A brief study on rice diseases recognition and image classification: fusion deep belief network and S-particle swarm optimization algorithm

Miryabbelli Jayaram[1], Gudikandhula Kalpana[2], Subba Reddy Borra[3], Battu Durga Bhavani[3]
[1]Department of Computer Science Engineering (Data Science), Sreyas Institute of Engineering and Technology, Hyderabad, India
[2]Department of Computer Science and Engineering (AI and ML), Malla Reddy Engineering College for Women (UGC-Autonomous), Hyderabad, India
[3]Department of Information Technology, Malla Reddy Engineering College for Women (UGC-Autonomous), Hyderabad, India

Article Info

Article history:

Received Nov 6, 2022
Revised Feb 28, 2023
Accepted Mar 9, 2023

Keywords:

Deep belief networks
Image classification
Image recognition
Rice diseases
S-particle swam optimization algorithm

ABSTRACT

In the regions of southern Andhra Pradesh, rice brown spot, rice blast, and rice sheath blight have emerged as the most prevalent diseases. The goal of this research is to increase the precision and effectiveness of disease diagnosis by proposing a framework for the automated recognition and classification of rice diseases. Therefore, this work proposes a hybrid approach with multiple stages. Initially, the region of interest (ROI) is extracted from the dataset and test images. Then, the multiple features are extracted, such as color-moment-based features, grey-level cooccurrence matrix (GLCM)-based texture, and shape features. Then, the S-particle swarm optimization (SPSO) model selects the best features from the extracted features. Moreover, the deep belief network (DBN) model trained by SPSO is based on optimal features, which classify the different types of rice diseases. The SPSO algorithm also optimized the losses generated in the DBN model. The suggested model achieves a hit rate of 94.85% and an accuracy of 97.48% with the 10-fold cross-validation approach. The traditional machine learning (ML) model is significantly less accurate than the area under the receiver operating characteristic curve (AUC), which has an accuracy of 97.48%.

Corresponding Author:

Miryabbelli Jayaram
Department of Computer Science Engineering (Data Science), Sreyas Institute of Engineering and Technology
Hyderabad, India
Email: drjayaram2022@gmail.com

1. INTRODUCTION

Among the most significant food crops for our population is rice. The production of the rice planting business has been rising steadily in recent years, but there are still certain negative elements, including rice diseases, that are reducing its yield. In the frigid regions of northern China, rice brown spots, sheath blight, and blast are currently the most prevalent diseases [1], [2]. It can happen at any time during the rice plant's growth cycle, which has a significant influence on the rice's quality and yield and, in extreme situations, can result in no output. This will not only have a negative influence on farmer income but also negatively impact China's fiscal revenue and food security. Now, the methods for identifying rice diseases primarily rely on farmers' judgment, checking disease books or doing an Internet search, consulting agricultural technicians, or asking a plant expert for assistance. Human eyes are not very good at identification, and recognition is much more individualized. A lapse in judgement prevents the proper identification of obstacles to achieving diseases. The number of books is updated at a sluggish rate, and due to the volume of information on the internet, it is accuracy cannot be guaranteed. Speaking with plant and agricultural technicians, a significant amount of time, money, and resources will be devoured. Much money and time will be spent, and it is simple. to pass up the ideal chance for treating diseases. Therefore, developing a new, effective method for detecting rice disease a technique to substitute the prevalent human eye recognition technique is essential to resolving the issue of rice disease identification, which has significant study consequences.

Though no study has been conducted till date that involves the combination of the S-particle swarm optimization (SPSO) and deep belief network (DBN) because the SPSO and DBN are not compatible, rice disease identification has a lot of potential. According to traditional identification’s flaws and limitations, we proposed a DBN-based method and SPSO for detecting rice brown spot, sheath blight, and blast. Images from the database of rice diseases were used by the DBN. Models such as SPSO are trained in two phases. First and foremost, deep belief network-restricted Boltzmann machine (RBMs) are used to build the model. The gradient's descent comes next. The DBN is trained using the SPSO method. As a categorizer, the SPSO and DBN are then assessed with 10-fold cross-validation.

These are the primary advances made in this study: SPSO and DBN approaches are used to precisely extract the rice disease spot’s edge in a new framework for the usual detection of rice diseases that addresses issues with the rice disease spot’s image segmentation. The suggested DBN and SPSO could detect the area of interest that is acquired by preprocessing the images of rice diseases by learning disease spot characteristics. In terms of feature characteristics, segmentation, and peak signal-to-noise ratio (SNR) experimental findings demonstrate the great accuracy of this approach. Because of this study, the field of rice disease identification is better integrated than it is with the traditional diagnostic technique for diagnosing rice diseases, and DBN and SPSO algorithms are combined for rice disease using machine learning (ML) technology. The initial identification significantly enhances the timeliness and precision of rice disease management and has practical relevance.

2. METHOD

The two strategies for classifying data are supervised learning classification and unsupervised learning classification. This study aims to extract features or traits from visual data. By using a supervised learning system, the information is used to categorize the images. Learning to recognize an image based on data and labels, including data mining, the traits of image classification provide us with knowledge. To determine the type of data, we can apply a classification algorithm to fresh images. Specifically, we extract the numerical image’s vector of features before using the ML approach to apply categorized information.

New techniques and approaches for quick and precise pattern recognition have emerged in recent years, thanks to the quick growth of computer vision, digital image processing, and pattern recognition technologies. Because of the rapid advancement of digital image pattern recognition technologies, processing, and computer vision, plant disease classification, diagnosis, and nondestructive detection, these sophisticated diagnostic techniques have had significant success in both theoretical study and real-world implementation [3]–[5]. However, the effectiveness of diagnosis is often subpar due to the limitations of theoretical analysis, the requirement for specialized talents, and the necessity for extensive experience and knowledge during training.

A probabilistic generation model called DBN is used during this period and is made up of numerous constrained RBMs [6]–[8]. Layer-by-layer stacking of the RBMs allows them to extract characteristics from the original data, which allows us to get certain high-level depictions of the original data [9]. This deep learning (DL) model’s central idea is to use a layer-by-layer greedy learning method to optimize the link deep neural network’s weight. Initially, to identify the rice disease characteristics, the training method is unsupervised layer-by-layer. The supervised learning process is reversed after adding the relevant classifier.

By switching to PSO, which achieves a balance between the global and local search methods by introducing a mode-dependent velocity update, we produced better results by using the SPSO technique to convert the optimization issue with limitations into an optimization issue without constraints. Numerous researchers have used the DBN in recent years in a variety of application fields, including face, speech, and image recognition, as well as the classification of plant leaves. A fundamental concept for feature extraction and following image identification for original images is presented by this multi-layer network design. The DBN has gained popularity [10].

Agriculture is the backbone of most countries. The early prediction of disease in rice plants can increase productivity and be helpful to farmers. However, the conventional ML models were focused on basic feature extraction. Moreover, the existing methods did not implement any feature selection mechanism. So, this work implemented a hybrid mechanism with multiple stages. Figure 1 shows the proposed rice disease image recognition flowchart. Initially, the region of interest (ROI) is extracted from the dataset and test images. Then, the multiple features are extracted, such as color, texture, and shape, using ROI properties. Here, the color features are extracted in first order, second order, and third order using the mean proposition. Here, the texture features are extracted using the grey-level cooccurrence matrix (GLCM). The extracted texture features are energy, or angular second order moment (ASM), entropy, contrast, and inverse difference momentum (IDM). Then, the features are combined using feature fusion. Further, the SPSO operation is introduced to select the optimal features. Finally, the DBN model is trained with the dataset features, which store the feature memory. In addition, the DBN testing model is used to perform image testing, which resulted in a disease recognition operation. The proposed SPSO-DBN model recognizes rice blast, sheath blight, and brown spot-based disease classes, as well as healthy rice image recognition.

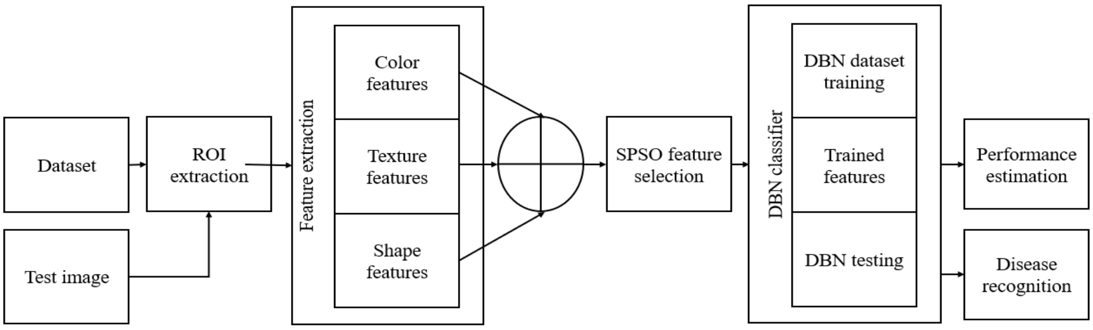


Figure 1. Rice disease image recognition flowchart for DBN and SPSO

2.1. Dataset

The rice disease images were taken on the rice trail domain of “The Heilongjiang Academy of Agricultural Reclamation Sciences,” China, using a vivo Y33T smartphone (12 million pixels, F/2.8 telephoto dual-lens camera, F/1.8 wide-angle camera, and iOS 10.1 operating system). The smartphone is mounted on a bracket, set to automatic exposure mode, and taken at a reasonable distance to capture an image of the infected rice leaves. The phone takes images with a 3,024×4,032-pixel resolution. Figure 2 displays the sample images for each class with rice blast image in Figure 2(a), the sheath blast, brown spot images in Figures 2(b) and 2(c), respectively. Figure 2(d) illustrates the healthy rice plant image. Table 1 contains a detailed description of the dataset.

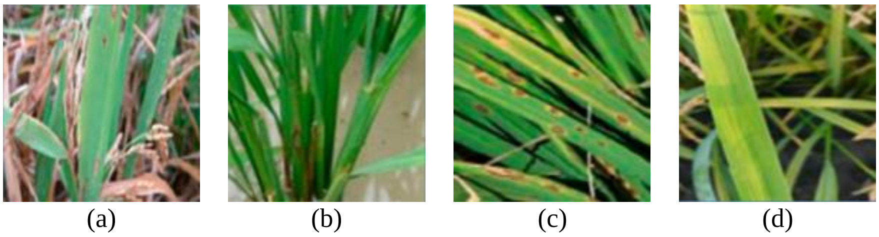


Figure 2. Rice diseases and healthy image, (a) rice blast, (b) sheath blight, (c) brown spot, and (d) healthy rice image

Table 1. Dataset description

Class Name	Testing images	Training images	Testing images
rice blast	2,000	1,500	500
sheath blight	2,000	1,500	500
brown spot	2,000	1,500	500
Health	2,000	1,500	500
Total	8,000	6,000	2,000

Three rice diseases were captured with 2,000 images for each class. As a result, 8,000 sample images were gathered in total, of which 2,000 were free of brown spot, sheath blight, and rice blast. Every image is saved in the jpeg format and compressed to a standard size of 400×300. Further, 75% of the samples are selected at random as training sets from the dataset, and the remaining 25% are chosen as testing sets. So, each class contains 15 numbers of training images and 500 numbers of testing images. In addition, a total of 6,000 training images are considered, along with 2,000 testing images.

2.2. Feature extraction

The goal of feature extraction is to reduce the number of resources necessary to describe a huge set of data. One of the primary issues with analyzing complicated data is the large number of variables involved. A large number of variables necessitates a huge amount of memory and compute capacity, and it may also lead a classification method to overfit to training examples and generalize poorly to new samples. The process of converting raw data into numerical features that may be processed while retaining the information in the original data set is referred to as feature extraction [11].

2.2.1. Color feature extraction

The image's orientation, size, and angle have no bearing on the color feature. It is a typical visual aspect of the image and is frequently utilized in image recognition. The red green blue (RGB), HSV, and lab color spaces are employed in this research to extract the color eigenvalues of rice disease. To acquire the energy, mean, and color set variance in HSV space, along with other color properties, the image of rice disease with disease spots is first converted from the "RGB" color space to the "HSV" color space. The color characteristic is then expanded upon by obtaining the third order of color moments corresponding to every element in lab space: RGB and HSV [12], [13]. The color moment (mean) of the first order depicts the image's average brightness. It is described (1).

$$\mu_j = \frac{1}{N}\sum_{j=1}^{N} P_{i,j} \tag{1}$$

The color moment (variance) of second order illustrates the discreteness of the grey level distribution in the image. It is expressed as underneath (2).

$$\sigma_i = \left[\frac{1}{N}\sum_{j=0}^{N}(p_{i,j} - \mu_j)^2\right]^{1/2} \tag{2}$$

The color moment of third order describes the asymmetry and skewness of the color components. It is described as (3):

$$s_i = \left[\frac{1}{N}\sum_{j=0}^{N}(p_{i,j} - \mu_j)^3\right]^{1/3} \tag{3}$$

where $p_{i,j}$ signifies the j^{th} pixel of the color image's i^{th} color component, and N denotes the image's pixels number.

2.2.2. Texture feature extraction

The order and organization of an object's parts can be reflected in its texture. It serves as a crucial visual component in the image description. It possesses rotation invariance and can compute the statistical characteristics of several pixel regions [14]–[18]. The gray-level co-occurrence matrix (GLCM) method, which is the most popular technique for extracting texture features. It may efficiently depict how an image changes at a specific distance and in a specific direction. Therefore, in this study, texture features are extracted using the co-occurrence matrix of the grey level. Figure 3 displays the texture-feature extraction effect. Formula (4) can be used to represent the GLCM (4).

$$p(a_1, a_2) = \frac{\neq\{[(x_1,y_1),(x_2,y_2)]\in S\}|f(x_1,y_1)=a_1 \& f(x_2,y_2)=a_2\}}{\#S} y \tag{4}$$

In this method, $f(x, y)$ denotes the image itself, S indicates the pixel pair is set with certain spatial connections in the image, and the expression on the equation's numerator right side indicates the number of pixel pairs with a spatial connection and a grey value of $\mathrm{a}_1, \mathrm{a}_2$, while the expression on the denominator indicates the overall pixel pair's number (# indicates the number of statistical elements).

The following are the recovered common feature parameters based on the GLCM:

a. ASM: It determines the texture's thickness and the homogeneity of the grayscale in the image.

$$ASM = \sum_{i=1}^{N}\sum_{j=1}^{N} p(i,j)^2 \tag{5}$$

b. Entropy: It exhibits neither uniformity nor complexity in the texture of the image. It is a measurement of the amount of texture information included in an image.

$$Entropy = \sum_{i=1}^{N} \sum_{j=1}^{N} p(i,j). \log p(i,j) \quad (6)$$

c. Contrast: It represents the image's clarity and gauges the depth of its textural groove.

$$Contrast = \sum_{i=1}^{N} \sum_{j-1}^{N} (i-j)^2 p(i,j) \quad (7)$$

d. IDM: It indicates the regularity and clarity of the image's texture as well as its local variation.

$$IDM = \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{p(i,j)}{1+(i-j)^2} \quad (8)$$

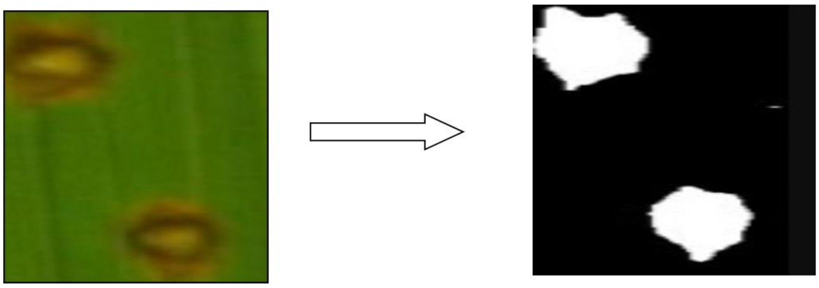


Figure 3. Texture feature extraction

2.2.3. Shape feature extraction

This study used the standard morphological criteria of the lesions' area, circularity, complexity, and number. From there, it came up with two new physical traits, such as the number of lesions, and suggested a ratio between the size and number of lesions. The "Lesion" area calculation formula is as (9):

$$S = \sum_{x=1}^{n} \sum_{y=1}^{m} f(x,y) \quad (9)$$

here, the binary image function is denoted by $f(x,y)$. Circularity (C) denotes the degree of deviation between the shape of the circle and the lesion area, and the "calculation" formula is described in (10).

$$C = d_{max} - d_{min} \quad (10)$$

where $\mathrm{d_{max}} - \mathrm{d_{min}}$ indicates the maximum and minimum diameter of the lesion's cross-section. The lesions E complexity is a formula for determining the perimeter of the lesion area per unit area, and it can be written as (11):

$$E = \frac{4\pi}{C} \quad (11)$$

here, N is the number of rice disease spots present on an infected leaf. The ratio of "lesion area" to the number of lesions is mentioned as (12).

$$R = \frac{S}{N} \quad (12)$$

2.3. S-particle swam optimization

The PSO technique incorporates a velocity update expression that relies on the mode using parameters from Markov to achieve a balance between local and global search. In general, early-stage particles should maintain their diversity and independence because this broadens the search space and prevents early convergence to the local optimum [19]–[22]. All groups might ultimately converge on the most correct particles and arrive at a more precise answer. These are the velocities and positions of the particles as they have evolved. Table 2 provides the pseudocode for SPSO.

Table 2. SPSO feature selection algorithm

```
Input: Fused features
Output: Optimal features
Step 1: Randomly initialize the $\boldsymbol{N}$ particles ($\boldsymbol{i = 1, 2, \ldots, n}$) swarm population.
          Here, swarm population and values are initialized by the fused features.
Step 2: Decide on the values of the hyperparameters $\boldsymbol{w}, \boldsymbol{c1}$, and $\boldsymbol{c2}$.
Step 3: For $\boldsymbol{Iter}$ inside the $\boldsymbol{range(max\ Iter)}$: # Repeat max $\boldsymbol{Iter}$ loops
"For    i in range(N): # for every particle:
      (a). Determine the ith particle swarm's new velocity [i].
           velocity is w*swarm[i].velocity plus r1*c1*(swarm[i].best
           Pos-swarm[i].position+r2*c2*(best pos swarmswarm[i].position)
      (b). If the velocity is outside the range [minx, max], clip it if the swarm is I
           velocity minx: swarm[i].velocity=minx elif swarm[i].
           velocity[k]>maxx: swarm[i].velocity[k]=maxx
      (c). Determine the i$^{th}$ particle's new position with its new velocity.
           Swarm[i].position+=Swarm[i].velocity
      (d). Update the most recent versions of this particle's and the Swarm's best versions,
           if any I
    fitness swarm[i].best
    Fitness:swarm[i].best
    Fitness is equal to swarm[i].fitness.
    swarm[i].best
         Pos=swarm[i].position
    if swarm[i].
         Fitness best pos swarm=swarm[i].fitness best fitness swarm=swarm[i].
     position
    End-for
End-for"
Step 4: Return the best swarm particle, which resulted in optimal features.
```

2.4. DBN classifier

The DBN classification model is introduced after the common DBN model, and RBM's components are re-explained in this section. An RBM is a two-layered random-generating neural network. Binary visible units are present in one layer, whereas binary hidden units are present in the other. A "nonlinear dynamic" system's energy function in the Hopfield network was used to construct an energy function that was used to determine the RBM state. As a result, the RBM model can be easily examined, and the system's objective function was changed into an extreme value. Since an RBM is considered a model, its energy function is written as (13):

$$E(v, h; \theta) = -\sum_{i,j} w_{i\,j} v_i h_j - \sum_i b_i v_i - \sum_j a_j h_j \tag{13}$$

here $\theta = (w, a, b)$ indicates the RBM parameter; $w_{i\,j}$ signifies the relationship value between visible units (v) and hidden units (h); and b_i and a_j indicates bias terms of the visible as well as hidden units. The hidden units' h conditional distributions and the visible units' v conditional distributions are written as (14) and (15).

$$p\left(h_j = {}^1/_v\right) = \frac{1}{1+exp\,(-\sum_i w_{i\,j} v_i - a_j)} \tag{14}$$

$$p\left(v_i = {}^1/_h\right) = \frac{1}{1+exp\,(-\sum_j w_{i\,j} h_j - a_i)} \tag{15}$$

The RBM's target function concentrates on finding a way to solve the h and v distribution that makes them as equal as possible. As a result, we determined their distribution's Kullback-Leibler (K-L) distance. Then cut it down. Obtaining the normalizing factor $Z(\theta)$ will help in calculating the expectation of the joint probability. is challenging, and its time complexity is $O(2^{m+n})$. As a result, Gibbs sampling was mainly on reconstructed information. Weights learning is expressed as (16) [23].

$$\Delta w_{i\,j} = E_{data}(v_i h_j) - E_{model}(v_i h_j) \tag{16}$$

DBN is a DL neural network made up of one back propagation (BP) layer and three RBM layers as shown in Figure 4 [24], [25]. The DBN has the benefit that it can extract nonlinear features from visible layer units of unlabeled training data using hidden layer units. There are two steps to training a DBN. Pretraining is the first step, where weights for the generative model are created using an unsupervised, greedy, "layer-by-layer" pretraining process. The second phase is fine-tuning, which uses the gradient descent approach to increase the discriminant model's capacity for generalization.

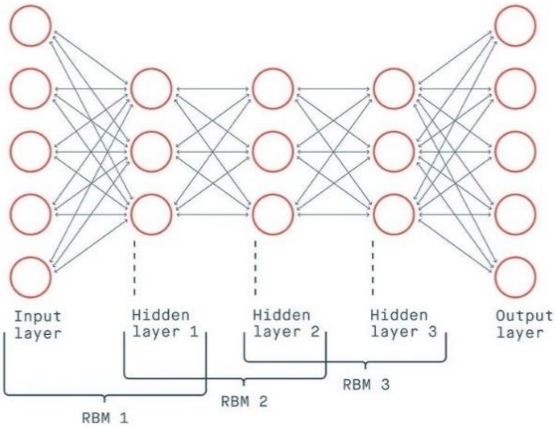


Figure 4. Architecture of DBN

The pretraining step of the contrastive divergence (CD) method calls for updating the bias and weights of the DBN using the (17) to (19):

$$w = w + \left[p\left(d^{(0)} = {}^{1}/_{s^{(0)}}\right) s^{(0)T} - p\left(d^{(1)} = {}^{1}/_{s^{(1)}}\right) s^{(1)T}\right] \tag{17}$$

$$b = b + \Delta\left[p\left(d^{(0)} = {}^{1}/_{s^{(0)}}\right) - p(d^{(1)} = {}^{1}/_{s^{(1)}})\right] \tag{18}$$

$$a = a + \Delta\left[s^{(0)} - s^{(1)}\right] \tag{19}$$

where $\mathrm{p}\left(\mathrm{d}^{(0)} = {}^{1}/_{\mathrm{s}^{(0)}}\right) = \sigma(\mathrm{w}_{\mathrm{j}}.\mathrm{s}^{(0)} + \mathrm{b}_{\mathrm{j}}$ indicates the probability that a neuron in the hidden layer would be turned on "Bayesian visible" neurons, and Δ indicates the learning rate. For the output BP layer, the minimum mean square error (MSE) criteria are employed in the fine-tuning stage, and the "cost function" is as (20).

$$E = \frac{1}{N}\sum_{i=1}^{N}(\widehat{X_\iota}(\,w^k, b^k) \text{ -}X_i)^2 \tag{20}$$

here E indicates the MSE of DBN, $\widehat{X_\iota}$ and X_i indicates that expected o/p and real output. *i* represents the sample index. w^k, b^k Represent weight and bias in the *k*th layer. The bias and weights parameters of the network are updated using the gradient descent technique in accordance with as (21).

$$(w^k, b^k) = (w^k, b^k) + \Delta.\frac{\delta E}{\delta(w^k, b^k)} \tag{21}$$

3. RESULTS AND DISCUSSION

3.1. Training and testing parameters

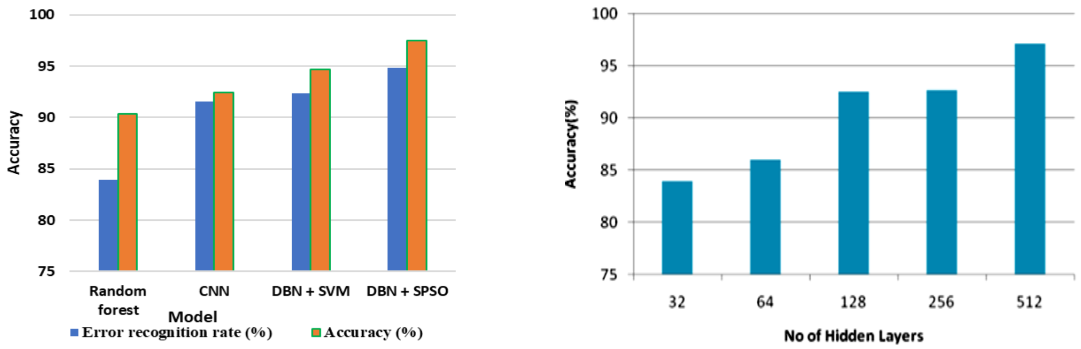
Python 3.6 along with TensorFlow 2.0 and Keras 2.2.4 open-source deep learning frameworks are used with the Ubuntu 18.04 X64 OS (operating system) to process the images and apply the algorithm. The influence on accuracy is the key consideration in the detection of rice blasts, sheath blight, and brown spots using SPSO and DBN. Following the experimental comparison, it was ultimately decided that there were 15 RBM hidden layers, 128 hidden layer nodes, 64 batches, a 0.01 learning rate, a stochastic gradient descent (SGD) optimizer, and a 0.9 momentum.

3.2. Experiment results

The DBN and SPSO models' stability and effectiveness, as well as the results of the recognition process, are validated using the 10-fold cross-validation approach. The model's accuracy and error recognition rates are 97.48% and 95.67%, respectively. Table 3 compares the performance of the proposed DBN+SPSO with existing ML methods such as the random forest model. Moreover, the proposed DBN+SPSO was also compared with the DL-based convolutional neural network (CNN) and deep belief networks (DBN)+support vector machine (SVM) models. Here, the proposed DBN+SPSO resulted in superior performance as compared to the existing ML and DL models. Figure 5(a) displays the accuracy comparison graph for the three models. It demonstrates the model’s high level of recognition capacity and ensures a low rate of false detections. Figures 5(b), 6(a), 6(b), and 7 shows the relationship between quality parameters and accuracy levels. The comparison demonstrates that the SPSO and DBN models utilized in this study give higher accuracy.

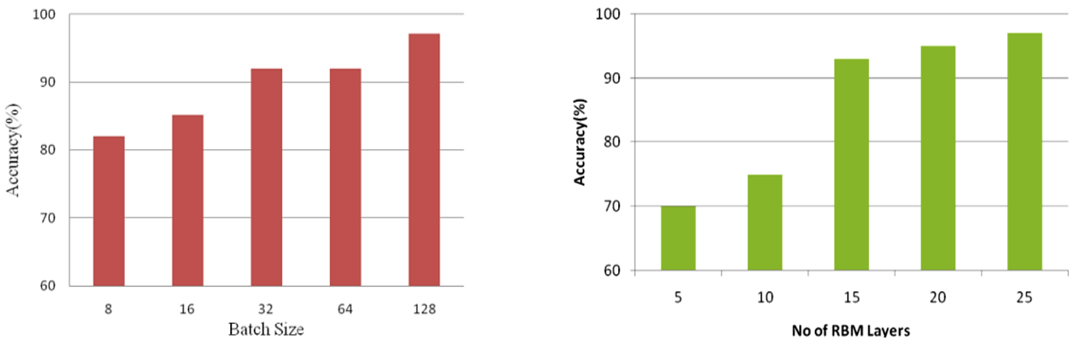
Table 3. Performance parameters of model

Model	Error recognition rate (%)	Accuracy (%)
Random forest	83.95	90.38
CNN	91.57	92.43
DBN + SVM	92.39	94.65
DBN + SPSO	94.85	97.48



(a) (b)

Figure 5. Performance of various model (a) accuracy of existing vs proposed method and (b) no of hidden layers vs accuracy



(a) (b)

Figure 6. Comparison of accuracy on batch size and RBM layers (a) shows batch size vs accuracy and (b) shows no of RBM layers vs accuracy

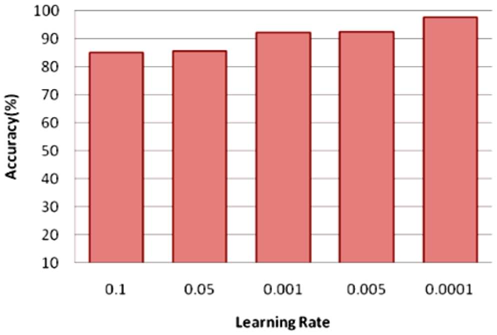


Figure 7. Learning rate vs accuracy

4. CONCLUSION

In this paper, a new way of finding rice diseases based on SPSO and DBN was suggested. Data preparation, the structure of the recognition model, and how the DBN and SPSO implementation technologies work have all been thoroughly explained. For the SPSO and DBN approaches to effectively detect rice disease from the ROI that is generated using preprocessing rice disease images, three features (containing the color, texture, and shape) were extracted. Additionally, many indices were suggested to validate the SPSO and DBN approaches that have been previously provided. These indices show that the SPSO and DBN approaches achieve improved recognition and influence and have good anti-interference and resilience. In comparison to the SVM and CNN models, the new technique has shown improved training performance, a faster accuracy rate, and improved recognition capability. Though the SPSO and DBN have improved their performance in rice disease recognition sectors, several issues remain. The first question is how to choose the best weights and bias factors. The second is how to obtain high-quality samples of rice disease images given the impact of lighting, foresight, and shooting angle. The final question is how to increase the effectiveness of model training, the level of recognition, and application capability.

REFERENCES

[1] U. B. Patayon and R. V. Crisostomo, "Peanut leaf spot disease identification using pre-trained deep convolutional neural network," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 3, pp. 3005–3012, Jun. 2022, doi: 10.11591/ijece.v12i3.pp3005-3012.

[2] S. B. Jadhav, V. R. Udup, and S. B. Patil, "Soybean leaf disease detection and severity measurement using multiclass SVM and KNN classifier," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 5, pp. 4077–4091, Oct. 2019, doi: 10.11591/ijece.v9i5.pp4077-4091.

[3] F. Cheng, H. Zhang, W. Fan, and B. Harris, "Image recognition technology based on deep learning," *Wireless Personal Communications*, vol. 102, no. 2, pp. 1917–1933, Sep. 2018, doi: 10.1007/s11277-018-5246-z.

[4] R. Cristin, B. S. Kumar, C. Priya, and K. Karthick, "Deep neural network based Rider-cuckoo search algorithm for plant disease detection," *Artificial Intelligence Review*, vol. 53, no. 7, pp. 4993–5018, Oct. 2020, doi: 10.1007/s10462-020-09813-w.

[5] H. Dong, Z. Wang, B. Shen, and D. Ding, "Variance-constrained H∞ control for a class of nonlinear stochastic discrete time-varying systems: The event-triggered design," *Automatica*, vol. 72, pp. 28–36, Oct. 2016, doi: 10.1016/j.automatica.2016.05.012.

[6] A. Fuentes, S. Yoon, and D. S. Park, "Deep learning-based techniques for plant diseases recognition in real-field scenarios," in *ACIVS 2020: Advanced Concepts for Intelligent Vision Systems*, 2020, pp. 3–14, doi: 10.1007/978-3-030-40605-9_1.

[7] S. Goudarzi, M. Kama, M. Anisi, S. Soleymani, and F. Doctor, "Self-organizing traffic flow prediction with an optimized deep belief network for the internet of vehicles," *Sensors*, vol. 18, no. 10, Oct. 2018, doi: 10.3390/s18103459.

[8] S. M. Hassan, A. K. Maji, M. Jasiński, Z. Leonowicz, and E. Jasińska, "Identification of plant-leaf diseases using CNN and transfer-learning approach," *Electronics*, vol. 10, no. 12, Jun. 2021, doi: 10.3390/electronics10121388.

[9] G. E. Hinton, "A practical guide to training restricted Boltzmann machines," in *Neural Networks: Tricks of the Trade*, 2012, pp. 599–619, doi: 10.1007/978-3-642-35289-8_32.

[10] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006, doi: 10.1126/science.1127647.

[11] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, no. 8, pp. 1771–1800, Aug. 2002, doi: 10.1162/089976602760128018.

[12] P. Kaur, H. S. Pannu, and A. K. Malhi, "Plant disease recognition using fractional-order Zernike moments and SVM classifier," *Neural Computing and Applications*, vol. 31, no. 12, pp. 8749–8768, Dec. 2019, doi: 10.1007/s00521-018-3939-6.

[13] Y. Li and X. Chao, "Semi-supervised few-shot learning approach for plant diseases recognition," *Plant Methods*, vol. 17, no. 1, Dec. 2021, doi: 10.1186/s13007-021-00770-1.

[14] Y. Li *et al.*, "The recognition of rice images by UAV based on capsule network," *Cluster Computing*, vol. 22, no. S4, pp. 9515–9524, Jul. 2019, doi: 10.1007/s10586-018-2482-7.

[15] Y. Lu, N. Zeng, X. Liu, and S. Yi, "A new hybrid algorithm for bankruptcy prediction using switching particle swarm optimization and support vector machines," *Discrete Dynamics in Nature and Society*, vol. 2015, pp. 1–7, 2015, doi: 10.1155/2015/294930.

[16] G. Madhulatha and O. Ramadevi, "Recognition of plant diseases using convolutional neural network," in *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, Oct. 2020, pp. 738–743, doi: 10.1109/I-SMAC49090.2020.9243422.

[17] Y. A. Nanehkaran, D. Zhang, J. Chen, Y. Tian, and N. Al-Nabhan, "Recognition of plant leaf diseases based on computer vision," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–18, Sep. 2020, doi: 10.1007/s12652-020-02505-x.

[18] C. R. Rahman *et al.*, "Identification and recognition of rice diseases and pests using convolutional neural networks," *Biosystems Engineering*, vol. 194, pp. 112–120, Jun. 2020, doi: 10.1016/j.biosystemseng.2020.03.020.

[19] H. Veisi and A. H. Mani, "Persian speech recognition using deep learning," *International Journal of Speech Technology*, vol. 23, no. 4, pp. 893–905, Dec. 2020, doi: 10.1007/s10772-020-09768-x.

[20] P. Wspanialy and M. Moussa, "A detection and severity estimation system for generic diseases of tomato greenhouse plants," *Computers and Electronics in Agriculture*, vol. 178, Nov. 2020, doi: 10.1016/j.compag.2020.105701.

[21] X. Wu *et al.*, "Attitude stabilization control of autonomous underwater vehicle based on decoupling algorithm and PSO-ADRC," *Frontiers in Bioengineering and Biotechnology*, vol. 10, Feb. 2022, doi: 10.3389/fbioe.2022.843020.

[22] D. Zhang, X. Zhou, J. Zhang, Y. Lan, C. Xu, and D. Liang, "Detection of rice sheath blight using an unmanned aerial system with high-resolution color and multispectral imaging," *PLOS ONE*, vol. 13, no. 5, May 2018, doi: 10.1371/journal.pone.0187470.

[23] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, Dec. 2015, doi: 10.1007/s11263-015-0816-y.

[24] P. K. Sethy, N. K. Barpanda, A. K. Rath, and S. K. Behera, "Deep feature based rice leaf disease identification using support vector machine," *Computers and Electronics in Agriculture*, vol. 175, Aug. 2020, doi: 10.1016/j.compag.2020.105527.

[25] R. Vedantham and E. S. Reddy, "A robust feature extraction with optimized DBN-SMO for facial expression recognition," *Multimedia Tools and Applications*, vol. 79, no. 29–30, pp. 21487–21512, Aug. 2020, doi: 10.1007/s11042-020-08901-x.

BIOGRAPHIES OF AUTHORS

Miryabbelli Jayaram received the B. Tech from Bapatla Engineering College. He completed M. Tech from SIT, JNTUH in 2004. He received Ph.D. from Rayalaseema University in 2019. He has 20 years of experience and working as Professor and Head in Department of CSE (Data Science), Sreyas Institute of Engineering and Technology, Hyderabad, India. He can be contacted at email: drjayaram2022@gmail.com.

Gudikandhula Kalpana holds a Ph.D. in Computer Science and Engineering in the year 2021 from JNTUH. Received her M. Tech in Computer Science and Engineering from JNTUH in the year 2011. Presently working as Professor and HOD of CSE-AI and ML Dept., in Malla Reddy Engineering College for Women (Autonomous), Hyderabad. Her research interests include ML, data analysis, agricultural computing, information security, cryptography and cloud systems. She can be contacted at email: dr.gkalpanamrecw@gmail.com.

Subba Reddy Borra received the B. Tech from Bapatla Engineering College. M.Tech obtained from JNTUK in specialization neural networks. Received Ph.D. from JNTUH in 2021. He has 22 Years of experience and working as Professor and Head in Department of Information Technology from Malla Reddy Engineering College for Women (UGC-Autonomous) Hyderabad, India. His area of interest is Image Processing and ML. He can be contacted at email: bvsr79@gmail.com.

Battu Durga Bhavani working as Associate Professor in the Department of Information Technology from Malla Reddy Engineering College for Women (UGC-Autonomous), Secunderabad, Telangana, India. She has 10 Years of experience. He can be contacted at email: durgabhavanimrecw@gmail.com.