



HML-SL: A hybrid machine learning technique for sugarcane leaf disease detection

^{1*}Mr.V.Selvakumar, ²Dr.Seetharaman

^{1*} Assistant Professor Department of Computer and Information Science
Annamalai University
vselvakumar1921@gmail.com

² Professor Department of Computer and Information Science
Annamalai University

Abstract: Plant diseases have traditionally affected plant growth and crop production in many regions of the world, reducing food supplies. Sugarcane farming is well-organized. Due to optimal growth conditions, it is farmers' first option. It is tied to the sugar industry and plays a major role in Brazil, India, China, economies and so on. The most valuable commercial crop is sugarcane. Diseases impact agricultural productivity and quality. Some of these are found when farmers visually check leaves. Most infections remain undetected, losing farmers millions. Identifying the infection type significantly reduces its harm. To overcome this difficulty, we propose a hybrid machine learning technique for sugarcane leaf disease detection (HML-SL) in this research work. Then, we develop the pigeon inspired recurrent neural network (PI-RNN) to classify the disease type on sugarcane leaf in terms of healthy, red rot, mosaic, ring spot and grassy shoot diseases. Finally, the performance of proposed HML-SL technique shows the effectiveness over the existing techniques. The simulation results demonstrated that the proposed PI-RNN classifier outperformed existing classifiers in accuracy, area under curve, precision, F-measure and Recall. Our research provides practical application for the sugarcane leaf disease detection as a precision agriculture system that predicts sickness by analyzing images of infected sugarcane leaf disease with Computer Vision, Image Processing, and Machine Learning algorithms.

Keywords: machine learning, sugarcane leaf, crop, ring spot, clustering.

DOI Number: 10.48047/nq.2022.20.19.NQ99145 NeuroQuantology2022;20(19): 1605-1629

1. Introduction

Sugar plays an important role in India's agriculture and economy [1]. Although there are different types of diseases, the 3 main diseases are rust spots, yellow spots and ring spots [2]. These diseases can be identified by sugarcane leaf spots. The disease can be detected with the naked eye, but using this method will give better evaluation results [3, 4]. In the reproductive process of reeds the important step is assessing the severity of plant diseases.



In a sugar mill this disease not only reduces yields but also increases yields [5]. The severity of the disease is depended by the sugarcane production. About 15% of cane spots on leaves are diseased and there is a significant loss in sugarcane production [6]. In order to control this disease and reduce the infection in the entire area of the cane, this disease should be detected and treated in advance [7]. The three main types of spot disease affecting tropical crop are rust spot, yellow spot and ring spot [8]. Symptoms of diseases caused by this fungus are obvious spots on the leaves. These sites can be automatically identified based on the characteristics of the spots [9]. However, there are limitations to accepting this guideline, which is often detrimental to the effort. Spontaneous diagnosis and observational assessment using the naked eye cannot accurately determine the subject and disease [10]. As the use of pesticides in the treatment of plant diseases increases the level of toxic pollutants in agricultural production, groundwater pollution occurs, which, if not treated properly, requires high production costs, reducing the risk of accurate analysis of incorrect analysis [11].

Phytoplasmosis is associated with two major diseases of the sugar cane, (i) Bull blind disease (GST), and (ii) white leaf disease (WLD) [12]. These diseases are found only in Asia and not in other parts of the world. Thailand, Taiwan and Japan reports White leaf diabetes. Bangladesh, India, Malaysia, Nepal, Pakistan and Sri Lanka has reported Grass shoot disease [13]. The identity of the phytoplasma associated with sugarcane in Sri Lanka has not yet been confirmed by the white leaf and extensive plowing [14]. Plant disease digital imaging is used to determine the type of spot disease based on a taxonomic model [15]. Paddy disease was diagnosed based on a systematic analysis using a fracture description of HSV-colored varnish and "S" components [16]. The 95.25% accuracy classification method is used to detect fungal diseases in reed leaves based on the isolated area of the image [17]. Some spot diseases such as yellow spot, brown spot, ring spot and rust spot differ with threshold technique [18]. At the same time, the severity of the disease points can be calculated by comparing the entire leaf of the affected area [19].

Leaf spots are a symptom of diabetic and fungal diseases. This symptom can be visualized [20]. The severity of the disease can be assessed by measuring the affected area and leaf area [21]. However, the risk of infection is caused by a number of factors: transmission from pieces promotes the spread of pathogens [22]; Favors the development of monocultural infections over large areas; the diversity of this culture (average 4 to 10 years) is not fully utilized at different potential levels [23]. Reeds can be infected by a variety of pathogens without symptoms, so the risk of infection is very high [24]. Yellow Leaf (YL) is a major viral disease caused by Sugarcane yellow leaf virus (SCYLV), which has recently become a threat to sugarcane cultivation worldwide [25]. From October through March, mature cane yellows most. As the season advances, the midrib and leaf blade yellow,



resulting in an overall yellowing of the leaves. Here, An Hybrid machine learning for sugarcane leaf disease detection (HML-SL) enhances disease prediction accuracy.

1.1 Contributions:

The main objective of the HML-SL approach that has been presented is to provide timely warning messages to farmers, which helps to prevent unanticipated financial losses.

- First, the proposed HML-SL disease detection technique is to introduce the JAYA clustering algorithm for efficient segmentation process which segments the disease portion from the input leaf image.
- Second, to illustrate the efficient feature selection process based on improved coyote optimization (ICO) algorithm.
- We develop the pigeon inspired recurrent neural network (PI-RNN) to classify the disease type on sugarcane leaf in terms of healthy, red rot, mosaic, ring spot and grassy shoot diseases.

1.2 Organization of paper

The article's remaining portions are: Sect.2 discusses recent literature works Sect.3 illustrates problem methodology and system model of proposed disease detection technique .Sect. 4 provides the detailed working function of proposed disease detection technique. The comparative analysis of proposed and existing disease detection and classifiers are discussed in Sect. 5. Sect.6 summarizes the research work.

2. Related works

Shruthi et al. [26] (2019) compared machine learning classification algorithms for plant disease detection. Their study found that CNN obtained high accuracy and identified more crop diseases. Annabel et al. [27] (2019) examined various plant diseases and machine learning classification algorithms for spotting them in plant leaves. Militante et al. [28] (2019) combined CNN architectures of deep learning models to identify sugarcane diseases with high accuracy. The models were trained using 14,725 images of normal and infected sugarcane leaves, obtained 95.4% accuracy.

Militante et al. [29] (2019) an approach to determine several plant diseases. The system detects apple, maize, grapes, potato, sugarcane, and tomato. The trained model obtained 96.5% accuracy, while the system showed 100% accuracy in identifying plant variety and disease type. Amarasingam et al. [30] examined the use of UAVs in the sugarcane sector to enhance pest and disease control, production estimation, phenotypic measurement, soil moisture monitoring, and nutritional status evaluation.

Kumpala et al. [31] (2022) simulated image identification using deep learning system based on the CNN algorithm YOLO. The first and second groups' average processing

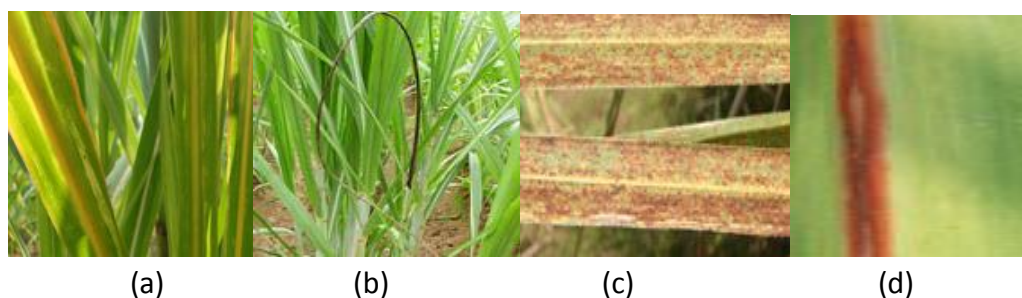


accuracy was 95.90% and 91.30%, with highest accuracy of 98.45% and 97.26%, respectively. The average estimated time duration was 1.46 sec and 1.53 sec, respectively. Balan et al. [32] (2022) investigated viral transmission from sugarcane parents to children. Eight- and three-month-old progenies from 18 hybrids and their parents were examined using PCR. All tested parents were virus-positive. 50 to 100% of progenies were SCBV-positive in PCR tests.

Using molecular approaches, Haregu et al. [33] (2022) identified and characterised SCBV isolates in Ethiopia. 76 out of 292 PCR-tested leaf samples (26%) were SCBV-positive. Three Ethiopian isolates (SCBV-EtS3, SCBV-EtS6 and SCBV-EtC10) revealed 99.5–100% nucleotide similarity among themselves and with Chinese isolated (accession numbers MH037614 and MH037915). SCBV-EtC2 shared 78% similarity with the other three Ethiopian isolated and 99.8% with Chinese isolated (KM214357 and KM214307). Zhao et al. [34] (2022) analyzed 107 sugarcane varieties developed between 1952 and 2020 and discussed replacement and exchange.

3. Problem methodology

Due to climate change and unexpected weather conditions foliage diseases are more prevalent in the sugarcane crop. Some of the leaf diseases are list as follows: Yellow leaf disease (Fig. 1a): The phloem of the plant is affected by a virus which causes a disease called yellow leaf disease transmitted from aphids, *Melanaphis sacchari* and *Rhopalosiphum maidis*. Smut (Fig. 1b): It is transmitted by fungal mycelium which causes a mass of black powdery spores that is grown in top of the plant. Rust (Fig. 1c): *Uredinia* transmits a rust in the sugarcane leaf which causes a leaf to changes its original color ,sometimes it leads to premature death of the leaf. Red rot disease (Fig. 1d): The fungus *Glomerella tucumanensis* causes a red rot disease. It forms a short line in between the leaves which is reddish in color. Ratoon stunting (Fig. 1e): A small bacterium called *Leifsonia xyli* subsp. *Xyli* causes a disease called ratoon stunting which retards the growth in plant. Wilt (Fig. 1f): *Hyaline* produces wilt, which causes cane shrinkage and smell. Sett rot (Fig. 1g): *Ceratocytis paradoxa* transmits a disease called sett rot which causes the leaf to turn red and shoots die after attaining certain stage. Grassy shoot disease (Fig. 1h): It is caused by mycoplasma which causes retarded growth and later the leaf turns white in color



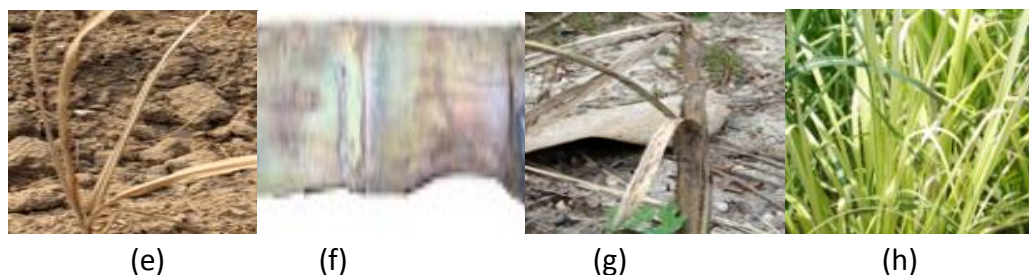


Fig. 1 Sugarcane leaf disease images (captured using 12 Mega pixel Digital Camera) (a) yellow leaf (b) smut (c) rust (d) red rot (e) ratoon stunting (f) wilt (g) sett rot and (h) grassy shoot

3.1 Research Gap

Srivastava et al. [35] have proposed a novel deep learning approach for evaluating a sugarcane plant's leaves, stem, colour, etc. to identify disease. The author utilized Inception v3, VGG-16, and VGG-19. These are different types of built-in sub-models. Orange software calculates the range of accuracy, precision, character, curve and sensitivity and selects the most accurate screen. 90.2% AUC is available with VGG-16 as a feature exhaust and SVM classifier [36][37]. Fungi, bacteria, pathogens, and viruses affect crop production and farmers' income [38]. Recently, image processing based techniques have been proposed to predict the diseases on various plants like groundnut, tomato, potato and etc [39][40]. For further improvement in sugarcane leaf disease detection, we propose a hybrid machine learning technique i.e. (HML-SL) which comprises following techniques are JAYA clustering algorithm, improved coyote optimization (ICO) algorithm and pigeon inspired recurrent neural network (PI-RNN) classifier.

The main contributions of proposed HML-SL technique are summarized as follows:

- JAYA clustering algorithm for efficient segmentation process which segments the disease portion from the input leaf image.
- An efficient feature selection process based on improved coyote optimization (ICO) algorithm.
- Pigeon inspired recurrent neural network (PI-RNN) to classify the disease type on sugarcane leaf in terms of healthy, red rot, mosaic, ring spot and grassy shoot diseases.
- Finally, the performance of proposed HML-SL technique shows the effectiveness over the existing techniques in terms of accuracy, area under curve, precision, F-measure and Recall.

3.2 System architecture of proposed HML-SL technique

Fig. 2 shows the image-based disease classification model for a sugarcane leaf image. First, the desired preprocessing approach distorts or enhances some features. Using JAYA algorithm, the segmentation component separates diseased from normal leaf. Then, feature

extraction is utilized to evaluate the shape, corner, and edge. ICO system selects the most comparable features. The leaf is then classified using PI-RNN as red root, mosaic, ring spot, and grassy shoot disease

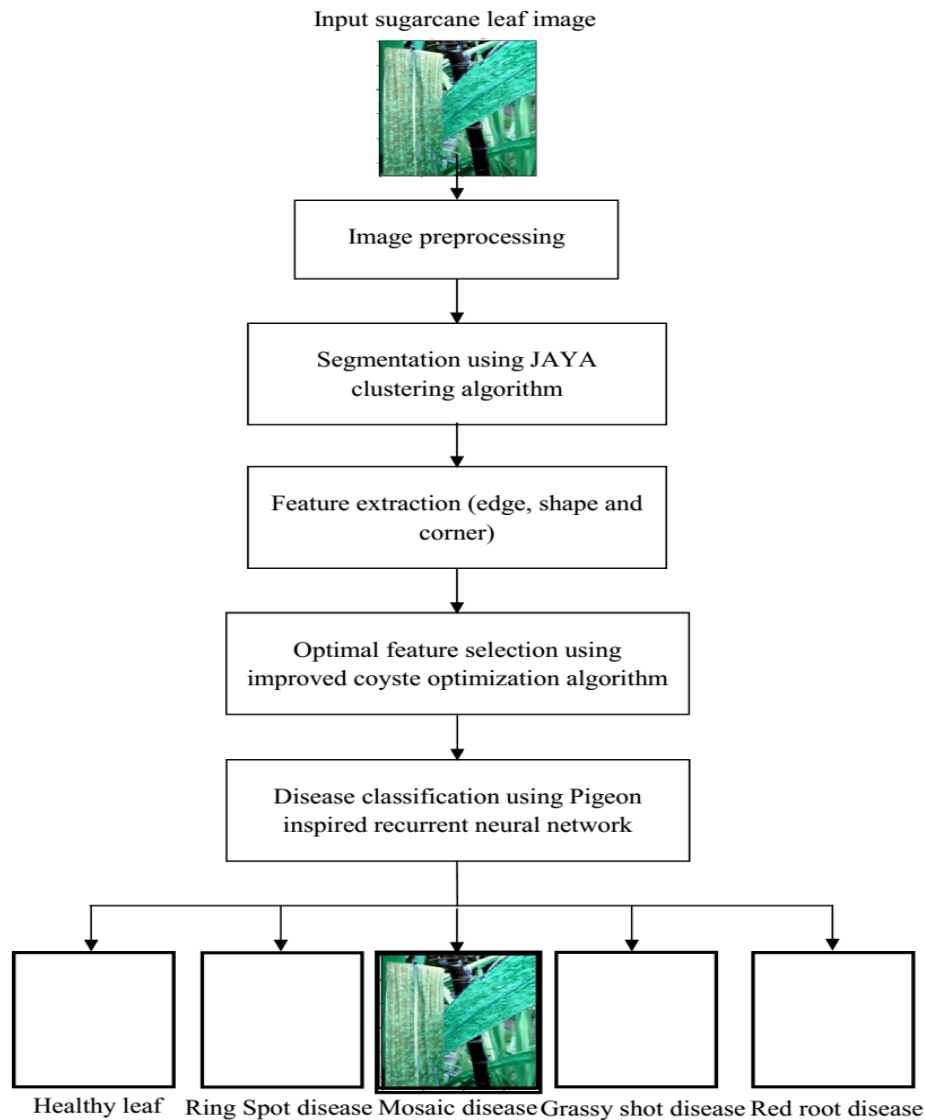


Fig.2 System architecture of proposed technique

4. Proposed system

4.1.1 Materials Required

4.1.1.1 Orange software

For creating feature extractor and classifier, we have used Orange software for this purpose. How were these photos obtained? How did you get the images of the leaves and diseases? The type of camera, definition, lighting with which it was obtained the pictures and other technical details are not mentioned.

Orange is an open-source data mining toolkit used for visualization of data, machine learning and data analysis. We visualize the sugarcane leaf images and diseases using

Orange toolkit [41]. The components or tools in Orange with the help of which data are visualized and the classifiers are applied are known as widgets. Programming in Orange is done visually in the workflow window. Pre-defined widgets are used by linking them together in the workflow. This workflow is considered as a canvas interface. Firstly, we have loaded the dataset in the Import Image widget of Orange and then selected different feature extractors from Image Embedding widget. Then, classifiers are connected with the test and score widget in order to compute the accuracy.

4.1.1.2 Technical details

Table 1: Technical details	
Toolkit	Orange Software
Camera	12 Mega pixel Digital Camera
Light system	Light System with 18 W cool light bulbs and light filters, White paper sheet
System	PC
Programming	MATLAB 7.4 Version

4. Hybrid machine learning technique for sugarcane leaf disease detection (HML-SL)

4.1 Segmentation using Jaya clustering algorithm

Segmentation is the process of separating the data from a group. In leaf disease, segmentation technique will separates the diseased part from the healthy leaf with the best accuracy ratio. Segmentation is an important step in image processing techniques for identifying the affected area. Clustering is said to be grouping of leaves that is diagnosed with disease. By using this clustering methodology we could able to predict the disease at initial stage. By identifying disease earlier we can segment both healthy and disease affected leaves. Jaya Algorithm (JA) is effectively used a simple and powerful global optimization algorithm for controllable and uncontrollable orientation problems. For a particular problem the solution will be acquired and omit the awful option based on the view. This algorithm needs only the countable amount of control parameters, such as least generation, size of population and factors of design. Jaya Algorithm is the best and most adaptive optimization approach and always strives to succeed by making the best decision. It doesn't contain any



particular parameters for control. The standard control parameters are required for changing and the solution can modernize in one stage only. In engineering application it is used easily. The good and bad solutions are used to update the solution of the candidates in JA. The importance is the Jaya algorithm does not require specific algorithm settings that require individual installation settings before any actual system testing is performed, as this may result in unavoidable errors by other algorithms without special requirements and testing systems with other methods is not only simple, but in many cases time consuming and the whole process is not required in JA. The leaves are separated by the segmentation process. Using the clustering the leaves are classified according to the color change of the leaves. Then, the classified leaves were indicated as good or bad solution by the Jaya algorithm. Assume the number of design variables i.e. $u=1, 2... n$ and start the JA with the initialization of the size population m (i.e. $q=1, 2... n$). Based on the objective function value, the good and bad solution of candidates is recognized from the population. In the following equation (2), substitute the candidate solution.

$$v'_{u,q,j} = v_{u,q,j} + s_{1,u,j}(v_{u,good,j} - |v_{u,q,j}|) - s_{2,u,j}(v_{u,bad,j} - |v_{u,q,j}|) \quad (1)$$

Here, the value of $v_{u,q,j}$ indicate the u^{th} variable for the q^{th} candidate, the value of the u^{th} variable $v_{u,good,j}$ indicates the good candidate solution and $v_{u,bad,j}$ refers the value of the u^{th} variable of bad candidate solution. If the value of the best object gives the value of the best candidate, get it; keep removing the old candidate. At the end of the iteration, the received candidate opens the decisions for the next iteration. Once the termination criteria are satisfied the procedure will be stopped. In iteration the good and bad solutions are simplified.

$$d_{n+1} = 4d_n(1 - d_n) \quad (3)$$

$$v'_{u,q,j} = v_{u,q,j} + d_{n,u,j}(v_{u,good,j} - |v_{u,q,j}|) - d_{n,u,j}(v_{u,bad,j} - |v_{u,q,j}|) \quad (4)$$

In the iteration the d_n be a chaotic random number and in the equation (16) substitute the solution. Let the upper and lower bound of the variable u is denoted as ub_u and lb_u respectively. The equation (5), (6) and (7) shows the value of the candidate solution variable $v_{u,q,j}$.

$$v_{u,q,j}^p = RAND(b, a) \quad (5)$$

$$b = \frac{(lb_u + ub_u)}{2} \quad (6)$$

$$a = lb_u + ub_u - v_{u,q,i} \quad (7)$$

In JA, for the good solution of candidate the equation of movement is considered. The motion equation is calculated as one of the first three decisions based on the importance of the exercise and not the best candidate.



$$v'_{u,q,j} = v_{u,q,j} + s_{1,u,j}(v_{u,good(sa),j} - |v_{u,q,j}| - s_{2,u,j}(v_{u,bad,j} - |v_{u,q,j}|)) \quad (8)$$

Let d_n be a chaotic random number generated by Equation (2) and RAND be a random number between [0, 1]. The solution of a new candidate is created using the following equation (9):

$$v'_{u,q,j} = v_{u,good,j} + RAND(2d_n - 1) \quad (9)$$

In this, substitutes a random number between a chaotic random number [0, 1].

$$v'_{u,q,j} = u_{v,good,j} + RAND(2RAND - 1) \quad (10)$$

The algorithm 1 shows the working function of the Jaya based clustering algorithm.

Algorithm 1 Segmentation using Jaya clustering algorithm

Input : $v_{u,q,j}$, $v_{u,good,j}$, ub_u and lb_u

Output : $v'_{u,q,j}$

- 1 Assume the value for the size population and the designs
- 2 Using the objective function value, the good and bad solutions are calculated
- 3 Calculate the candidate solution of the lower and upper bound
- 4 Compute the new candidate solution
- 5 Update the random number in the equation
 $v'_{u,q,j} = u_{v,good,j} + RAND(2RAND - 1)$
- 6 End

4.2 Feature extraction using improved coyote optimization (ICO)

Feature extraction is a size reduction process, so the initial data source is controlled for processing groups. One of the characteristics of this big data set is that it has many variables, the processing of which requires large computing resources. Feature extraction is a method of combining variables with characteristics that select and/or effectively reduce the amount of data needed to process real data sets when they are accurately and precisely described. Feature selection is one of the most important issues in the field of machine learning and communication. In general, feature selection aims to reduce data size and create more robust taxonomic models. The method of selecting features affects the results of classification and clustering. A feature is that subgroup classification and clustering should be strongly supported. Filtering and packing methods are two methods of selecting characters called large data packets. In filter mode, properties were selected based on sample separation that differed from the learning process. Only the relationship between the separation properties is taken into account, so the selected properties may not be optimal. Packaging methods often look for key features that are used based on the learning



method and often give better results than the filtration method Image processing feature is used in many applications. Color, texture, shape and edges are diagnostic features of plant diseases. How color is distributed in a picture, it makes the picture solid and solid. It can also be used to identify areas of diseased plants. Here we are mainly concentrating on color of the leaf captured using 12 Mega pixel Digital Camera & Light System with 18 W cool light bulbs and light filters.

Sugarcane diseased leaves are taken for this study. Images are taken in controlled environment and are stored in the JPEG format. Infected leaf is placed flat on a white background; Light sources are placed at 45 degree on each side of the leaf so as to eliminate any reflection and to get even light everywhere, thus a better view and brightness. The leaf is zoomed on so as to ensure that the picture taken contains only the leaf and white background.

The color change in the leaf will be showing the diseased part. Optimal feature selection is used to detect the change in color of the leaf, if the disease is present. Here the disease is caused by virus which tends to change the color in the sugarcane; also it will stop the growth of the plant. Considering this, an improved coyote optimization (ICO) technique is used for feature selection. The main advantage of other ICO is that the developed method allows changing the ecological and social structure of the coyote and compares it with other methods provided by other literary technologies. ICO also focuses on the exchange of practices and experiences between coyotes and their social construction. ICO optimization process can be explained by the following steps: By considering the upper and lower bounders, the population of coyotes is started randomly as follows:

$$C_h^{q,T}(y, x) = Rand((y, x) * (va_i - ka_i)) + ka_i(y, x) \quad (10)$$

The upper and lower bound are represented by va_i and ka_j respectively. The coyotes in the pack are the created solution which can be defined as:

$$C_{h,i}^{q,T}(y, x) = \begin{bmatrix} C_{1,1} & C_{1,2} & \dots & C_{1,x} \\ C_{2,1} & C_{2,2} & \dots & C_{2,x} \\ \vdots & \vdots & \ddots & \vdots \\ C_{y,1} & C_{y,2} & \dots & C_{y,x} \end{bmatrix} \quad (11)$$

The main function is represented by adaption of coyotes which is defined as:

$$B_h^{q,T}(y,1) = g(C_{h,i}^{q,T}(y, x)) \quad (12)$$

According to their objective function, arrange the coyotes in order to determine the alpha.



$$R_{h,i}^{q,T} = \begin{bmatrix} R_{1,1} & R_{1,2} & \dots & R_{1,x} \\ R_{2,1} & R_{2,2} & \dots & R_{2,x} \\ \vdots & \vdots & \ddots & \vdots \\ R_{y,1} & R_{y,2} & \dots & R_{y,x} \end{bmatrix} \quad (13)$$

The representation of alpha of the pack is the best-adapted coyote to social condition

$$alpha(1, x) = g(R_{h,i}^{q,T}) \quad (14)$$

Coyotes information is determined by the tendency of the pack:

$$i^{q,T} = \left\{ \begin{array}{l} L_{Mh+1}^{q,T}, i \quad Mh \text{ is odd} \\ \frac{L_{Mh+1}^{q,T}}{2}, \left\{ \begin{array}{l} L_{Mh,i}^{q,T} \\ + L_{\frac{Mh+1}{2},i}^{q,T} \end{array} \right\} \quad \text{Otherwise} \end{array} \right\} \quad (15)$$

where $L^{q,T}$: Ranked social condition, T: an instant of time. The coyotes can be updated according to alpha and the tendency:

$$NEW_C_h^{q,T} = C_h^{q,T} + z_1 * \delta_1 + z_2 * \delta_2 \quad (16)$$

$$\delta_1 = alpha^{q,T} - C_{z_1}^{q,T} \quad (17)$$

$$\delta_2 = t^{q,T} - C_{z_2}^{q,T} \quad (18)$$

The pack in the coyotes is generated randomly in the pack. Estimate the new objective function after the update.

$$NEW_Oa_h^{q,T} = G(NEW_C_h^{q,T}) \quad (19)$$

The following shows the best objective function

$$C_h^{q,T+1} = \left\{ \begin{array}{ll} NEW_C_h^{q,T} & NEW_Oa_h^{q,T} \\ C_h^{q,T} & otherwise \end{array} \right\} \quad (20)$$

Equation (21) shows the coyote's new birth

$$Pup_i^{q,T+1} = \left\{ \begin{array}{ll} C_{z_1,i}^{q,T} & R_i < A_r \text{ or } i = i_1 \\ C_{z_2,i}^{q,T} & R_i \geq b_r + A_b \text{ or } i = i_2 \\ y_i & otherwise \end{array} \right\} \quad (21)$$

The working function of proposed feature selection using ICO algorithm is given in Algorithm 2.

Algorithm 2 Optimal feature selection using ICO algorithm

Input: $C_h^{q,T}(y, x), C_{h,i}^{q,T}(y, x)$

Output : $Pup_i^{q,T+1}$



- 1 Compute the population of coyotes
- 2 Create the coyote solution
- 3 Adaption of coyotes is represented by

$$B_h^{q,T}(y,1) = g(C_{h,i}^{q,T}(y,x))$$

- 4 Update the coyotes

$$NEW_C_h^{q,T} = C_{h,i}^{q,T} + z_1 * \delta_1 + z_2 * \delta_2$$

- 5 If A=1,the pup dies only in the coyote pack
- 6 If the pup survive, the oldest coyote in the pack will die, else pup dies
- 7 End if
- 8 End else
- 9 Update the age of the coyote

4.3 Leaf disease classification using pigeon inspired recurrent neural network (PI-RNN)

Sugar is a perennial plant in the Phocaea family that grows on its stalk (reed) and is mainly used to produce sucrose. The cane has a thick and residual stalk that is clearly divided into nodes and internodes. The stem helps the growth of the plant which is arranged in either sides of the stem. The leaves are tubular, net-like and thick at the edges and stem. The disease detection is needed to enhance the growth of sugarcane crop, which gives continues monitoring the health condition. In this paper, we utilized the pigeon inspired recurrent neural network (PI-RNN) for leaf disease classification. Recurrent neural network (RNN) switching modes and has internal memory to automate intermediate scenes. The past information is stored by feedback network. Fig. 2 shows the simple structure of RNN. The input sequence converts the data into RNN vectors and allows the current monitor to generate output in the context of the network's internal (hidden) state.

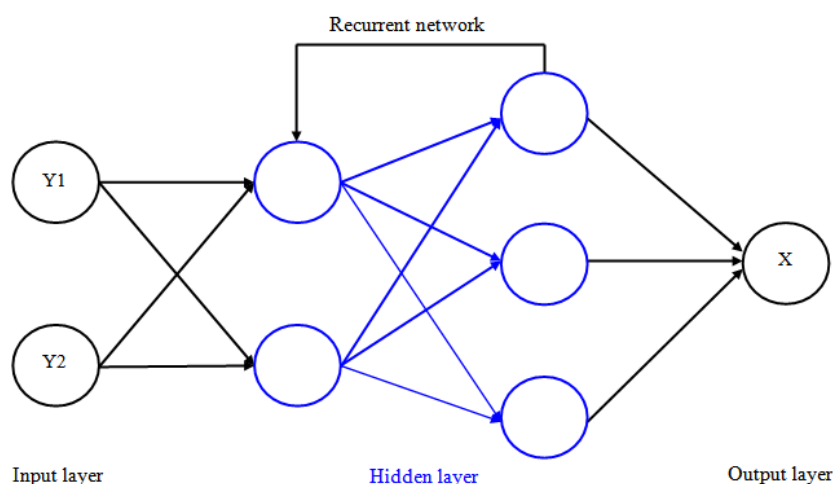


Fig.3 Structure of RNN



RNN which generates more data is prevented by disappearance of gradient in database. LSTM In addition to the cell state, LSTM consists of three gates (i.e., a forget gate, an input gate and an output gate). The information entered in the current time range is related to the hidden status of the previous timeline and it sends it to the input gate via the activation function:

$$E_s = \mathcal{G}(Y_s Z^E + T_{s-1} u^E + a_E) \quad (22)$$

where E_s is not recall gate. Y_s is the input at time step s and T_{s-1} represent hidden state at the preceding time step $t-1$. Z^E is the weight of the input layer and u^E is recurrent heaviness of the hidden state. The a_E is the bias of the input layer. The input gate decides which in sequence will be store in the cell state. There are two tasks in the input gate: The efficient value is described by first task, in the first task, while in the second task, tanh layered network is controlled by creating a value vector for all new programs. The equations for the two tasks are as follow

$$j_s = \mathcal{G}(Y_s Z^j + T_{s-1} u^j + a_j) \quad (23)$$

$$\tilde{H}_s = \tanh(Y_s Z^h + T_{s-1} u^h + a_h) \quad (24)$$

$$H_s = H_{s-1} * E_s + j_s * \tilde{H}_s \quad (25)$$

The output gate determines the hidden levels at which the sigmoid activation function is predicted. Sends to the newly modified cell level function and multiplies as follows to create a create output, as follows

$$K_s = \mathcal{G}(Y_s Z^K + T_{s-1} u^K + a_K) \quad (26)$$

$$T_s = K_s * \tanh(H_s) \quad (27)$$

{W,U} represents the weight of the matrixes. The input layer, hidden layer and output layer which is connected with weights is represented by $Z = \{Z^j, Z^h, Z^K\}$ respectively, while $u = \{u^j, u^h, u^K\}$ indicates the weight of the previous latent state (i.e., repeat weight) in the input layer, the lattice layer, and the output layer. The anthology of bias vectors can be agreed as a matrix $A = \{a^j, a^h, a^K\}$, with the indices analogous to the matching layers as describe formerly. The contributions of the two candidates are combined, using the sigmoid function to determine the values of one from zero and one.

$$W_s = \mathcal{G}(Y_s Z^W + c_{s-1} u^w + a_w) \quad (28)$$

where W_s is represent the update gate, the Y_s the input vector at time step s while c_{s-1} is the preceding output from preceding units .The Z^W is the weight of the input layer, and u^w is the repeated weight. The a_w is the bias of the input layer. The new input unit and concealed state are multiplied by their matching weights $\{Z^R, U^R\}$ and the result is summarized during the sigmoid process. The output of the reset gate is as follows,



$$r_s = \mathcal{G}(Y_s Z^R + T_{s-1} u^R + a_R) \quad (29)$$

The new reminder phone uses the reset reset gateway to store the previous state order. Once the input gate is multiplied by its weight, the element increment comes out of the remaining gate (Θ). This will allow the network to queue up only the relevant past. The current memory interaction is considered as follows

$$\tilde{H}_s = \tanh(Y_s Z + u(r_s \Theta c_{s-1})) \quad (30)$$

$$c_s = W_c \Theta c_{s-1} + (1 - W_c) \Theta \mathcal{G}(\tilde{H}_s) + a_c \quad (31)$$

The update portal determines the current content of the memory and what information should be collected from the previous steps. The new hidden state of the output of the current memory must be stored in the final memory and the sigmoid (\mathcal{G}) activation function must be passed. In RNN, the random collection of a number of hidden neurons capacity cause either over-fitting or under-fitting problems. In this process, two operators are used to help the pigeons find the best locations and solve the problem. For the optimization difficulty, each pigeon has a situation, which is definite as follows:

$$Y_j = [y_{j1}, y_{j2}, \dots, y_{jd}] \quad (32)$$

The dimension is described by d, and the pigeon population is described by 1, 2, ... N. The velocity of each pigeon is described as follows:

$$v_j = [V_{j1}, V_{j2}, \dots, V_{jn}] \quad (33)$$

First, estimate the location and speed of the dust in the search area. Then, as the number of repetitions increases, the v_j can be updated with the following repetitions

$$v_j(s) = v_j(s-1) \cdot e^{-rs} + rand \cdot (Y_{Gbest} - Y_j(s-1)) \quad (34)$$

where S is the number of current iterations, $S = 1, 2, \dots, S_{1MAX}$, S_{1MAX} is the and the maximum number of iterations. The weight coefficient used to control the recurrence rate is graph, the compass operator r is a numeric value distributed randomly from 0 to 1, Y_{Gbest} is the global optimal solution. Then the next Y_j is calculated as follows

$$y_j(s) = y_j(s-1) + v_j(s) \quad (35)$$

Algorithm 3 PI-RNN Process

Input : E_s, H_s

Output : $K_s, Y_j(s)$

1. compute the vector of recurrent neural network hidden layer
 2. Create the recurrent neural network algorithm
 3. If= S_{MAX} value are frequency system.
 4. Calculate the Z and U value with sigmoid function.
-



5. Hadamard (Θ) should be practical in the rest gate (r_s) and proceeding out c_{s-1}
6. End for
7. Get non-dominated pigeon individuals for $j = 1, \dots, n_q$
8. Initialize the best position of each pigeon $Y_{g_{best}} = Y(1)$
9. Adaption of PI-RNN is represented by $Y_j = [y_{j1}, y_{j2}, \dots, y_{jd}]$
10. Update velocity (v) and position $Y(s) : v_j(s) = v_j(s-1) \cdot e^{-rs} + rand \cdot (Y_{G_{best}} - Y_j(s-1))$
11. Evaluate each of the pigeons in population and update the best position of each pigeon
12. Update the secondary repository
13. End

The first evolutionary process used by the map and compass operator will be stopped until the frequency increases. The remaining pigeons will use the landmark operator as a navigation guide which is defined as Y_{CENTER} . Hence, the position at the iteration N_{th} can be updated by

$$Y_j(s) = Y_j(s-1) + rand \cdot (Y_{CENTER}(s-1) - Y_j(s-1)) \quad (36)$$

$$Y_{CENTER}(s) = \frac{\sum_{j=1}^n Y_j(s) \cdot fitness(Y_j(s))}{n_q \cdot \sum_{j=1}^n fitness(Y_j(s))} \quad (37)$$

$$n_q(s) = ceil\left(\frac{n_q(s-1)}{2}\right) \quad (38)$$

The number of the iteration S is represented by $S = 1, 2, \dots, S_{2MAX}$, S_{2MAX} is the greatest number of which the signpost operator works. The optimization of fitness is obtained by:

$$fitness(Y_j(s)) = E_{MAX}(Y_j(s)), \text{ For maximum problems} \quad (39)$$

$$fitness(Y_j(s)) = \frac{1}{E_{MIN}(Y_j(s)) + \varepsilon}, \text{ For maximum problems} \quad (40)$$

The pigeons are positioned close to the focus before the repetitions arrive S_{2MAX} . The algorithm 3 describes the working function of the PI-RNN classifier.



Hyperparameter Tuning

Choosing optimal hyperparameters for a learning algorithm is hyperparameter tuning. The value of a hyperparameter is defined before learning starts. Tuning hyperparameters is essential in machine learning (ML). It controls a machine learning model's behavior. This study evaluates hyperparameter-tuned ML models. Hyperparameter tuning selects optimal learning algorithm hyperparameters. A hyperparameter is a pre-learning model argument. Hyperparameter tuning is essential to ML algorithms. Linear regression without hyperparameters is vanilla. Regularization is a hyperparameter for ridge and lasso. Decision tree hyperparameters are max depth and min leaf observations.

Optimal hyperparameters regulate over- and under-fitting of the model. Optimal hyperparameters varied across datasets. Steps are taken to obtain the optimal hyperparameters:

1. Each hyperparameter setting is evaluated.
2. Best-fitting hyperparameters are selected.

Hyperparameter tuning could improve ML model accuracy.

5. Results and discussion

5.1 Dataset Formation

Image acquisition and data collection play a vital role in proper functioning of any deep learning framework. In our case, the dataset was formed by collecting images of diseased and non-diseased sugarcane from Mawana Sugar Mill Pvt. Ltd, Haryana. It was then divided into two sets—training set and test set. A total of 1000 images were present in training set and 800 images in test set (Table 2)

Dataset	Category	No. of images
Training set	Diseased	500
	Non-diseased	500
Total		1000
Test set	Diseased	400
	Non-diseased	400
Total		800
Entire dataset		1800



In this section, we evaluate the performance of proposed HML-SL technique and existing techniques. Data augmentation generates additional data points from existing data to artificially enhance data size. Small data changes or deep learning algorithms may generate new data points. Data augmentation is a technique to artificially create new training data from existing training data. Fig. 4 displays diseased sugarcane test leaves. The proposed PI-RNN classifier is compared to support vector machine (SVM), neural network (NN), k nearest neighbor (KNN), stochastic gradient descent (SGD), AdaBoost, logistic regression (LR) and naive Bayes (NB) classifiers.

5.2 Orange Scoring Method

Scoring of a deep learning model is a necessary thing to do as it is very important to know how well a model is functioning. The Orange Data Mining Library provides following scoring methods on the basis of which a classifier works. The test and score widget is used for this purpose. It scores the classifiers on the basis of these classifiers by using scoring methods like AUC, accuracy, F-measure, precision and recall. The performance of proposed classifier is evaluated through different metrics are area under curve, accuracy, precision, f-measure and recall. The performance is high when its value is closer to 1 which is used to compute the area under curve. For effective calculation, accuracy is used in deep learning framework to classify healthy and unhealthy leaf. For balanced measure F measure is used. The best value is 1 and worst value is 0.

$$F_{measure} = \frac{P * R}{P + R} \quad (41)$$

The sum of T_p and F_p is denoted by number of true and false positives which is divided by T_p used for calculating precision.

$$P = \frac{T_p + F_p}{T_p} \quad (42)$$

The sum of T_p and F_N is denoted by number of true and false negatives which is divided by T_p used for calculating recall.

$$R = \frac{T_p + F_N}{T_p} \quad (43)$$

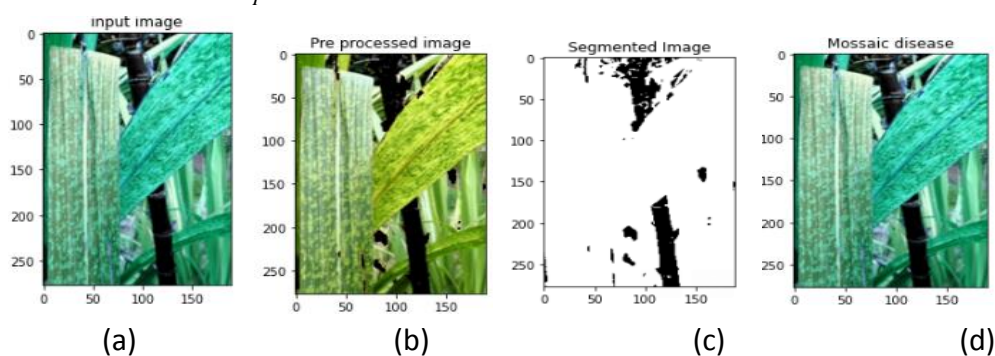


Fig. 3 Mosaic disease detection from sugarcane leaf (a) input image (b) pre processed image(c) segmented image and (d) mosaic prediction image respectively

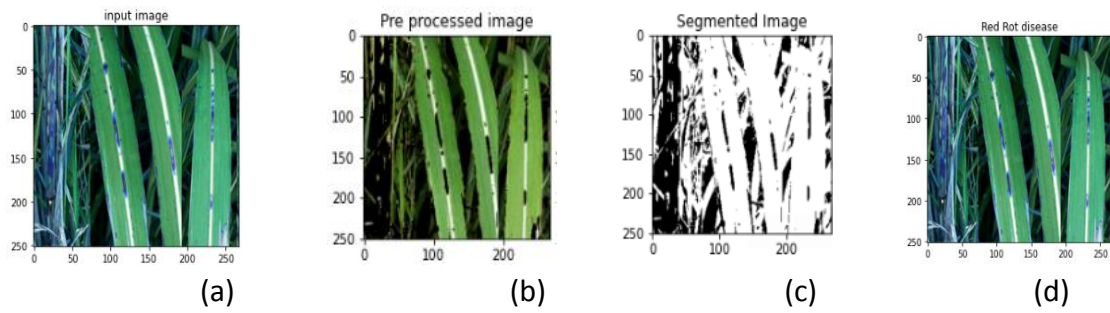


Fig. 4 Red root disease detection from sugarcane leaf (a) input image (b) pre processed image (c) segmented image and (d) Red root prediction image respectively

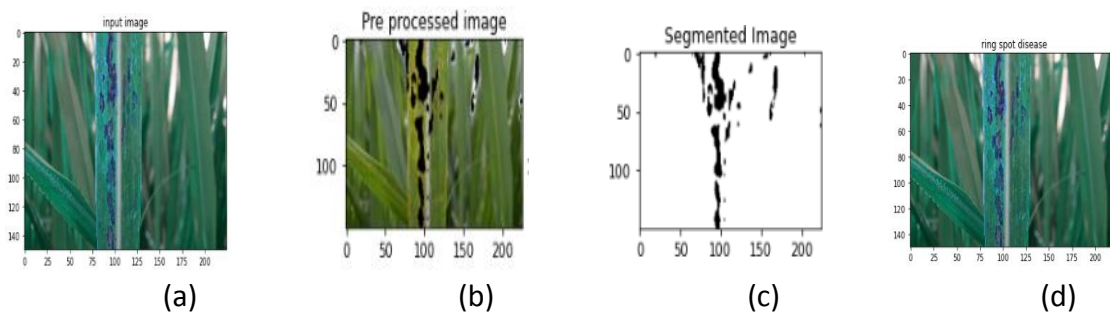


Fig. 5 Ring spot disease detection from sugarcane leaf (a) input image (b) pre processed image (c) segmented image and (d) ring spot prediction image respectively

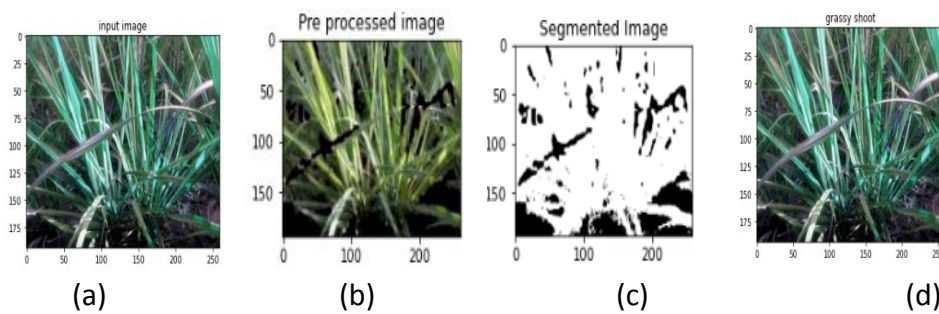


Fig. 6 Grassy shoot disease detection from sugarcane leaf (a) input image (b) pre processed image (c) segmented image and (d) grassy shoot prediction image respectively

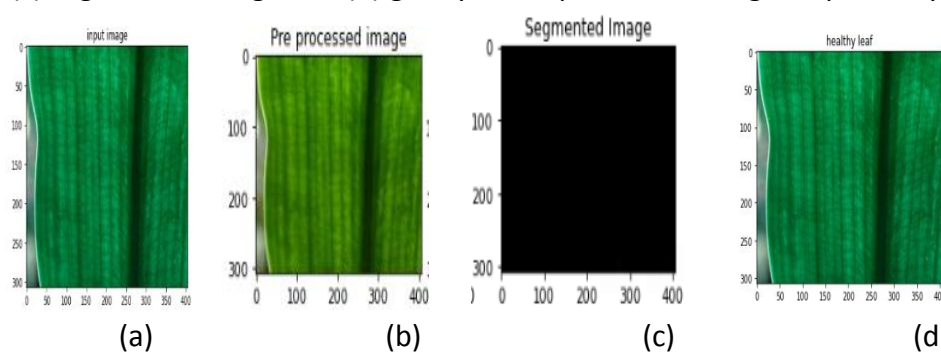


Fig. 7 Healthy disease detection from sugarcane leaf (a) input image (b) pre processed image (c) segmented image and (d) healthy image respectively

5.3 Comparative analysis of proposed and existing classifiers

Table 3 compares PI-RNN with existing classifiers. The PI-RNN classifier has a higher AUC than VCG16 LR, VCG19 LR, and inception V3 LR. The PI-RNN classifier has a better AUC than VCG16 SVM, VCG19 SVM, and inception V3 SVM. The PI-RNN classifier has a higher AUC than VCG16 SGD, VCG19 SGD, and inception V3 SGD. The PI-RNN classifier has a higher AUC than VCG16 NN, VCG19 NN, and inception V3 NN. The PI-RNN classifier has a higher AUC than VCG16 KNN, VCG19 KNN, and inception V3 KNN. The PI-RNN classifier has a higher AUC than VCG16 NB, VCG19 NB, and inception V3 NB. The PI-RNN classifier has a higher AUC than VCG16 Adaboost, VCG19 Adaboost, and inception V3 Adaboost.

Table 3 Comparative analysis of proposed and existing classifiers

Classifiers	Performance metrics (%)				
	Area under curve	Accuracy	F-measure	Precision	Recall
Inception v3 LR	91.7	83.8	83.7	83.8	83.8
Inception v3 SVM	89.7	81.9	81.8	82.7	81.9
Inception v3 SGD	81.9	81.9	81.9	81.9	81.9
Inception v3 NN	90.3	80.0	80.0	80.0	80.0
Inception v3 KNN	92.3	79.4	79.3	79.6	79.4
Inception v3 NB	80.2	75.6	75.6	75.8	75.6
Inception v3 Adaboost	76.2	72.5	72.5	72.6	72.5
VGG 19 LR	92.3	85.0	85.0	85.1	85.0
VGG 19 SVM	90.2	83.8	83.7	84.5	83.8
VGG 19 SGD	86.3	86.3	86.2	86.5	86.3
VGG 19 NN	90.9	84.4	84.4	84.4	84.4
VGG 19 KNN	90.8	83.1	83.0	83.8	83.1
VGG 19 NB	80.5	80.0	80.0	80.0	80.0
VGG 19 Adaboost	90.8	83.8	83.7	83.9	83.8
VGG 16 LR	88.9	83.1	83.1	83.1	83.1
VGG 16 SVM	90.2	84.4	84.3	85.0	84.4
VGG 16 SGD	80.6	80.6	80.6	80.7	80.6
VGG 16 NN	90.9	83.8	83.7	83.9	83.8
VGG 16 KNN	90.5	83.8	83.7	83.8	83.8
VGG 16 NB	81.6	80.6	80.6	80.7	80.6
VGG 16 Adaboost	89.8	82.5	82.5	82.7	82.5
PI-RNN	93.21	91.97	87.96	90.14	89.57



PI-RNN classifier accuracy is 9%, 7%, and 8% greater than VCG16 LR, VCG19 LR, and inception V3 LR. PI-RNN classifier accuracy is 8%, 8%, and 10% greater than VCG16 SVM, VCG19 SVM, and inception V3 SVM. PI-RNN classifier accuracy is 12%, 6%, and 10% greater than VCG16 SGD, VCG19 SGD, and inception V3 SGD. The proposed PI-RNN classifier is 8.8%, 8%, and 13% higher than VCG16, VCG19, and inception V3 NN. PI-RNN classifier accuracy is 8%, 9%, and 13% greater than VCG16 KNN, VCG19 KNN, and inception V3 KNN. The proposed PI-RNN classifier yields 12%, 13%, and 17% higher than VCG16 NB, VCG19 NB, and inception V3 NB. The proposed PI-RNN classifier achieves 10%, 8.8%, and 21% higher than VCG16, VCG19, and inception V3 Adaboost. F measure PI-RNN is 5%, 2%, and 4% higher than VCG16 LR, VCG19 LR, and inception V3 LR. The proposed PI-RNN classifier has a higher F measure than VCG16 SVM, VCG19 SVM, and inception V3 SVM. The proposed PI-RNN classifier has a higher F measure than VCG16 SGD, VCG19 SGD, and inception V3 SGD. The proposed PI-RNN classifier has a higher F measure than VCG16 NN, VCG19 NN, and inception V3 NN. The proposed PI-RNN classifier has a higher F measure than VCG16 KNN, VCG19 KNN, and inception V3 KNN. The proposed PI-RNN classifier has a higher F measure than VCG16 NB, VCG19 NB, and inception V3 NB. The proposed PI-RNN classifier has a higher F measure than VCG16 Adaboost, VCG19 Adaboost, and inception V3 Adaboost.

PI-RNN classifier has more precision than VCG16 LR, VCG19 LR, and inception V3 LR. PI-RNN classifier is 5%, 6%, and 8% more precise than VCG16 SVM, VCG19 SVM, and inception V3 SVM. PI-RNN classifier precision is 10%, 6%, and 9% greater than VCG16 SGD, VCG19 SGD, and inception V3 SGD. The proposed PI-RNN classifier is 6%, 6%, and 11% more precise than VCG16, VCG19, and inception V3 NN. The developed PI-RNN classifier has greater precision than VCG16 KNN, VCG19 KNN, and inception V3 KNN. PI-RNN classifier precision is 10%, 11%, and 15% greater than VCG16 NB, VCG19 NB, and inception V3 NB. The proposed PI-RNN classifier has higher precision than VCG16 Adaboost, VCG19 Adaboost, and inception V3 Adaboost. PI-RNN classifier has a better recall than VCG16 LR, VCG19 LR, and inception V3 LR. The proposed PI-RNN classifier has a greater recall than VCG16 SVM, VCG19 SVM, and inception V3 SVM. The proposed PI-RNN classifier has greater recall than VCG16 SGD, VCG19 SGD, and inception V3 SGD. The proposed PI-RNN classifier has greater recall than VCG16 NN, VCG19 NN, and inception V3 NN. The developed PI-RNN classifier has greater recall than VCG16 KNN, VCG19 KNN, and inception V3 KNN. Our proposed PI-RNN classifier has higher recall than VCG16 NB, VCG19 NB, and inception V3 NB. The proposed PI-RNN classifier has greater recall than VCG16 Adaboost, VCG19 Adaboost, and inception V3 Adaboost.

6. Conclusion

A hybrid machine learning algorithm for detecting sugarcane leaf disease (HML- SL) has been developed. One way is to segment out the diseased leaf tissue using the JAYA



algorithm. The second approach is to use enhanced coyote optimization to extract the most efficient features from the most optimal feature selection. Final classification of sugarcane leaf diseases is based on the presence of red rot, mosaic virus, ring spot, and grassy shoot using a neural network inspired by pigeons. Simulation accuracy of the proposed PI-RNN classifier is 9.7%, 8.5%, and 13% greater than existing approaches such as VGG16, VGG19, and Inception IV using classifiers, according to our results. Our proposed PI-RNN classifier achieves 93.21% AUC, 91.97% accuracy, 87.96 % F-measure, 90.14% Precision and 89.57% Recall for detecting sugarcane leaf disease. Thus agricultural producers can use the results of our research as it provides practical application for the sugarcane leaf disease detection as a precision agriculture system that predicts sickness by analyzing images of infected sugarcane leaf disease with Computer Vision, Image Processing, and Machine Learning algorithms.

Data availability statement:

Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Conflict of interest:

Dear editor

I have no conflict of interest

References

1. Bathrinath, S., M. Dhanasekar, B. Dhanorvignesh, Z. Kamaldeen, B. Santhi, R. K. A. Bhalaji, and K. Koppiahraj. "Modeling sustainability risks in sugar industry using AHP-BWM." *Materials Today: Proceedings* 50 (2022): 1397-1404.
2. Murugeswari, R., Z. Sharik Anwar, V. Raja Dhananjeyan, and C. Naveen Karthik. "Automated Sugarcane Disease Detection Using Faster RCNN with an Android Application." In *2022 6th International Conference on Trends in Electronics and Informatics (ICOEI)*, pp. 1-7. IEEE, 2022.
3. Balan, Sanju, K. Nithya, K. Anita Cherian, and R. Viswanathan. "True seed transmission of Sugarcane bacilliform virus (SCBV) in sugarcane." *Sugar Tech* 24, no. 2 (2022): 513-521.
4. Kumar, Sandeep, Arpit Jain, Anand Prakash Shukla, Satyendr Singh, Rohit Raja, Shilpa Rani, G. Harshitha, Mohammed A. AlZain, and Mehedi Masud. "A Comparative Analysis of Machine Learning Algorithms for Detection of Organic and Nonorganic Cotton Diseases." *Mathematical Problems in Engineering* 2021 (2021).
5. Kamal, Abdullah Bin, Muhammad Kamal Sheikh, Bismah Azhar, Muhammad Munir, Mirza Barjees Baig, and Michael R. Reed. "Role of Agriculture Extension in Ensuring



- Food Security in the Context of Climate Change: State of the Art and Prospects for Reforms in Pakistan." *Food Security and Climate-Smart Food Systems* (2022): 189-218.
6. Hechavarría, La O., Yaquelín Puchades Izaguirre, Yosel Pérez Pérez, Gabriela Michavila, Mario A. Casas González, Juana Pérez Pérez, Omelio Carvajal Jaime et al. "Assessment of sugarcane cultivars with stable reaction to *Xanthomonas albilineans* under mechanical inoculation conditions." *Tropical Plant Pathology* 47, no. 2 (2022): 214-221.
 7. Bera, Surojit, Trinetra Mukherjee, Subhabrata Das, Sandip Mondal, Suprabhat Mukherjee, and Sagnik Chakraborty. "Exploration of Extremophiles for Value-Added Products: A Step Towards Sustainable Bio-refinery." *Extremophiles* (2022): 110-134.
 8. Gopi, R., B. Mahendran, K. Chandran, M. Nisha, and R. Viswanathan. "Plant and Weather Factors on Resistance of *Saccharum officinarum* Germplasm Against Ring Spot Disease." *Sugar Tech* 23, no. 4 (2021): 720-729.
 9. Burbano, Roberto CV, Marcel F. da Silva, Alisson E. Coutinho, Marcos C. Gonçalves, Ivan A. dos Anjos, Luciana OS Anjos, Dilermando Perecin, and Luciana R. Pinto. "Marker-trait Association for Resistance to Sugarcane Mosaic Virus (SCMV) in a Sugarcane (*Saccharum* spp.) Panel." *Sugar Tech* (2022): 1-13.
 10. Gonzalez Viejo, Claudia, Damir D. Torrico, Frank R. Dunshea, and Sigfredo Fuentes. "Emerging technologies based on artificial intelligence to assess the quality and consumer preference of beverages." *Beverages* 5, no. 4 (2019): 62.
 11. Boudh, Siddharth, and Jay Shankar Singh. "Pesticide contamination: environmental problems and remediation strategies." In *Emerging and eco-friendly approaches for waste management*, pp. 245-269. Springer, Singapore, 2019.
 12. Salehi, Bahare, Alessandro Venditti, Claudio Frezza, Aysun Yüçetepe, Ümit Altuntaş, Sibel Uluata, Monica Butnariu et al. "Apium plants: Beyond simple food and phytopharmacological applications." *Applied Sciences* 9, no. 17 (2019): 3547.
 13. Razdan, Maharaj Krishen, and Dennis T. Thomas, eds. *Mulberry: Genetic Improvement in Context of Climate Change*. CRC Press, 2021.
 14. Huang, Ying-Kun, Wen-Feng Li, Rong-Yue Zhang, and Xiao-Yan Wang. "Diagnosis and control of sugarcane important diseases." In *Color Illustration of Diagnosis and Control for Modern Sugarcane Diseases, Pests, and Weeds*, pp. 1-103. Springer, Singapore, 2018.
 15. Zhang, Liangji, Guoxiong Zhou, Chao Lu, Aibin Chen, Yanfeng Wang, Liujun Li, and Weiwei Cai. "MMDGAN: A fusion data augmentation method for tomato-leaf disease identification." *Applied Soft Computing* (2022): 108969.
 16. Husin, Z. H., N. S. A. M. Taujuddin, S. Sari, A. R. A. Ghani, M. Siti Norsuha, R. Kogeethavani, and B. S. N. Aliah. "Bacterial Leaf Blight (BLB) Disease Detection Using



- Hue, Saturation, Value (HSV) Band Threshold Method." In Proceedings of the 12th National Technical Seminar on Unmanned System Technology 2020, pp. 403-420. Springer, Singapore, 2022.
17. Cosme, Fernanda, António Inês, and Alice Vilela. "Consumer's acceptability and health consciousness of probiotic and prebiotic of non-dairy products." *Food Research International* 151 (2022): 110842.
 18. Javidan, Seyed Mohamad, Ahmad Banakar, Keyvan Asefpour Vakilian, and Yiannis Ampatzidis. "Diagnosis of Grape Leaf Diseases Using Automatic K-Means Clustering and Machine Learning." Available at SSRN 4062708.
 19. Gao, Yang, Youxin Zhang, Jialin Yu, Li Liu, Yawen Wang, Jingxue Zhang, Minghui Chen et al. "Evaluation of shade tolerance of one annual and five perennial Medicago species under different artificial shade levels." *New Zealand Journal of Crop and Horticultural Science* (2022): 1-19.
 20. Abuelewa, Mohammed HS, Fadi ES Harara, Mustafa MK Al-Ghoul, Sami M. Okasha, and Samy S. Abu-Naser. "Rule Based System for Diagnosing Bean Diseases and Treatment." *International Journal of Engineering and Information Systems (IJEAIS)* 6, no. 5 (2022).
 21. Sanghera, Gulzar Singh, Aishwarya Saini, and Lenika Kashyap. "Physiological and biochemical traits associated with waterlogged tolerance in sugarcane (*Saccharum spp. hybrid complex*)." *Plant Physiology Reports* 27, no. 1 (2022): 44-55.
 22. Li, Dexian, Zixin Li, Xiaoxia Wang, Li Wang, Yonghong Li, and Deguang Liu. "Increasing risk of aphids spreading plant viruses in maize fields on both sides of China's Heihe-Tengchong line under climate change." *Pest Management Science* (2022).
 23. Luján, Adela M., Steve Paterson, Elze Hesse, Lea M. Sommer, Rasmus L. Marvig, M. D. Sharma, Ellinor O. Alseth et al. "Polymicrobial infections can select against *Pseudomonas aeruginosa* mutators because of quorum-sensing trade-offs." *Nature Ecology & Evolution* (2022): 1-10.
 24. Belay, Abebech Jenber, Ayodeji Olalekan Salau, Minale Ashagrie, and Melaku Bitew Haile. "Development of a chickpea disease detection and classification model using deep learning." *Informatics in Medicine Unlocked* (2022): 100970.
 25. Viswanathan, R., T. Ramasubramanian, C. Chinnaraja, R. Selvakumar, T. Lakshmi Pathy, K. Manivannan, and R. Nithyanantham. "Population dynamics of *Melanaphis sacchari* (Zehntner), the aphid vector of sugarcane yellow leaf virus under tropical conditions in India." *Tropical Plant Pathology* (2022): 1-18.
 26. Shruthi, U., V. Nagaveni, and B. K. Raghavendra. "A review on machine learning classification techniques for plant disease detection." In 2019 5th International conference on advanced computing & communication systems (ICACCS), pp. 281-284. IEEE, 2019.



27. Annabel, L. Sherly Puspha, T. Annapoorani, and P. Deepalakshmi. "Machine Learning for Plant Leaf Disease Detection and Classification—A Review." In 2019 International Conference on Communication and Signal Processing (ICCSP), pp. 0538-0542. IEEE, 2019.
28. Militante, Sammy V., and Bobby D. Gerardo. "Detecting Sugarcane Diseases through Adaptive Deep Learning Models of Convolutional Neural Network." In 2019 IEEE 6th International Conference on Engineering Technologies and Applied Sciences (ICETAS), pp. 1-5. IEEE, 2019
29. Militante, Sammy V., Bobby D. Gerardo, and Nanette V. Dionisio. "Plant leaf detection and disease recognition using deep learning." In 2019 IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE), pp. 579-582. IEEE, 2019.
30. Amarasingam, Narmilan, Surantha Salgadoe, Kevin Powell, Luis Felipe Gonzalez, and Sijesh Natarajan. "A review of UAV platforms, sensors, and applications for monitoring of sugarcane crops." *Remote Sensing Applications: Society and Environment* (2022): 100712.
31. Kumpala, Inchaya, Nontawat Wichapha, and Piyawat Prasomsab. "Sugar Cane Red Stripe Disease Detection using YOLO CNN of Deep Learning Technique." *Engineering Access* 8, no. 2 (2022): 192-197.
32. Balan, Sanju, K. Nithya, K. Anita Cherian, and R. Viswanathan. "True seed transmission of Sugarcane bacilliform virus (SCBV) in sugarcane." *Sugar Tech* 24, no. 2 (2022): 513-521.
33. Haregu, Simatsidk, Dawit Kidanemariam, and Adane Abraham. "Molecular identification and characterization of badnaviruses infecting sugarcane in Ethiopia." *Acta virologica* 66, no. 1 (2022): 3-10.
34. Zhao, Yong, Fenggang Zan, Jun Deng, Peifang Zhao, Jun Zhao, Caiwen Wu, Jiayong Liu, and Yuebin Zhang. "Improvements in Sugarcane (*Saccharum spp.*) Varieties and Parent Traceability Analysis in Yunnan, China." *Agronomy* 12, no. 5 (2022): 1211.
35. Srivastava, S., Kumar, P., Mohd, N., Singh, A. and Gill, F.S., 2020. A Novel Deep Learning Framework Approach for Sugarcane Disease Detection. *SN Computer Science*, 1(2), pp.1-7.
36. Suresh, G., Seetharaman, K.(2022). Real-time automatic detection and classification of groundnut leaf disease using hybrid machine learning techniques. *Multimed Tools and Applications* (2022), 1-29. <https://doi.org/10.1007/s11042-022-12893-1>. (SCI Indexed)
37. Seetharaman, K., Mahendran, T. (2022). Leaf Disease Detection in Banana Plant using Gabor Extraction and Region-Based Convolution Neural Network (RCNN).



Journal of Institution of Engineers (India) Series A (2022).
<https://doi.org/10.1007/s40030-022-00628-2>. (SCI Indexed)

38. Kalaivani, S., Seetharaman, K. (2022). A three-stage ensemble boosted convolutional neural network for classification and analysis of COVID-19 chest X-ray images, International Journal of Cognitive Computing in Engineering, 3, 35-45. (Scopus Indexed)
39. Seetharaman, K., Mahendran, T., (2021). Detection of Disease in Banana Fruit using Gabor Based Binary Patterns with Convolution Recurrent Neural Network, Turkish Online Journal of Qualitative Inquiry (TOJQI), 12(9), 6958 - 6966. (SCI Indexed)
40. Vasanthi, M., Seetharaman, K. (2021). A Hybrid Method for Biometric Authentication Oriented Face Detection Using Autoregressive Model with Bayes Backpropagation Neural Network, Soft Computing, 25, 1659–1680 DOI: 10.1007/s00500-020-05500-8 (ISSN: 1432-7643; SCI Indexed).
41. Seetharaman, K.(2019).Melanoma Image Classification Based on Color, Shape, and Texture Features Using Multivariate Statistical Tests: Journal of Computational and Theoretical Nanoscience, 16(4), 1717–1724. (ISSN: 1546-1955; SCI Indexed).

