Journal of Current Research in Engineering and Science
JCRES
Bi-Annual Online Journal (ISSN: 2581 - 611X)

VOLUME:9, ISSUE:1, JANUARY 2026, PAPER NUMBER:22

Advanced Agriculture Pest Prediction Framework Using Cloud-Based Salp Swarm Algorithm and Radial Basis Function Networks

P. Bharathi[1*], Dr. K. Dhanalakshmi[2],

[1*]*Research scholar*
Department of Computer Science Engineering,
Kongunadu Arts and Science College, Coimbatore-*641029.*
bharathipadma19@gmail.com

[2]*Associate Professor & Head, Department of Information Technology,*
Kongunadu Arts and Science College, Coimbatore-641029.
kdhanalakshmimca@gmail.com

Abstract: *Insect pests are one of the primary factors causing significant harm to agricultural crops of economic importance. Accurate agricultural pest prediction, critical for ensuring food security and a stable agricultural sector, relies on efficient pest classification. To improve the accuracy and stability of segmentation, a cognitive segmentation approach to pest images is presented in the paper. However, the wide variety of pest species and subtle differences between them make insect pest recognition challenging, often requiring specialized expertise. This research introduces a robust Feature Selection (FS) and Hybrid Deep Learning (HDL) model for efficient pest recognition using the IP102 dataset within a Cloud Computing (CC) framework. Initially, pre-processing is performed using an Adaptive Median Filter to remove noise and enhance classifier performance. Feature selection is then applied using the Salp Swarm Algorithm, SSAFS is employed to determine the ideal combination of handcrafted features to maximize classification success while minimizing the number of features. Segmentation results are achieved by k- Means clustering and continuously followed by a Radial Basis Function Network-based classifier for the identification of selected tea pests. The proposed model is evaluated using metrics such as accuracy, precision, recall, and F-measure, demonstrating superior performance compared to existing recognition models.*
Keywords: *Feature Selection (FS), Hybrid Deep Learning (HDL), Adaptive Median Filter, Salp Swarm Algorithm, k-Means clustering, Radial Basis Function Network-based classifier*

1. INTRODUCTION:

Agricultural pests pose a significant challenge to global food security and sustainable farming practices. These pests can lead to substantial crop losses, impacting the livelihood of farmers and the agricultural economy at large [1]. Accurate and timely prediction of pest outbreaks is crucial to mitigate these impacts and ensure effective pest control measures. In recent years, advancements in data-driven techniques have opened new avenues for improving pest prediction accuracy [2]. By leveraging environmental, meteorological, and crop-related data, predictive models can forecast pest outbreaks and aid in informed decision-making [3]. The integration of such predictive frameworks in agriculture has the potential to reduce pesticide usage, enhance crop yields, and promote eco-friendly farming practices are represented in Fig. 1.

Fig. 1. A sample of five classes of insect pest images; called aphids, flea beetles, Cicadellidae, flax budworm, and red spider mite.

Machine learning (ML) has revolutionized pest prediction by providing robust tools to analyze complex patterns in agricultural data [4]. ML models, such as Support Vector Machines, Artificial Neural Networks, and Decision Trees, have been employed to predict pest infestations based on historical data, weather conditions, and crop health indicators [5]. These models can adapt to diverse agricultural landscapes, offering predictions that are both scalable and localized. Furthermore, advanced techniques like ensemble learning and hybrid approaches have improved prediction accuracy by combining the strengths of multiple algorithms [6]. However, the success of these models heavily depends on the quality of data, model training, and the ability to capture the nonlinear relationships between variables [7].

Despite their promise, traditional ML models face several challenges in pest prediction. Data scarcity, noise, and imbalance in datasets often hinder model performance. In addition, the dynamic and unpredictable nature of pest behavior makes it difficult for static models to generalize effectively. Computational overhead and the lack of interpretability in some advanced models also limit their practical deployment in resource-constrained agricultural settings. Addressing these challenges requires the development of algorithms that are not only efficient but also capable of adapting to the variability inherent in agricultural systems.
The proposed framework leverages the Salp Swarm Algorithm (SSA) and Radial Basis Function Networks (RBFNs) to address the limitations of conventional ML models in pest prediction. SSA, a bio-inspired optimization algorithm, mimics the swarming behavior of salps to achieve global optimization, making it suitable for selecting optimal features and tuning model parameters. RBFNs, known for their ability to approximate nonlinear functions, provide an effective predictive mechanism for pest dynamics. The integration of SSA with RBFNs enables the development of a robust, adaptive, and computationally efficient pest prediction model. This hybrid approach not only enhances prediction accuracy but also offers interpretability and scalability, making it a valuable tool for sustainable agricultural practices.

2. LITERATURE REVIEW:

Karar et al [8] introduced a new mobile application to automatically classify pests using a deep-learning solution to support specialists and farmers. The developed application utilized a faster region-based convolutional neural network (Faster R-CNN) to accomplish the recognition task of insect pests based on cloud computing. Additionally, a database of recommended pesticides was linked with the detected crop pests to guide farmers. This study was successfully validated on five groups of pests: Aphids, Cicadellidae, Flax Budworm, Flea Beetles, and Red Spider. The proposed Faster R-CNN demonstrated the highest accurate recognition results of 99.0% for all tested pest images. Moreover, the deep learning method outperformed other previous recognition methods, including Single Shot Multi-Box Detector (SSD) MobileNet and traditional back-propagation (BP) neural networks. The main prospect of this study was to implement the developed application for online recognition of agricultural pests in both open-field settings, such as large farms, and greenhouses for specific crops. This approach provided a practical and efficient tool to enhance pest management practices for farmers and agricultural specialists.

Khan et al [9] introduced an optimized model was introduced, tailored specifically for UAV-based applications. Alterations to the YOLOv5s model, which included advanced attention modules, expanded cross-stage partial network (CSP) modules, and refined multiscale feature extraction mechanisms, enabled precise pest detection and classification. Inspired by the efficiency and versatility of UAVs, the study aimed to revolutionize pest management in sustainable agriculture while also detecting and preventing crop diseases. Rigorous testing was conducted on a medium-scale dataset, identifying five agricultural pests: ants, grasshoppers, palm weevils, shield bugs, and wasps. Comprehensive experimental analysis showcased superior performance compared to various YOLOv5 model versions. The proposed model achieved higher performance, with an average precision of 96.0%, an average recall of 93.0%, and a mean average precision (MAP) of 95.0%. Furthermore, the inherent capabilities of UAVs, combined with the YOLOv5s model tested in this study, demonstrated a reliable solution for real-time pest detection. This approach showed significant potential to optimize and improve agricultural production within a drone-centric ecosystem.

Balasubramaniam et al [10] proposed a machine learning-based system to address the problem of crop disease and pest prediction, focusing on the chilli crop as a case study. The performance of the proposed system was assessed using performance metrics such as accuracy, Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE). The experimental results revealed that the proposed method achieved an accuracy of 0.90, an MSE of 0.37, an MAE of 0.15, and an RMSE of 0.61. This model was designed to predict pests and diseases and notify farmers using a combination of the Random Forest Classifier, the AdaBoost Classifier, the K-Nearest Neighbour, and Logistic Regression. Among these methods, Random Forest was found to be the most accurate model.

Sharma and Rathore [11] proposed to improve crop yield prediction accuracy and facilitate decision-making for farmers. The system utilized a hybrid deep learning approach that combined convolutional neural networks (CNNs) and long short-term memory networks (LSTMs) to process multi-sensor data, including soil moisture data, weather data, and crop growth data. LSTMs were employed to capture temporal dependencies in the input data, while CNNs were used to extract spatial features. The system was implemented on a cloud platform, enabling farmers to access it from anywhere using a web-based interface. The system provided real-time crop yield predictions and alerted farmers to potential risks such as pests, diseases, and adverse weather conditions. Additionally, data visualization tools were incorporated, allowing farmers to monitor crop growth and make informed decisions regarding fertilization, crop management practices, and irrigation. Experimental results demonstrated that the proposed hybrid deep learning approach outperformed traditional machine learning methods for crop yield prediction, achieving a prediction accuracy of over 90%. The proposed model was shown to increase agricultural production, improve the quality and profitability of farming operations, and contribute to sustainable agriculture practices.

Lee and Yun [12] proposed a deep Learning techniques model that predict diseases through the previous growth environment information of crops, including air temperature, relative humidity, dew point, and CO2 concentration. By utilizing large-scale public data on crops such as strawberry, pepper, grape, tomato, and paprika, the model demonstrated its ability to predict the risk score of crop pests and diseases. It achieved high predictive performance with an average AUROC of 0.917. Based on the predicted results, the model helped prevent pests or facilitated post-processing measures. This environmental data-based crop disease prediction model and learning framework were expected to be universally applicable to

various facilities and crops for disease and pest prevention.
Palani et al [13] presented a new AI-driven crop disease and pest outbreak prediction model. The algorithm utilized satellite imagery, meteorological data, historical pest and disease incidence records, and field IoT sensor feeds to dynamically anticipate hazards. Recurrent neural networks (RNNs) for time-series data analysis, convolutional neural networks (CNNs) for high-resolution satellite imagery analysis, and other machine learning techniques were key elements of the system for detecting diseases and pests. The model's accuracy was carefully evaluated against historical epidemics, showing significantly better predictions than conventional methods. This approach enabled farmers to obtain early warnings and prevent biotic dangers from causing harm. The predictive insights supported agronomists and policymakers in maximizing pest and disease control, promoting sustainable agriculture and food security. To conclude, AI-driven predictive analysis had the potential to revolutionize agricultural pest and disease control. This study demonstrated how AI could transform crop harvests by integrating multiple data sources and cutting-edge algorithms. Future research aimed to improve the model's precision, expandability, and adaptability to different crops and regions.

Deshmukh et al [14] focused on utilizing the MobileNetV2 algorithm for pest classification, leveraging image reshaping and feature extraction techniques. The results indicated that MobileNetV2 outperformed other pre-trained models, achieving a higher accuracy of 0.95. By enhancing pest detection capabilities, AI-based technologies provided promising solutions to bolster agricultural production and mitigate economic losses.

PATIL et al [15] proposed a machine learning approach as a base for accurate predictions. Crop prediction was performed using a classification model, while yield prediction utilized regression models to learn from the data. Multiple ML models were analyzed based on performance metrics, and the best-performing model was incorporated into the backend. Among the models used for yield prediction, Random Forest Regression produced the best results with an MAE of 0.64 and an R^2 score of 0.96. For crop prediction, the Naïve Bayes classifier delivered the most accurate results with an accuracy of 99.39%. The study emphasized how machine learning could revolutionize crop management techniques by providing farmers with insights for optimizing resource allocation and boosting overall crop yield.

3. PROPOSED METHODOLOGY:

3.1 Proposed Approach for Pest Detection

The rapid advancement of digital technology has made image processing approaches employed in agricultural research to assist researchers in solving complex problems [16]. The analysis of the image offers a realistic chance for the automation of pest detection in crops. Automated pest identification is highly beneficial for producers with extensive agricultural lands and limited pest scouting skills. This paper proposes an efficient image processing-based approach utilizing a Hybrid Deep Learning algorithm to detect crop pests. The process flow of the proposed methodology is shown in Fig. 2.

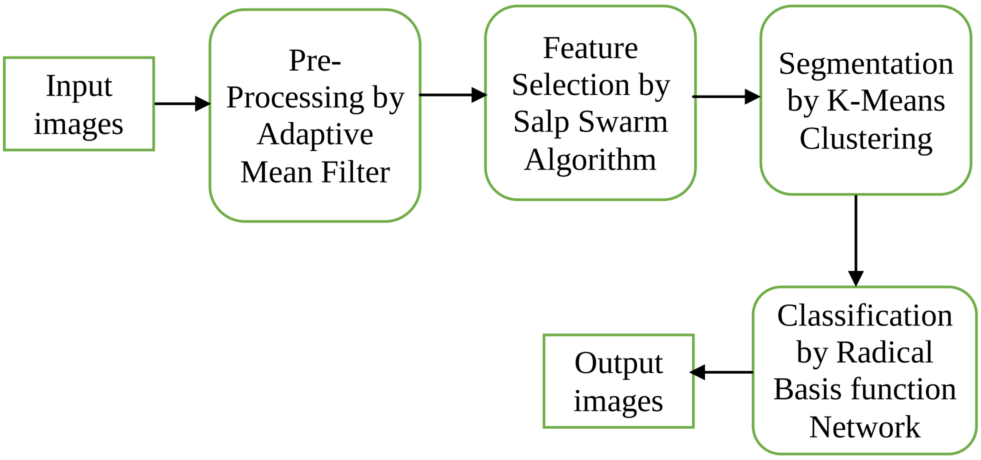

Fig. 2. Proposed Process Flow

An input crop image is initially exposed to pre-processing by an adaptive median filter, which performs denoising and reduces misclassified pixels. The filtered image is further segmented into multiple regions by an iterative process known as k- Means clustering, which is insensitive to rotation, scaling and absence of contrast. After the segmentation, the Radial Basis Function Network-based classifier approach describes features that retain the image details and performs effective pest detection with improved accuracy

3.2 Pre-processing by Adaptive Median Filter

The adaptive median filter employs noise detection and filtering algorithms to remove the impulsive noise. The window size utilized for image pixel filtering has adaptive characteristics. If a particular condition is not met, the window size is increased, whereas the median value of the window is utilized for filtering the pixel when the condition is met [17]. Consider P_{ij} as the corrupted image pixel, P_{min} as the minimum pixel value, P_{max} as the maximum pixel value, S as the current window size, S_{max} as the maximum window size attained and P_{med} as the assigned median of the window, the algorithm for filtering using an adaptive median filter requires two stages, described as follows.

Stage 1

i. If $P_{min}, P_{med}, P_{max},$, The median value is not an impulse; hence, the algorithm moves to stage 2 whether the present pixel is an impulse.
ii. Otherwise, the window size is increased, and stage 1 is repeated till the value of median does not equal an impulse enabling the algorithm to move to stage 2; else, the window size of the maximum value is attained in which the value of median is assigned to the filtered value of image pixel.

Stage 2:

i. If $P_{min}, P_{med}, P_{max},$ The present value of a pixel is not an impulse; hence, the image pixel filtered remains unchanged.
ii. Otherwise, the pixel value of the image is equal to either P_{max} or P_{min} and hence, the image pixel filtered is assigned to the median value from stage 1.

Initially, the size of the filtering window of each noise pixel is selected as 3 × 3: When the count of uncorrupted window pixels equals at least three, the median of the uncorrupted pixels substitutes the corrupted pixel. Otherwise, the window sides extend iteratively outward by one pixel until the uncorrupted pixel count equals at least three. Subsequently, all pixel medians replace the noise pixel in the filtering window in the following iterations; hence, the image's filtering is adaptive.

3.3 Feature selection by Salp Swarm Algorithm (SSA)

Feature selection with the SSAFS algorithm Salp swarm algorithm (SSA), is a relative new metaheuristic algorithm. The inspiration of SSA is the swarming behavior of the sea organism called salps. The salps are barrel-shaped, free-floating tunicates from the family of Salpidae [18]. When navigating and foraging in the ocean, salps often float together in chains of salps. The basic idea behind the SSA algorithm is to imitate the swarming behavior of salps in the deep oceans based on the salps chain. The salp at the head of the chain acts as the leader, and the following salps are followers. Each individual represents a candidate solution for the targeted problem (food source). A population of N salp individuals is defined as a 2-dimensional matrix $X = \{x_1, \ldots. x_i \ldots. x_N\} = \{x_j^i | 1 \leq i \leq N, 1 \leq j \leq D\}$. x_j^i denotes the position of the leader at the jth dimension, x_j^i presents the position of the ith follower at the jth dimension $(2 \leq i \leq N)$.when SSA algorithm is used for feature selection problems, all solutions are limited to the binary values, i.e., $x_j^i \epsilon [0,1]$. x_j^i Represents the jth feature (color or texture) on the ith image sample. If $x_j^i = 1$, the jth feature is selected. The feature selection framework based on the SSA feature selection algorithm is shown in figure.

Population initialization

As shown in Fig. 3, the first step of SSAFS is population initialization. In this step, a swarm of N salp individuals is randomly generated. The quality of initial population is closely related to the convergence speed of the algorithm. The chaotic map is a nonlinear dynamic system that could generate random numbers with special dynamics characteristics. The generated random numbers exhibit non-repetitiveness, ergodicity, regularity, and unpredictability. In our study, chaotic map with ergodic property is employed to initialize uniform distributed salp to improve solutions diversity. In the original SSA, the initial state of the ith salp is defined as Equation (1).

$$x_j^i(k) = \left(ub_j(k) - lb_j(k)\right) * o^k + lb_j(k) \tag{1}$$

Where $x_j^i(k)$ represents the position of the ith follower in the jth dimension space at the iteration k; $ub_j(k)$ and $lb_j(k)$denote the upper and lower bounds of the jth dimension, respectively. Equation (2) describes the expression of the logistic mapping.

$$o^{k+1} = \mu o^k(1 - o^k) \tag{2}$$

Where o^k represents the state of a variable at time step k, o^{k+1} represents the state of the variable at the next time step k+1, μ describes a parameter that controls the behavior of the system. $(1 - o^k)$ Introduces a nonlinearity in the recurrence relation. Considering the fact that feature selection is a binary question, need to define a transfer function to make the binary version of SSA from the continuous version. Therefore, the variable $x_j^i(k)$ shown in Eq. 1 will be further transferred as Boolean state through the following Equations (3) and (4):

$$x_j^i(k) = \begin{cases} 1, T\left(x_j^i(k)\right) \geq C \\ 0, T\left(x_j^i(k)\right) < C \end{cases} \tag{3}$$

$x_j^i(k)$ Represents a variable indexed by $i, j, k,$ which likely represents a binary decision (taking values 0 or 1). $Tx_j^i(k)$ This seems to be a function or transformation applied to $x_j^i(k)$. C0 and C threshold constants that define conditions for $x_j^i(k)$.

$$T\left(x_j^i(k)\right) = \frac{1}{1+exp^{-x_j^i(k)}} \tag{4}$$

In Equation (4), $T\left(x_j^i(k)\right)$ represented a transformation function applied to $x_j^i(k)$, $exp^{-x_j^i(k)}$ represents the exponential function, which decays rapidly for large positive values of x and grows for large negative values of x.

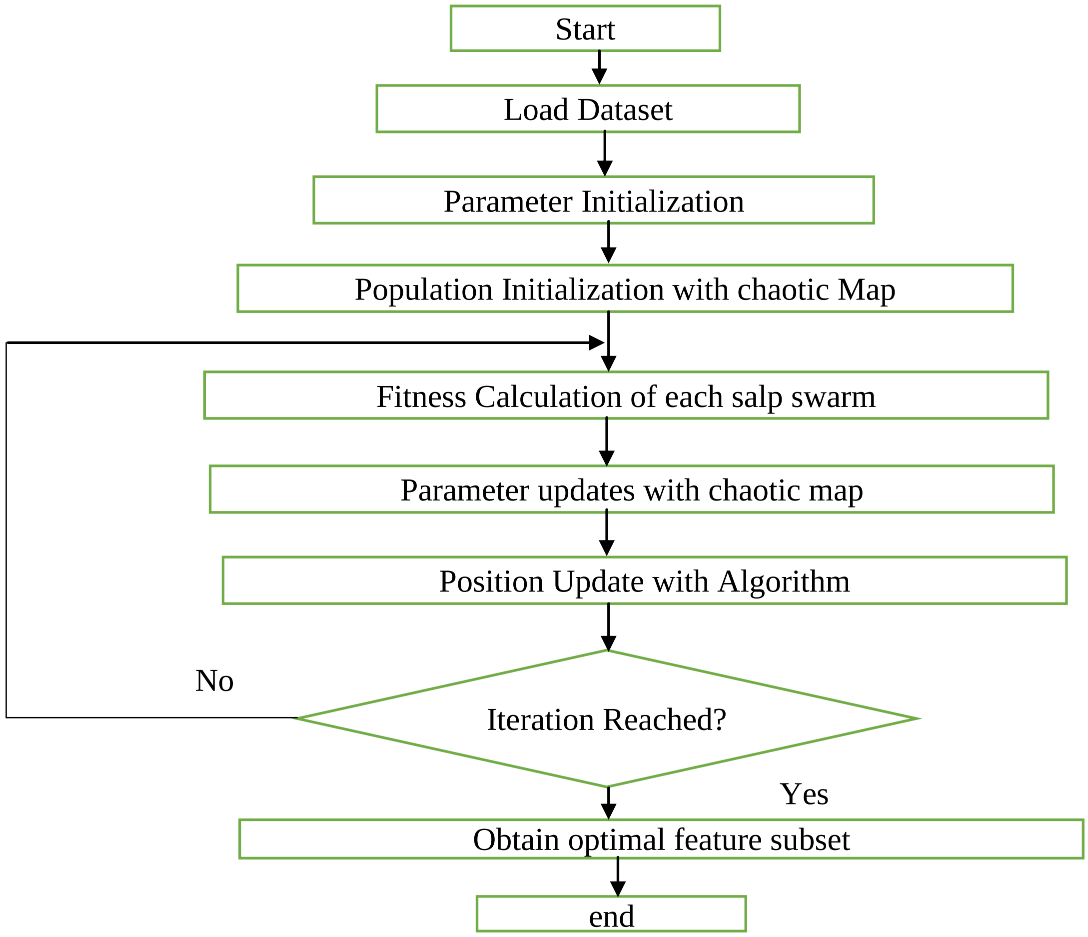

Fig. 3. The flowchart of the proposed algorithm SSAFS.

Fitness calculation

In SIAs, the fitness function is an important metric to evaluate the strengths of individuals within a population. The fitness value reflects the goodness of fit of each candidate's solution to the targeted question. Therefore, the selection of the fitness function determines the balance of the multi-objective algorithm in the optimization process. As a multi-objective problem, features selection try to minimize the subset of selected features and maximize the accuracy of the output for a given classifier, simultaneously. According to the above basis, the fitness function for deter mining solutions in this situation built to achieve a balance between 2 objectives is defined in Equation (5).

$$Fitness \ = \rho Err(FS) + \phi \frac{|FS|}{N} \tag{5}$$

Where the function $Err(*)$ denotes the classification error of the potential feature subset FS, and | FS | and N denote the number of selected features and the total number of features. The coef ficients ρ and ϕ are the balance parameters that control the classification accuracy and the rate of features being selected. In addition, ρ and ϕ satisfy that $\rho \in [0,1]$ and $\rho + \phi = 1$. A smaller fitness value is better. K-NN algorithm is a non-parametric supervised learning algorithm. It relies on the closest k labeled instances (neigh bors) to learn a function that produces an appropriate prediction for a given unlabelled example. Here, the k-NN model was employed as the classification method to evaluate the feature subsets generated by SSAFS, where k = 3. Specifically, it uses 5-fold cross-validation of k-NN to calculate the value $Err(*)$ of a feature subset

Population Evaluation

The role of the leader of the salp chain is to search for the food source. Hence, the position of the leader is dynamically updated based on the location of the food source. In the original SSA, the leader is updated by using the following Equation (6):

$$x_j^1(k) = \begin{cases} F_j(k) + c_1 * \left[\left(ub_j(k) - lb_j(k) * c_2 + lb_j(k)\right)\right], c_3 \geq 0.5 \\ F_j(k) - c_1 * \left[\left(ub_j(k) - lb_j(k) * c_2 + lb_j(k)\right)\right], c_3 < 0.5 \end{cases} \quad (6)$$

Where $x_j^1(k)$ denotes the position of the leader in the jth dimensional space in the kth iteration, and *F(k) j* represents the position of the food source. $ub_j(k)$ and $lb_j(k)$ are the upper and lower boundaries, respectively. Parameters c_2 and c_3 control the step size and the directions of the next move, respectively $(c_2, c_3 \in [0,1])$. specifically, c_1 is defined for balancing global search and local search capabilities.

$$c_1 = 2e^{-\left(\frac{4k}{K}\right)^2} \quad (7)$$

Where k is the index of the current iteration, and K is the maximum of iterations. From Equation (7), that found that the position of leader is mainly determined by the position of the current optimal food source. Once the leader falls into a local optimum, the whole population will fall into local stagnation.

3.4 Segmentation By K-Mean Clustering

K-means clustering is an unsupervised classification algorithm based on a clustering technique. The algorithm divides data into a predetermined class K based on minimizing the error function and is widely used [19]. To accurately segment whiteflies from crop leaf images, an improved K-means clustering algorithm was adopted and is formulated as Algorithm 1.

Algorithm 1 Improved K-Means Clustering Algorithm
Input:
Number of desired output clusters, k;
Pixels of pest image, x_i.
Output:
Number of the cluster assigned to each input pattern
1: Step 1: Initial cluster centres, $\mu_1, \mu_2, \ldots, \mu_k$, adaptively learned based on x_i
2: Step 2. For each x_i do
3: Compute Euclidean distance between x_i to μ_i;
4: Classify x_i according to nearest μ_i;
5: end for
6: Step 3. Recompute μ_i according to the mean of the k-clusters
7: Step 4. Computer Manhattan distance (L) between each class according to the mean of a* component of each cluster
8: Step 5. Repeat steps 2-5 until max(L)

The selection of initial cluster centres is very important for accurate segmentation of insect pest images. If a fixed classification centre is used, adaptability of the algorithm will be a problem. For randomly selected initial cluster centres, K- means does not guarantee unique clustering.

3.5 Classification by A Radial Basis Function Network-Based Classifier

Radial Basis Function network creates a number of radial structures that are used to classify the input vector into its corresponding output class. The output class is evaluated depending the Euclidian distance of the input vector and the stored vector usually called the centre [20]. The radial basis function network consists of three different layers; an input layer, a single hidden layer and an output layer. The weights between the input and the hidden layers are fixed to 1, the weights from the hidden to output layer tend to vary from one iteration to another during the learning phase. The Hidden layer stores the radial centres along with the radius of each cluster (class). At first the input vector is used to calculate the Euclidian distance from the stored vector, this distance parameter zi is then fed to an activation function to generate the intermediate result. The final output of the network is obtained by obtaining the weighted sum from the hidden to output layers and a linear activation function in the Fig. 4.

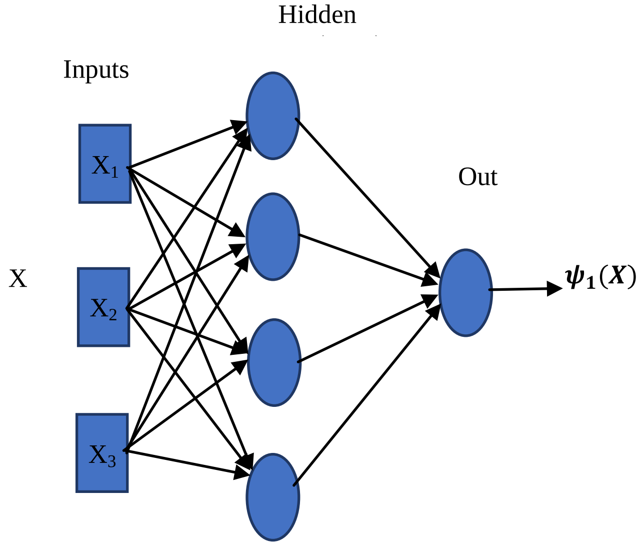

Fig. 4 Radial Basis Function networks

Different types of activation functions can be used in Radial Basis Function networks such as Gaussian, Multiquadrics and Inverse Multiquadric. In our current experiment have used the Gaussian Activation function because the other two functions require the centres to be different or non-singular where as our dataset being a real field experimental data contains redundant information which can only be captured by Gaussian distribution. Similarly, several learning mechanisms for Radial Basis Function networks have been suggested such as Pseudo Inverse, Gradient Descent and Hybrid Leaning. It has employed supervised Gradient Descent technique to train our network as the data was obtained from different tea gardens in a cumulative manner. Using Gradient Descent rule or commonly known as Error Back-propagation rule can adjust the free parameters like the weights (W) from the hidden to output layer, the radial spreads σ and the radial centres C in a supervised manner. The biggest advantage of this method is that it helps us to track the error in the network with each iteration or epoch and adjust the parameters accordingly. In our current experiment have used 31 input parameters each representing the refined features and 3 output classes corresponding to the three pests to be identified. Before training the network, have initialized the radial centres C in the hidden layer by random distribution of the input vectors.

In the training phase, the Euclidian distance between the input pattern i and the radial centres are obtained as follows the equation (8):

$$z_i = \sqrt{\sum (x - x_i)^2} \tag{8}$$

z_i Represents the computed result (possibly a distance or error measure). $x - x_i$ Represents the data points (could be scalar values or vectors). $\sum$ Indicates summation over multiple terms. $(x - x_i)^2$ Computes the squared difference between x and x_i.

The weighted sum ($\Sigma \emptyset W$) is calculated and then added with a bias (B) value to get the value of the final output.

$$Output \;=\; \sum \emptyset W + B \tag{9}$$

Equation (9) describes after getting the output, ∅ symbol means an empty set, which does not contribute to summation, the error (ε) in the result has been obtained by using RMS rule.
For each pattern, it repeats all the steps until and unless a termination condition is reached; for example, number of epochs <=40.

4. RESULT AND DISCUSSION:

The suggested HDL- SSA approach is assessed by contrasting its results with those of the current classifiers. In addition to classification accuracy, the classifier is evaluated using the average outcomes for each classifier and the statistical metrics provided in equations (10-13).

The ratio of correctly identified positive results to all expected positive information is termed as precision.

$$Precision = TP/TP + FP \tag{10}$$

The ratio of correctly identified positive results to all data in the definite class is termed as sensitivity.

$$Recall = TP/TP + FN \tag{11}$$

The weighted average of Precision and Recall is known as the F1 score. That requires both outcomes as FN (False Negatives), and FP (False Positives).

$$F1\,Score = 2 * (Recall * Precision) / (Recall + Precision) \tag{12}$$

The following formula determines accuracy by the positives and negatives:

$$Accuracy = (TP + FP)/(TP + TN + FP + FN) \tag{13}$$

Here TP stands for True Positive, FP stands for False Positive, TN stands for True Negative, and FN stands for False Negative, respectively. Every standard and suggested classifier has these parameter values computed.

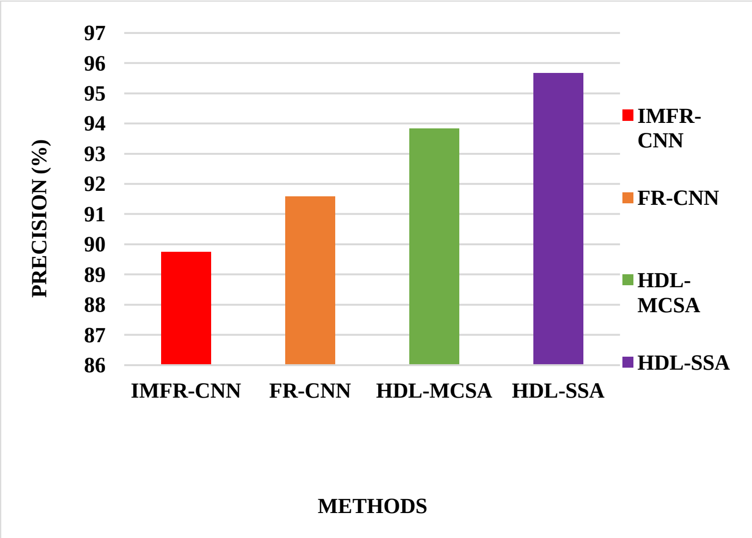

Fig. 5. Results of a precision comparison between the proposed HDL-SSA and the current classifiers

Fig. 5 represents the bar graph compares the precision percentages of four methods—IMFR-CNN, FR-CNN, HDL-MCSA, and HDL-SSA—showing a progressive improvement from IMFR-CNN (approximately 89%) to HDL-SSA (around 96%). Among the methods, HDL-SSA achieves the highest precision, significantly outperforming the others, while IMFR-CNN exhibits the lowest. FR-CNN and HDL-MCSA demonstrate intermediate performance, with precision values of approximately 92% and 94%, respectively. The results indicate that HDL-SSA is the most effective method in terms of precision, highlighting its superior capability compared to the others.

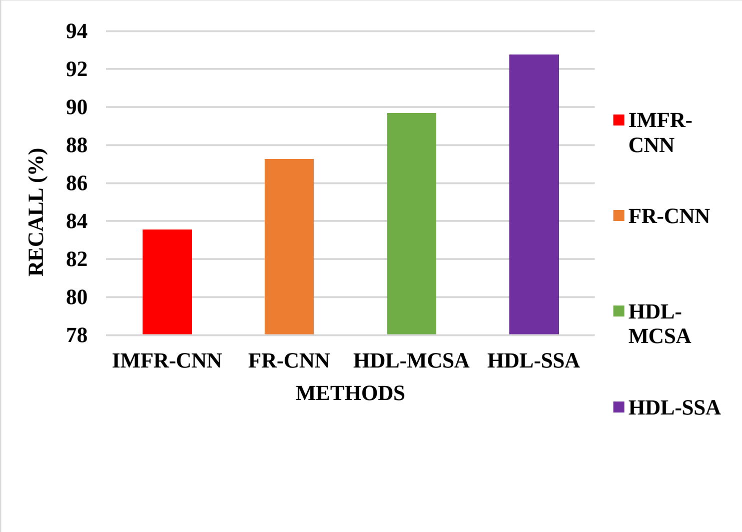

Fig. 6. Results of a Recall comparison between the proposed HDL-SSA and the current classifiers

Fig. 6 describes the bar graph illustrates the recall percentages of four methods: IMFR-CNN, FR-CNN, HDL-MCSA, and HDL-SSA. The recall increases progressively from IMFR-CNN, which has the lowest value of around 83%, to HDL-SSA, which achieves the highest at approximately 92%. FR-CNN shows a noticeable improvement over IMFR-CNN with a recall of around 87%, while HDL-MCSA further improves to about 89%. HDL-SSA outperforms all other methods, indicating its superior ability to identify relevant instances. The graph highlights the consistent enhancement in recall across the methods, with HDL-SSA demonstrating the most effective performance.

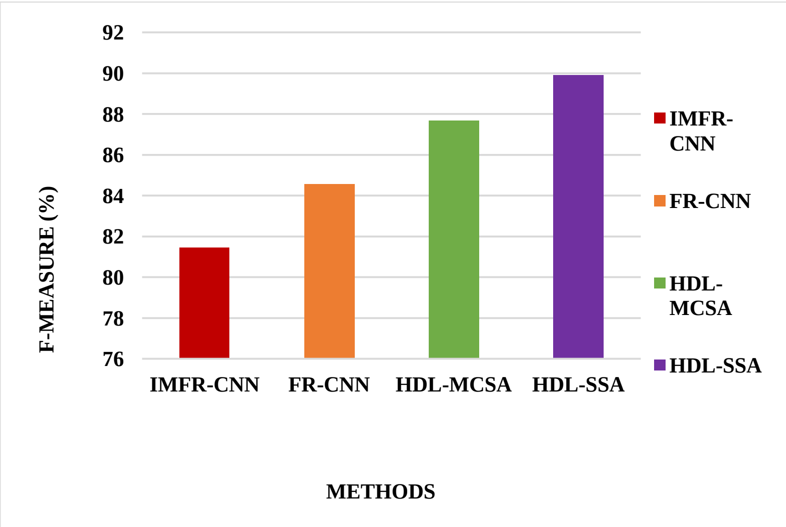

Fig. 7. Results of a F-Measure comparison between the proposed HDL-SSA and the current classifiers

Fig. 7 shows the bar graph compares the F-measure percentages of four methods: IMFR-CNN, FR-CNN, HDL-MCSA, and HDL-SSA. The F-measure steadily increases across the methods, starting from approximately 81% for IMFR-CNN, improving to around 85% for FR-CNN, and further increasing to approximately 88% for HDL-MCSA. HDL-SSA achieves the highest F-measure, reaching about 91%. This progression demonstrates the improved balance between precision and recall as the methods advance, with HDL-SSA emerging as the most effective technique in terms of overall performance.

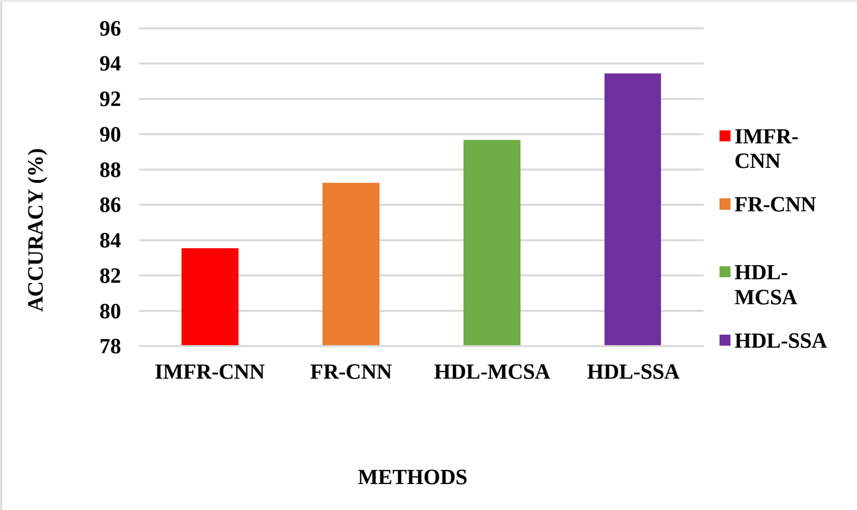

Fig. 8. Results of Accuracy comparison between the proposed HDL-SSA and the current classifiers

Fig. 8 represents the bar graph illustrates the accuracy percentages of four methods: IMFR-CNN, FR-CNN, HDL-MCSA, and HDL-SSA. The accuracy shows a progressive improvement, starting with IMFR-CNN at approximately 83%, increasing to about 87% for FR-CNN, and further improving to around 90% for HDL-MCSA. HDL-SSA achieves the highest accuracy at approximately 92%. This consistent upward trend highlights the increasing effectiveness of the methods, with HDL-SSA demonstrating superior performance in achieving the highest accuracy among the evaluated techniques.

5. CONCLUSION:

In conclusion, the proposed framework for advanced agriculture pest prediction leverages the Salp Swarm Algorithm (SSA) and Radial Basis Function Networks (RBFNs) within a cloud computing environment to deliver highly accurate and efficient pest prediction. By integrating SSA for feature optimization and RBFNs for robust classification, the framework significantly enhances prediction accuracy while reducing computational complexity. The use of cloud computing enables real-time data processing, scalability, and accessibility, making it highly suitable for large-scale agricultural applications. This approach not only supports timely pest management decisions but also contributes to sustainable farming practices by reducing the overuse of pesticides. For future work, the framework can be extended to incorporate multi-source data inputs, such as drone imagery and IoT sensor data, to improve predictive accuracy further. Additionally, integrating explainable AI techniques will enhance interpretability, allowing farmers to better understand the underlying factors influencing pest outbreaks. Finally, the development of a user-friendly mobile application for real-time pest monitoring and prediction will ensure the framework's accessibility and practicality for end-users.

6. REFERENCES:

[1] A. B. Kathole, J. Katti, S. Lonare, and G. Dharmale, “Identify and classify pests in the agricultural sector using metaheuristics deep learning approach,” Franklin Open, 2023, vol. 3, pp. 100024.

[2] N. Tantalaki, S. Souravlas, and M. Roumeliotis, “Data-driven decision making in precision agriculture: The rise of big data in agricultural systems,” Journal of agricultural & food information, 2019, vol. 20, no. 4, pp. 344-380.

[3] Q. Xiao, W. Zheng, Y. He, Z. Chen, F. Meng, and L. Wu, “Research on the Agricultural Pest Identification Mechanism Based on an Intelligent Algorithm,” Agriculture, 2023, vol. 13, no. 10, pp. 1878.

[4] P. B. Prithvi, F. Zahin, and S. S. Anny, “Pest detection system using machine learning techniques (Doctoral

dissertation, Brac University).

[5] D. Marković, D. Vujičić, S. Tanasković, B. Đorđević, S. Ranđić, and Z. Stamenković, “Prediction of pest insect appearance using sensors and machine learning,” Sensors, 2021, vol. 21, no. 14, pp. 4846.

[6] A. K. Singh, M. Yeasin, R. K. Paul, A. K. Paul, and A. Sarkar, “Dynamic Ensemble based Machine Learning Models for Predicting Pest Population,” Frontiers in Applied Mathematics and Statistics, 2024, vol. 10, pp. 1435517.

[7] E. G. Frank, “The economic impacts of ecosystem disruptions: Costs from substituting biological pest control,” Science, 2024, vol. 385, no. 6713, pp. eadg0344.

[8] M. E. Karar, F. Alsunaydi, S. Albusaymi, and S. Alotaibi, “A new mobile application of agricultural pest’s recognition using deep learning in cloud computing system,” Alexandria Engineering Journal, 2021, vol. 60, no. 5, pp. 4423-4432.

[9] A. Khan, S. J. Malebary, L. M. Dang, F. Binzagr, H. K. Song, and H. Moon, “AI-Enabled Crop Management Framework for Pest Detection Using Visual Sensor Data,” Plants, 2024, vol. 13, no. 5, pp. 653.

[10] S. Balasubramaniam, S. G. Nelson, M. Arishma, and A. S. Rajan, “Machine Learning based Disease and Pest detection in Agricultural Crops,” EAI Endorsed Transactions on Internet of Things, 2024, vol. 10, pp. 1-8.

[11] A. K. Sharma, and A. S. Rathore, “AGRO-Cloud Model and Smart Algorithm to Increase Crop Yield Prediction to Improve Agriculture Quality,” Online Social Networks in Business Frameworks, 2024, pp. 371-388.

[12] S. Lee, and C. M. Yun, “A deep learning model for predicting risks of crop pests and diseases from sequential environmental data,” Plant Methods, 2023, vol. 19, no. 1, pp. 145.

[13] H. K. Palani, S. Ilangovan, P. G. Senthilvel, D. R. Thirupurasundari, and R. Kumar, “AI-Powered Predictive Analysis for Pest and Disease Forecasting in Crops,” In 2023 International Conference on Communication, Security and Artificial Intelligence (ICCSAI), 2023, pp. 950-954.

[14] G. Deshmukh, S. Gorde, K. Lunge, N. Labade, and V. S. Mahalle, “Detection of Insects and Pests in Agriculture field using MobileNet,” (Doctoral dissertation, Sant Gadge Baba Amravati University).

[15] P. Patil, P. Athavale, M. Bothara, S. Tambolkar, and A. More, “Crop Selection and Yield Prediction using Machine Learning Approach,” Current Agriculture Research Journal, 2023, vol. 11, no. 3, pp. 968-980.

[16] L. Nanni, G. Maguolo, and F. Pancino, “Insect pest image detection and recognition based on bio-inspired methods,” Ecological Informatics, 2020, vol. 57, pp. 101089.

[17] B. Divya, and M. Santhi, “Automatic Detection and Classification of Insects Using Hybrid FF-GWO-CNN Algorithm,” Intelligent Automation & Soft Computing, 2023, vol. 36, no. 2, pp. 1881-1898.

[18] X. Xie, F. Xia, Y. Wu, S. Liu, K. Yan, H. Xu, and Z. Ji, “A novel feature selection strategy based on Salp swarm algorithm for plant disease detection,” Plant Phenomics, 2023, vol. 5, pp. 0039.

[19] Z. Wang, K. Wang, Z. Liu, X. Wang, and S. Pan, “A cognitive vision method for insect pest image segmentation,” IFAC-PapersOnLine, 2018, vol. 51, no. 17, pp. 85-89.

[20] G. Banerjee, U. Sarkar, and I. Ghosh, “A radial basis function network-based classifier for detection of selected tea pests,” International Journal, 2017, vol. 7, no. 5, pp. 665-669.