Cos}(\upsilon_{c,v}^q * \omega^E(E,v) * B_c^q)\right] \tag{35}$$

where v is coordinate selector, ω^E denote the position of leading rider, E specifies the index of leading rider, $\upsilon_{c,v}^q$ indicates the steering angle of cth rider in vth coordinate, and B_c^q is the distance travelled by cth rider. The update based on the overtaker is used in the weight update process for maximizing the success rate by determining the position of overtaker and is given by,

$$\omega_{q+1}^o(c,v) = \omega_q(c,v) + \left[\delta_q^*(c) * \omega^E(E,v)\right] \tag{36}$$

where $\delta_q^*(c)$ represent the direction indicator, $\omega_q(c,v)$ represents the position of cth rider in the vth coordinate. The attacker tends to seize the position of leaders by following the update process of leader, but the attackers update the values in the coordinates rather than the selected values, and thus, the update process of the attacker is given by,

$$\omega_{q+1}(c,w) = \omega^E(E,v) + \left[\mathrm{Cos}\upsilon_{c,v}^q * \omega^E(E,v)\right] + B_c^q \tag{37}$$

The weight update is devised based on levy flight movement and the weight update derived using CS algorithm is expressed as,

$$\omega_{q+1}^{c,w} = \omega_q^{c,w} + \gamma \oplus levy(K) \tag{38}$$

where $\omega_q^{c,w}$ represents the weight at current iteration, γ represents the step size, $\oplus$ represents the entry wise multiplication operator, and $levy(K)$ denotes the levy flight with dimension K.

Assuming the leading position of ROA $\omega^E(E, v)$ be done using ω_{q+1} and thus, the equation is given by,

$$\omega^E(E, v) = \omega_q^{c,w} + \gamma \oplus levy(K) \tag{39}$$

After substituting, the above equation in Eq. (37), the final equation for proposed Rider-CSA is given as,

$$\omega_{q+1}(c, w) = \omega_q^{c,w} + \gamma \oplus levy(K)\left[1 + \mathrm{Cos}v_{c,v}^q\right] + B_c^q \tag{40}$$

(iv) Determination of feasible solution based on fitness function

The solutions are categorized based on the fitness values, and the solution yielding minimum fitness values is ranked as the best solution.

(v) Terminate

The iteration is repeated until maximum number of iterations and terminates upon the generation of the global optimal solution. Algorithm 1 represents the pseudocode of the proposed Deep neural network based Rider-Cuckoo Search Algorithm for plant disease detection.

Algorithm 1: Proposed deep neural network based Rider-Cuckoo Search Algorithm for plant disease detection

```
Proposed DBN-Based Rider-CSA
                    Input: Image Database D
                    Output: plant disease detected images
1                   Begin
2                       Read the images from the database D
3                       Pre-processing
4                       Segmentation based on piFCM
5                       Feature extraction
6                       Classification using the proposed DBN classifier
                            {
7                           Initialize the weights
8                           Compute the error based on equation (33)
9                           Finding the new weights based on equation (40)
                            Repeat until maximum iteration reaches
10                          Return the feasible solution based on fitness func-
                        tion
11                          }
12                  End
```

4 Discussion of results

The results of the proposed method are revealed and the efficiency of the method is proved based on the comparative analysis using accuracy, sensitivity, and specificity metric.

4.1 Experimental setup

The experimentation is performed in MATLAB using Windows 10 OS, with 2 GB RAM and Intel i3 processor. The metrics adapted for analyzing the methods include accuracy, sensitivity, and specificity.

Accuracy The accuracy indicates the accurate detection of plant disease, which is computed as,

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{41}$$

where *TP* denote the true positive *TN* represents the true negative *FP* refers false positive and *FN* is the false negative.

Sensitivity The sensitivity detects the correct portion of plant disease using the sample of inputs identified by the classification results. The sensitivity is also known as True positive rates and is formulated as,

$$Sensitivity = \frac{TP}{TP + FN} \tag{42}$$

Specificity The specificity detects the wrong portion during the plant disease detection. The specificity is also known as false positive rate (FPR) and is formulated as,

$$Specificity = \frac{TN}{TN + FP} \tag{43}$$

4.2 Dataset description

The experimentation is done in real time by collecting input plant images from PlantVillage-Dataset (Plant village dataset 2018) and new plant diseases dataset (https://www.researchgate.net/deref/https%3A%2F%2Fwww.kaggle.com%2Fvipoooool%2Fnew-plant-diseases-dataset%2F). Few sample plant images are considered as the input for performing plant disease identification. Here, two plant images are given as an input, to which the disease needs to be identified. The dataset contains different versions of the data, which are present in the raw directory. The data includes Original RGB images, grayscaled version of the raw images, and segmented RGB images with just the leaf segmented and color corrected.

New plant diseases dataset dataset is recreated using offline augmentation from the original dataset. The original dataset can be found on github repo. This dataset consists of about 87 K rgb images of healthy and diseased crop leaves which is categorized into 38 different classes. The total dataset is divided into 80/20 ratio of training and validation set preserving the directory structure. A new directory containing 33 test images is created later for prediction purpose.

4.3 Experimental results

The experimental results of proposed Rider-CSA method for plant disease detection are deliberated in Fig. 3. We have considered four images for the proposed method from the database, thus the corresponding output images are shown in the Fig. 3. The input image taken for the plant disease identification is shown in Fig. 3a). The input image undergoes segmentation process in order to generate the segments of the image. Thus, the segmented results generated using piFCM technique is depicted in Fig. 3b). Finally, Fig. 3c), portrays the disease detected images of the plants.

4.4 Comparative analysis

The methods employed for the analysis includes: Otsu method (Zhou et al. 2018), exponential Spider Monkey Optimization (ESMO) (Kumar et al. 2018), and BRBFNN (Chouhan et al. 2018). The existing methods and the proposed Rider-CSA algorithm are taken for comparison and analyzed based on performance metrics, which involves accuracy, sensitivity, and specificity.

4.4.1 Analysis using dataset 1

(a) Analysis on the basis of training data percentage

Figure 4 illustrates the analysis of methodologies based on training data percentage using accuracy, sensitivity, and specificity measures. The analysis of existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA based on accuracy metric is illustrated in Fig. 4a). When the training data percentage is 50, then the corresponding accuracy values computed by existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.769, 0.752, 0.614, and 0.798, respectively. Likewise, for 90% training data, the corresponding accuracy values computed by existing Otsu method, ESMO, BRBFNN, and proposed Rider-CSA are 0.794, 0.869, 0.798, and 0.877, respectively. The analysis of existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA based on sensitivity metric is illustrated in Fig. 4b). When the training data percentage is 50, then the corresponding sensitivity values computed by existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.569, 0.647, 0.592, and 0.592, respectively. Likewise, for 90% training data, the corresponding sensitivity values computed by existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.655, 0.789, 0.662, and 0.826, respectively. The analysis of existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA based on specificity metric is illustrated in Fig. 4c). When the training data percentage is 50, then the corresponding specificity values computed by existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.730, 0.773, 0.520, and 0.771, respectively. Likewise for 90% training data, the corresponding specificity values computed by existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.830, 0.831, 0.803, and 0.855, respectively. Based on the results, it is noted that the proposed Rider-CSA performs better than other existing methods in terms of accuracy, sensitivity and specificity.

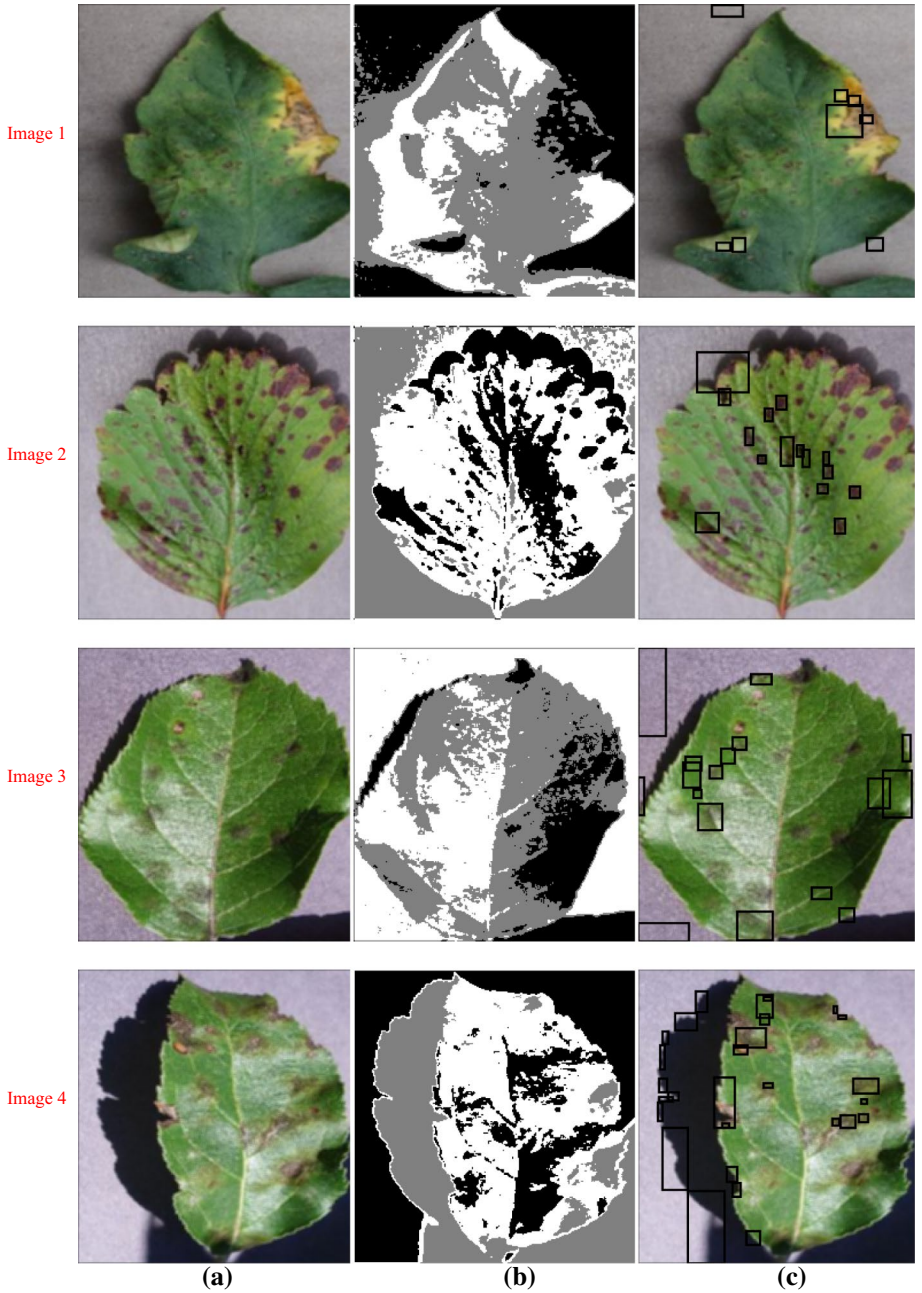

Fig. 3 Experimental results, **a** input image, **b** segmented image, **c** plant disease detected image

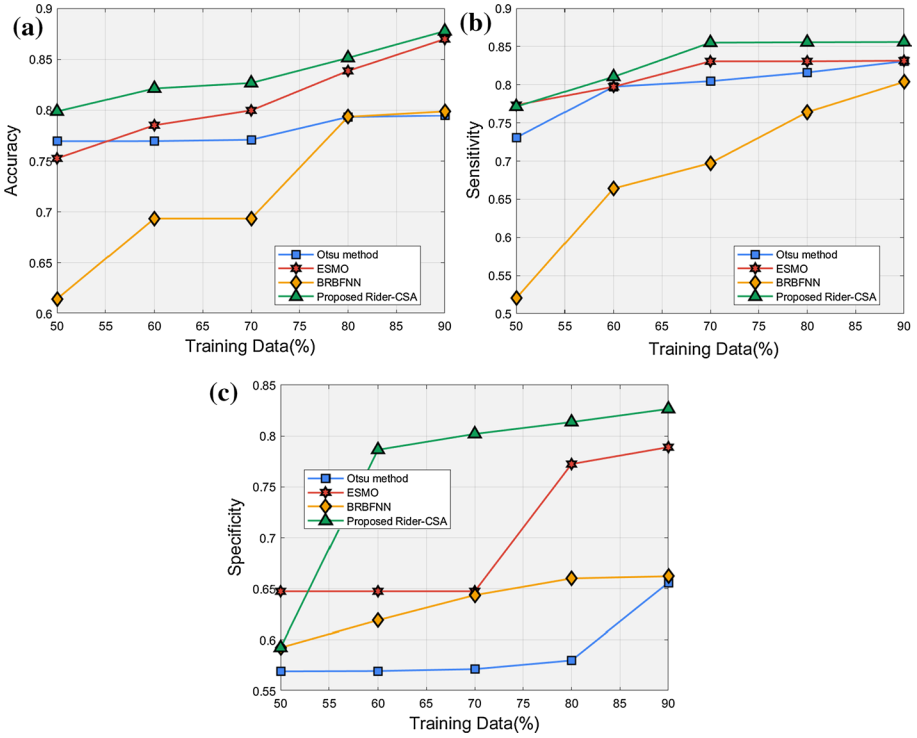

Fig. 4 Comparative analysis based on training data percentage using **a** accuracy, **b** sensitivity, **c** specificity

(b) Analysis based on K-Fold

Figure 5 illustrates the analysis of methodologies based on K-value using accuracy, sensitivity, and specificity measures. The analysis of existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA based on accuracy metric is illustrated in Fig. 5a). When the K-value is 6, then the corresponding accuracy values computed by existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.646, 0.534, 0.568, and 0.676, respectively. Likewise for K-value 10, the corresponding accuracy values computed by existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.682, 0.696, 0.682, and 0.812, respectively. The analysis of existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA based on sensitivity metric is illustrated in Fig. 5b). When the K-value is 6, then the corresponding sensitivity values computed by existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.791, 0.685, 0.685, 0.833, respectively. Likewise for K-value 10, the corresponding sensitivity values computed by existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.841, 0.851, 0.834, and 0.862, respectively. The analysis of existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA based on specificity metric is illustrated in Fig. 5c). When the K-value is 6, then the corresponding specificity values computed by existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.568, 0.572, 0.573, and 0.641, respectively. Likewise for K-value 10, the corresponding specificity values computed by existing Otsu method, ESMO, BRBFNN

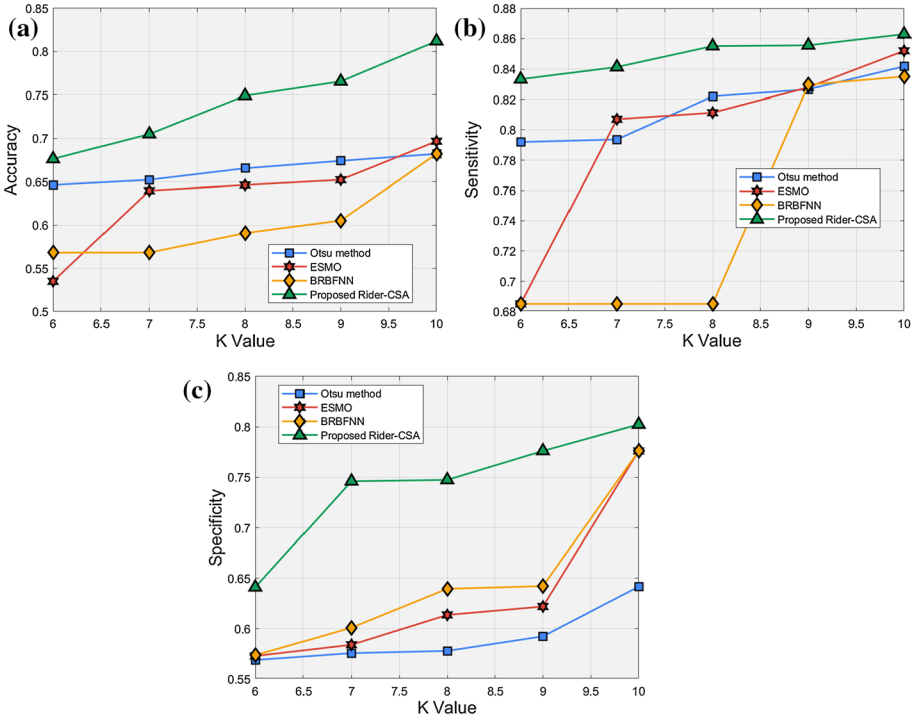

Fig. 5 Comparative analysis based on training data percentage using **a** accuracy, **b** sensitivity, **c** specificity

and proposed Rider-CSA are 0.641, 0.776, 0.776, and 0.802, respectively. Based on the analysis result, it is noted that the proposed Rider-CSA performs better than other existing methods in terms of accuracy, sensitivity and specificity.

4.4.2 Analysis using dataset 2

(a) Analysis on the basis of training data percentage

Figure 6 illustrates the analysis of methodologies based on training data percentage using accuracy, sensitivity, and specificity measures. The analysis of existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA based on accuracy metric is illustrated in Fig. 6a). When the training data percentage is 50, then the corresponding accuracy values computed by existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.578, 0.620, 0.582, and 0.581, respectively. Similarly, the corresponding values of accuracy for an existing Otsu method, ESMO, BRBFNN, and proposed Rider-CSA with 90% training data are 0.792, 0.903, 0.896, and 0.919, respectively. Thus, it shows that the accuracy for the proposed method of every dataset is increased. The analysis of existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA based on sensitivity metric is illustrated in Fig. 6b). When the training data is 50%, then the corresponding values of sensitivity are computed by existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA

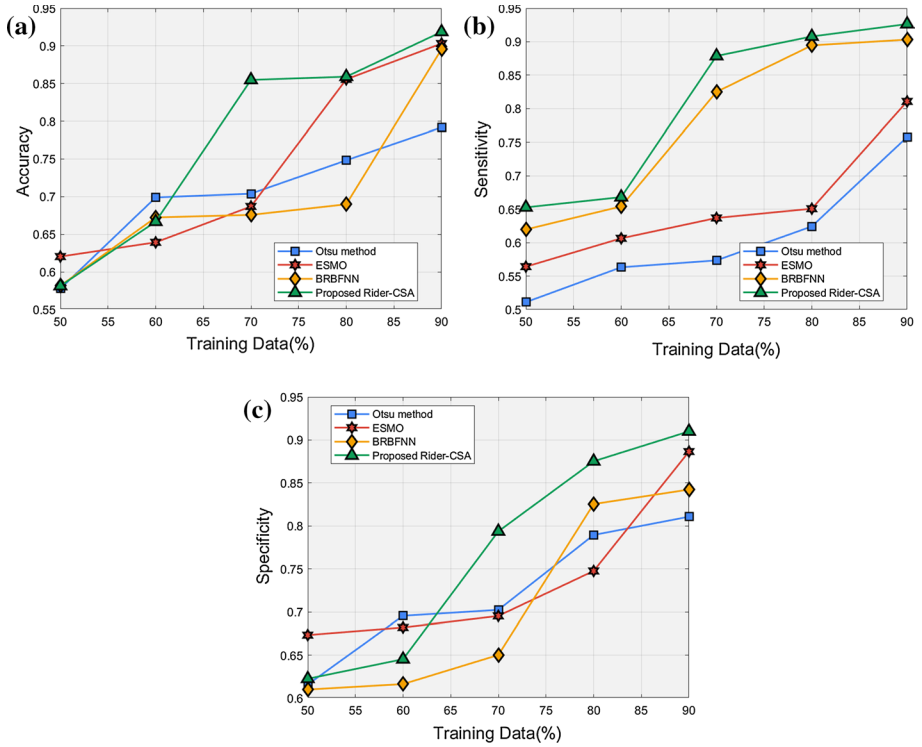

Fig. 6 Comparative analysis based on training data percentage using **a** accuracy, **b** sensitivity, **c** specificity

as 0.511, 0.564, 0.619, and 0.653, respectively. Similarly, the values of the corresponding sensitivity for every training data (60%, 70%, 80%, and 90%) are increased in the proposed method than the existing method. Based on specificity metric, the existing method analysis of Otsu, ESMO, BRBFNN and proposed Rider-CSA is illustrated in Fig. 6c). When the training data 50%, then the corresponding specificity values computed for the existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.616, 0.673, 0.61, and 0.623, respectively. Similarly, for the 90% of training data, the corresponding specificity values are computed by the existing method of Otsu, ESMO, BRBFNN and proposed Rider-CSA are 0.811, 0.887, 0.842, and 0.91, respectively. Based on the results, it is noted that the proposed Rider-CSA performs better than other existing methods in terms of accuracy, sensitivity and specificity.

(b) Analysis based on K-Fold

Figure 7 illustrates the analysis of methodologies based on K-value using accuracy, sensitivity, and specificity measures. The analysis of existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA based on accuracy metric is illustrated in Fig. 7a). When the value of K is 6, then the corresponding accuracy values are computed by existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.601, 0.665, 0.64, and 0.729, respectively. Similarly, when the value of K is 7, 8, 9, and 10, then the corresponding

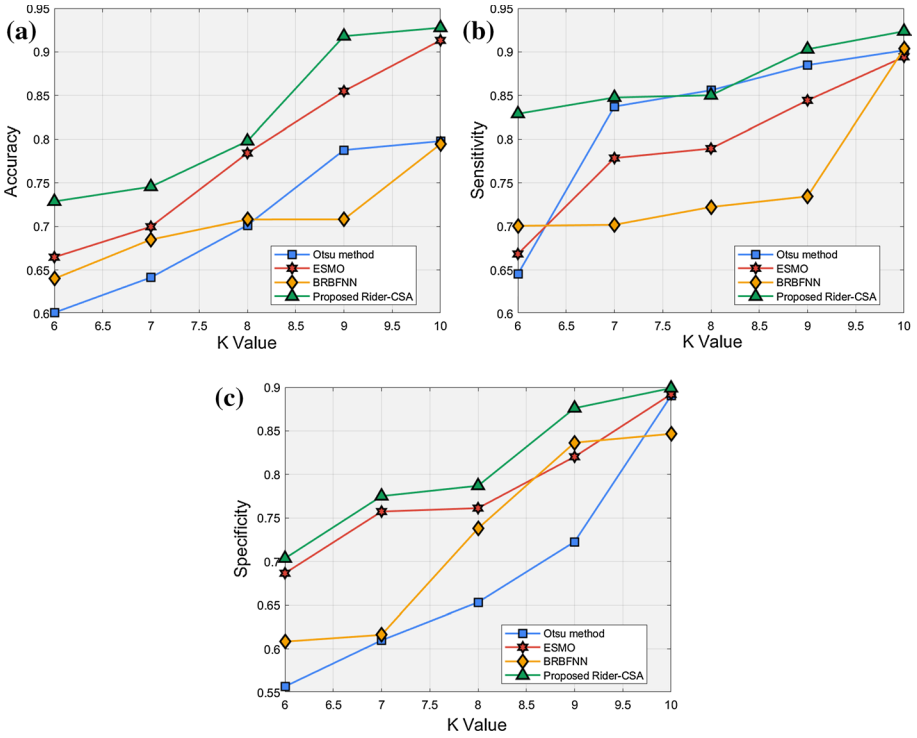

Fig. 7 Comparative analysis based on training data percentage using **a** accuracy, **b** sensitivity, **c** specificity

values of accuracy is improved for the proposed Rider-CSA method. The analysis of existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA based on the sensitivity metric is illustrated in Fig. 7b). When the K-value is 6, then the corresponding sensitivity values computed by the existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.645, 0.668, 0.701, 0.829, respectively. Likewise, when K = 10, then the corresponding sensitivity values computed by the existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.902, 0.894, 0.904, and 0.924, respectively. The analysis of the existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA based on specificity metric is illustrated in Fig. 7c). When the value of K is 6, then the proposed Rider-CSA has the high specificity as 0.704 than the existing method of Otsu, ESMO, BRBFNN as 0.557, 0.687, and 0.608 respectively. Likewise for K-value 10, the corresponding specificity values computed by the existing Otsu method, ESMO, BRBFNN and proposed Rider-CSA are 0.890, 0.892, 0.847, and 0.899, respectively. Based on the analysis result, it is noted that the proposed Rider-CSA performs better than other existing methods in terms of accuracy, sensitivity, and specificity.

4.5 Comparative discussion

Table 1 elaborates the analysis of existing methodologies based on accuracy, sensitivity, and specificity based on training data and K-fold values. The maximal accuracy of proposed Rider-CSA is 0.877, whereas the existing Otsu method, ESMO, BRBFNN acquired

Table 1 Comparative discussion

Metrics	Analysis	Otsu method	ESMO	BRBFNN	Proposed Rider-CSA
Accuracy	Training data	0.794	0.869	0.798	0.877
	K-Value	0.682	0.696	0.682	0.812
Specificity	Training data	0.655	0.789	0.662	0.826
	K-Value	0.641	0.776	0.776	0.802
Sensitivity	Training data	0.830	0.831	0.803	0.855
	K-Value	0.841	0.851	0.834	0.862

the accuracy values of 0.794, 0.869, and 0.798, respectively. The maximal specificity of proposed Rider-CSA is 0.826, whereas the specificity measured by the existing Otsu method, ESMO, BRBFNN is 0.655, 0.789, and 0.662, respectively. The maximal sensitivity of proposed Rider-CSA is 0.862, whereas the sensitivity measured by the existing Otsu method, ESMO, BRBFNN is 0.841, 0.851, and 0.834, respectively.

5 Conclusion

In this paper, an effective image processing method is devised for determining the plant diseases. Initially, the input image is given to the pre-processing phase for eliminating the noise and artifacts present in the image. After obtaining the pre-processed image, it is subjected to the segmentation phase for obtaining the segments. The segmentation is carried out using piFCM. Each segment undergoes a feature extraction phase in which the texture features is extracted using information gain, HOG, and entropy. The obtained texture features is subjected to the classification phase, which uses the DBN for effective plant disease detection. Here, the proposed Rider-CSA is employed for training the DBN. The proposed Rider-CSA is designed by integrating the ROA and CSA. The proposed Rider-CSA-based DBN performs the classification process to detect plant disease. The implementation of the proposed approach is carried out using the Plant village database, based on the metrics, namely sensitivity, accuracy, and specificity. The results proved that the proposed Rider-CSA outperformed other existing methods with maximal accuracy of 0.877, sensitivity of 0.862, and specificity of 0.877, respectively.

References

Aher KV, Waykar SB (2014) A survey on feature based image retrieval. Int J Adv Res Comput Sci Softw Eng 4:241–245

Akhter MS, Akanda AM, Kobayashi K, Jain RK, Mandal B (2019) Plant virus diseases and their management in Bangladesh. Crop Prot 118:57–65

Barbedo JGA (2016) A review on the main challenges in automatic plant disease identification based on visible range images. Bio Syst Eng 144:52–60

Barbedo JGA (2019) Plant disease identification from individual lesions and spots using deep learning. Bio Syst Eng 180:96–107

Bharill N, Patel OP, Tiwari A, Li DL, Mohanthy M, Kaiwartya OP, Prasad M (2019a) A generalized enhanced quantum fuzzy approach for efficient data clustering. IEEE Access 7:50347–50361

Bharill N, Patel OP, Tiwari A, Prasad M (2019b) A novel quantum-inspired fuzzy based neural network for data classification. IEEE Trans Emerg Top Comput. https://doi.org/10.1109/TETC.2019.2901272

Binu D, Kariyappa BS (2018) RideNN: a new rider optimization algorithm-based neural network for fault diagnosis in analog circuits. IEEE Trans Instrum Meas 99:1–25

Cheng EJ, Chou KP, Rajora S, Jin BH, Tanveer M, Lin CT, Young KY, Lin WC, Prasad M (2019) Deep sparse representation classifier for facial recognition and detection system. Pattern Recognit Lett 125:71–77

Chouhan SS, Kaul A, Singh UP, Jain S (2018) Bacterial foraging optimization based radial basis function neural network (BRBFNN) for identification and classification of plant leaf diseases: an automatic approach towards plant pathology. IEEE Access 6:8852–8863

Chuanlei Z, Shanwen Z, Jucheng Y, Yancui S, Jia C (2017) Apple leaf disease identification using genetic algorithm and correlation based feature selection method. Int J Agric Biol Eng 10(2):74–83

Daga B, Bhute A, Ghatol A (2011) Implementation of parallel image processing using NVIDIA GPU framework. In: Proceedings of the international conference on advances in computing, communication and control. Springer, Berlin, Heidelberg, pp 457–464

Damer N, Führer B (2012) Ear recognition using multi-scale histogram of oriented gradients. In: Proceedings of eighth international conference on intelligent information hiding and multimedia signal processing. IEEE, pp 21–24

Deepa S (2017) Steganalysis on images using SVM with selected hybrid features of Gini index feature selection algorithm. Int J Adv Res Comput Sci 8(5):305

Dong X, Shen J (2018) Triplet loss in siamese network for object tracking. In: The European conference on computer vision (ECCV). pp 459–474

Dong X, Shen J, Shao L, Van Gool L (2016) Sub-markov random walk for image segmentation. IEEE Trans Image Process 25(2):516–527

Dong X, Shen J, Wang W, Liu Y, Shao L, Porikli F (2018) Hyperparameter optimization for tracking with continuous deep Q-learning. In: 2018 IEEE/CVF conference on computer vision and pattern recognition

Dong X, Shen J, Wu D, Guo K, Jin X, Porikli F (2019) Quadruplet network with one-shot learning for fast visual object tracking. IEEE Trans Image Process 28(7):3516–3527

Francis J, Anoop BK (2016) Identification of leaf diseases in pepper plants using soft computing techniques. In: IEEE conference on emerging devices and smart systems (ICEDSS). pp 168–173

Gandomi AH, Yang XS, Alavi AH (2013) Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. Eng Comput 29(1):17–35

Ganesan P, Sajiv G, Leo LM (2017) CIELuv color space for identification and segmentation of disease affected plant leaves using fuzzy based approach. In: Proceedings of international conference on science technology engineering and management (ICONSTEM). pp 889–894

Gangappa M, Mai KC (2019) "Enhanced crow search optimization algorithm and hybrid NN-CNN classifiers for classification of land cover images. Multimedia Res 2(3):12–22

Goceri E (2011) Automatic kidney segmentation using Gaussian mixture model on MRI sequences. from book Electrical Power Systems and Computers: Selected Papers from the 2011 International Conference on Electric and Electronics (EEIC 2011) in Nanchang, China on June 20–22, 2011, Vol 3, pp 23-29

Goceri E (2013) A comparative evaluation for liver segmentation from spir images and a novel level set method using signed pressure force function. Ph.D. thesis, Izmir Institute of Technology, Electronics and Communication Engineering, Izmir

Göçeri E (2016) Fully automated liver segmentation using Sobolev gradient-based level set evolution. Numer Methods Biomed Eng 32(11):2765. https://doi.org/10.1002/cnm.2765

Goceri N, Goceri E (2015) A neural network based kidney segmentation from MR images. In: IEEE 14th international conference on machine learning and applications (ICMLA). Miami, FL, USA

Goceri E, Gooya A (2018) On the importance of batch size for deep learning. In: International conference on mathematics (ICOMATH2018), an Istanbul meeting for world mathematicians, minisymposium on approximation theory and minisymposium on math education, 3–6 July 2018, Istanbul, Turkey

Göçeri E, Ünlü MZ, Dicle O (2013) A Comparative performance evaluation of various approaches for liver segmentation from SPIR images. Turk J Electr Eng Comput Sci (TJEECS) 23(3):743–768

Göçeri E, Gürcan MN, Diclec O (2014) Fully automated liver segmentation from SPIR image series. Comput Biol Med 53:265–278

Guettari N, Capelle-Laize AS, Carre P (2016) Blind image steganalysis based on evidential k-nearest neighbors. In: Image processing, in proceedings of international conference (ICIP). pp 2742–2746

Heil J, Häring V, Marschner B, Stumpe B (2019) Advantages of fuzzy k-means over k-means clustering in the classification of diffuse reflectance soil "spectra: a case study with West African soils. Geoderma 337:11–21
Hinton GE, Osindero S, Teh Y (2006) A fast learning algorithm for deep belief nets. Neural Comput 18:1527–1554
Kamble JK (2018) Plant disease detector. In: Proceedings of international conference on advances in communication and computing technology (ICACCT). pp 97–101
Kodovsky J, Fridrich J, Holub V (2012) Ensemble classifiers for steganalysis of digital media. IEEE Trans Inf Forensics Secur 7(2):432–444
Kumar S, Sharma B, Sharma VK, Sharma H, Bansal JC (2018) Plant leaf disease identification using exponential spider monkey optimization. Sustain Comput Inform Syst, Available online 27 October 2018 (in press)
Li X, Deng Y, Ding L (2008) Study on precision agriculture monitoring framework based on WSN. In: Proceedings of international conference on anti-counterfeiting, security and identification. pp 182–185
Lu Y, Yi S, Zeng N, Liu Y, Zhang Y (2017) Identification of rice diseases using deep convolutional neural networks. Neurocomputing 267:378–384
Menaga D, Revathi S (2018) Least lion optimisation algorithm (LLOA) based secret key generation for privacy preserving association rule hiding. IET Inf Secur 12(4):332–340
New plant diseases dataset. Available at, https://www.researchgate.net/deref/https%3A%2F%2Fwww.kaggle.com%2Fvipoooool%2Fnew-plant-diseases-dataset%2F
Patil S, Chandavale A (2015) A survey on methods of plant disease detection. Int J Sci Res 4(2):1392–1396
Plant village dataset. https://github.com/spMohanty/PlantVillage-Dataset. Accessed on December 2018
Prasad M, Rajora S, Gupta D, Daraghmi YA, Daraghmi E, Yadav P, Tiwari P, Saxena A (2018) Fusion based En-FEC transfer learning approach for automobile parts recognition system. IEEE SSCI
Rajora S, Vishwakarma DK, Singh K, Prasad K (2018) CSgI: a deep learning based approach for marijuana leaves strain classification. IEEE IEMCON
Ramezani M, Ghaemmaghami S (2010) Towards genetic feature selection in image steganalysis. In: Proceedings of consumer communications and networking conference (CCNC). pp. 1–4
Ratre A (2019) Taylor series based compressive approach and Firefly support vector neural network for tracking and anomaly detection in crowded videos. J Eng Res 7(4):115–137
Roobaert D, Karakoulas G, Chawla NV (2006) Information gain, correlation and support vector machines. In: Guyon I, Nikravesh M, Gunn S, Zadeh LA (eds) Feature extraction. Springer, Berlin, pp 463–470
Saxena A, Prasad M, Gupta A, Bharill N, Pate OP, Er MJ, Ding W, Lin CT (2017) A review of clustering techniques and developments. Neurocomputing 267:664–681
Sheikhan M, Pezhmanpour M, Moin MS (2012) Improved contourlet-based steganalysis using binary particle swarm optimization and radial basis neural networks. Neural Comput Appl 21(7):1717–1728
Shen J, Du Y, Wang W, Li X (2014) Lazy random walks for superpixel segmentation. IEEE Trans Image Process 23(4):1451–1462
Shen J, Hao X, Liang Z, Liu Y, Wang W, Shao L (2016) Real-time superpixel segmentation by DBSCAN clustering algorithm. IEEE Trans Image Process 25(12):5933–5942
Shen J, Peng J, Shao L (2018) Submodular trajectories for better motion segmentation in videos. IEEE Trans Image Process 27(6):2688–2700
Shen J, Dong X, Peng J, Jin X, Shao L, Porikli F (2019) Submodular function optimization for motion clustering and image segmentation. IEEE Trans Neural Netw Learn Syst 30(9):2637–2649
Singh KK (2018) An artificial intelligence and cloud based collaborative platform for plant disease identification, tracking and forecasting for farmers. In: Proceedings of international conference on cloud computing in emerging markets (CCEM). pp 49–56
Sui B (2013) Information gain feature selection based on feature interactions. ETD Collection
Tetila EC, Machado BB, de Souza Belete NA, Guimarães DA, Pistori H (2017) Identification of soybean foliar diseases using unmanned aerial vehicle images. IEEE Geosci Remote Sens Lett 14(12):2190–2194
Thomas R, Rangachar MJS (2018) Hybrid Optimization based DBN for Face Recognition using Low-Resolution Images. Multimedia Research 1(1):33–43
Wang W, Shen J (2018) Deep visual attention prediction. IEEE Trans Image Process 27(5):2368–2378
Wang P, Fu H, Zhang K (2018a) A pixel-level entropy-weighted image fusion algorithm based on bidimensional ensemble empirical mode decomposition. Int J Distrib Sens Netw 14(12):1–16

Wang W, Shen J, Shao L (2018b) Video salient object detection via fully convolutional networks. IEEE Trans Image Process 27(1):38–49
Wang W, Shen J, Ling H (2019) A deep network solution for attention and aesthetics aware photo cropping. IEEE Trans Pattern Anal Mach Intell 41(7):1531–1544
Wu J, Wu Z, Cao J, Liu H, Chen G, Zhang Y (2017) Fuzzy consensus clustering with applications on big data. IEEE Trans Fuzzy Syst 25(6):1430–1445
Zhang X, Qiao Y, Meng F, Fan C, Zhang M (2018) Identification of maize leaf diseases using improved deep convolutional neural networks. IEEE Access 6:30370–30377
Zhou C, Yang G, Liang D, Yang X, Xu B (2018) An integrated skeleton extraction and pruning method for spatial recognition of maize seedlings in MGV and UAV remote images. IEEE Trans Geosci Remote Sens 56(8):4618–4632