Robust Maize Leaf Disease Detection and Classification in Agriculture using Enhanced Machine Learning Models

Olusola Bamidele Ayoade, PhD
Emmanuel Alayande University of Education, Oyo
Department of Data Science, Informatics & Computer Science

Mumini Oyetunji Raji, PhD
Emmanuel Alayande University of Education, Oyo
Department of Data Science, Informatics & Computer Science

Aminat Adejoke Akindele
Emmanuel Alayande University of Education, Oyo
Department of Data Science, Informatics & Computer Science

Kemi Jemilat Yusuf-Mashopa
Emmanuel Alayande University of Education, Oyo
Department of Data Science, Informatics & Computer Science
yusuf-

Muinat Folake Abdulrauff
Emmanuel Alayande University of Education, Oyo
Department of Data Science, Informatics & Computer Science

Ibrahim Adebayo Raji
Emmanuel Alayande University of Education, Oyo
Department of Data Science, Informatics & Computer Science

Fatima Bolanle Musah
Emmanuel Alayande University of Education, Oyo
Department of Data Science, Informatics & Computer Science

ABSTRACT
Maize (corn) is a major and high-yielding crop, cultivated worldwide, although diseases may cause severe yield reductions. Monitoring and identifying maize diseases throughout the growth cycle are crucial tasks. Accurately detecting diseases is an issue for farmers who need expertise in plant pathology, while professional diagnosis can be time-consuming and expensive. Meanwhile, conventional machine learning techniques based on optimised support vector machine with optimisation techniques and image recognition models were impacted with issues which affecting their performance in the classification task. Therefore, this study proposes an improved machine learning model based on optimised support vector machine (SVM) (IMLMBOSVM) using enhanced Binary Particle Swarm Optimisation and Enhanced Reptile search Algorithm to optimise the support vector machine. The aim of the (IMLMBOSVM-EBPSO) and (IMLMBOSVM-ERSA) methods is to detect and categorise maize leaf diseases. The IMLMBOSVM-EBPSO and IMLMBOSVM-ERSA methods apply luminosity to convert RGB images to grayscale images, bi-histogram equalisation for contrast enhancement of the images, morphological filtering for sharpening of the images, adaptive median filtering for noise removal, Sobel edge detection method for segmenting lesions from uninfected part of the leaf, gray level co-occurrence matrix to extract both texture, shape features and colour moment to extract colour features, and the EBPSO or ERSA technique for hyperparameters tuning of the SVM. The IMLMBOSVM-EBPSO and IMLMBOSVM-ERSA techniques exploit EBPSO-SVM and ERSA-SVM for maize leaf disease classification, respectively. Experimental evaluation was conducted to validate the IMLMBOSVM-EBPSO and IMLMBOSVM-ERSA approaches, and the results show that IMLMBOSVM-EBPSO achieved a false positive rate of 2.48%, a specificity of 97.53%, a sensitivity of 97.33%, a precision of 97.84%, and an accuracy of 97.41%. The IMLMBOSVM-ERSA achieved a false positive rate of 3.60%, a specificity of 96.40%, a sensitivity of 96.34%, a precision of 96.85%, and an accuracy of 96.36%.

General Terms
Pattern Recognition

Keywords
Adaptive Median Filtering, Classification Task, Colour Features, GrayScale Image, Lesion

1. INTRODUCTION
Maize leaf diseases drastically decrease crop production; thus, monitoring and detecting diseases during the growing season is essential [1]. Conventionally, plant pathologists, field experts, or cultivators analyse all diseases by physically examining the signs of the crop's diseases with the naked eye [2]. This technique is not possible at a higher level due to limitations, such as physical accessibility, resource availability, cost, and time [3]. Often, the unavailability of field experts can prevent the precise therapy of the ailments in the earlier phases. Hence, a cost-effective and fast technique for diagnosing crop diseases is needed [4]. In the existing conditions, automatic disease identification employing machine learning techniques surpasses the traditional disease identification approach in terms of accuracy, cost-effectiveness and effectiveness. A digital image-assisted automated detection model in maize crops can be a sustainable option for reaching the stakeholders, namely, maize cultivators and the country's massive population [5].

Recently, numerous machine learning models based on support vector machines have been explored to identify a variety of diseases affecting maize plants. For instance, Rajagopal [6] developed a classification model using leaf images for automatic classification. The study leverages the advantages of

Legion Kernels and Parallel Support Vector Machine (LK-PSVM) clubbed with fuzzy C means Image segmentation to offer a framework that can handle diverse leaf images and which can effectively differentiate the type of the disease. The proposed model utilised a total of 55,400 images consisting of various plants' leaves spreading across 38 labels. The model LK-PSVM achieved an accuracy of 96.12%, precision of 91.15%, sensitivity of 96.43%, and specificity of 93.33%. Reddy and Adimoolam [7] developed machine learning models to determine the accuracy in leaf disease detection using the Naive Bayes (NB) classification algorithm and compared its accuracy features with the Support Vector Machine (SVM) algorithm for improving the accuracy of the prediction. The NB classification and the SVM algorithms were implemented and compared in terms of accuracy results. The SVM appears to be more significant with 95% accuracy than NB with 91%, with a significant value (0.081).

Raju *et al.* [8] developed a Support Vector Machine (SVM) based on the detection of various plant leaves using Image Processing Techniques. Plant leaf disease detection and identification include the stages like image acquisition, image pre-processing, image segmentation, feature extraction and classification. The leaf pictures are used for detecting plant diseases. Therefore, the use of image processing techniques to find and classify diseases in agricultural applications is useful. The model achieved 97.2% accuracy and 96.6% precision. Islam *et al.* [9] developed a method for classifying the early-stage stress symptoms of pepper seedlings and enabling their identification and quantification using image processing and a support vector machine (SVM). Two-week-old pepper seedlings were grown under different temperatures (20, 25, and 30 °C), light intensity levels (50, 250, and 450 μmol $m^{-2}s^{-1}$), and day–night hours (8/16, 10/14, and 16/8) in five controlled plant growth chambers. Eighteen colour features, nine texture features using the gray-level co-occurrence matrix (GLCM), and one morphological feature were extracted from each image. To reduce feature overlap, sequential feature selection (SFS) was applied, and a support vector machine (SVM) was used for stress classification. The SVM model, using these selected features, achieved a classification accuracy of 82.00%, precision of 91.00%, and sensitivity of 81.00%.

However, many of the classification models based on support vector machines performed better in detecting and classifying plant diseases, but some of the models can be sensitive to noise and outliers, especially when classes overlap considerably. Also, fine-tuning the hyperparameters of the SVM can be challenging [10], and SVM models are sensitive to the scaling of input features [11]. Therefore, many researchers have been integrating the optimisation techniques such as binary particle swarm optimisation (BPSO) and reptile search algorithm (RSA) to SVM models to enable them to be robust to noise, outliers, and overlapping classes, leading to improved performance and generalisation ability. Optimisation techniques such as BPSO and RSA can help SVM in scaling input features to ensure that all features contribute equally to the decision boundary by preventing features with larger values from dominating the model, leading to an improved model performance and stability. In addition, BPSO and RSA can make it possible to overcome the challenges associated with fine-tuning the SVM hyperparameters.

However, BPSO is prone to premature convergence, where the algorithm gets stuck in a local optimum before reaching the global optimum, due to the way particles move and interact [12]. The sigmoid transfer function in BPSO may not accurately capture the characteristics of discrete optimisation, leading to suboptimal results [13]. BPSO, like other optimisation algorithms, is sensitive to parameter settings such as inertia weight, acceleration coefficients, and population size [14]. Finding the optimal parameter settings can be challenging and time-consuming.

While RSA is a promising optimisation technique inspired by crocodile hunting, it struggles with some issues which limit their performance in optimisation problems. These issues arise from the algorithm's inherent complexity and the computational resources required for its execution. RSA's complexity stems from its simulation of crocodile behaviour, which involves multiple phases such as encircling, hunting, and prey capture. Each of these phases adds to the overall computational burden, especially when handling complex or high-dimensional optimisation problems. RSA, like other algorithms inspired by biological patterns, can have a high computational overhead due to the need to track and manage the population of "reptile searchers" (simulating crocodiles) and their interactions with the environment [15]. These limitations lead to slow convergence speed, making it difficult for the algorithm to reach the global optimum promptly. Additionally, the algorithm can get trapped in local optima, failing to find the best solution [16]. Therefore, this study develops an enhanced binary particle swarm optimisation (EBPSO) and an enhanced reptile search algorithm (ERSA) model to detect and classify three maize diseases (i.e., maize northern leaf blight, maize common rust, and maize gray leaf spot).

2. MATERIAL AND METHOD

In this study, an improved machine learning model based on optimised support vector machine (SVM) using enhanced Binary Particle Swarm Optimisation and Enhanced Reptile search Algorithm to optimise the support vector machine was formulated for diagnosing maize diseases. The IMLMBOSVM-EBPSO and IMLMBOSVM-ERSA approaches provide the capabilities of an optimisation technique, with a support vector machine (SVM), which is optimised by BPSO or the RSA optimisation algorithm for tuning the hyperparameters. Subsequently, the fine-tuned hyperparameters of the SVM were applied to the classification task, followed by an assessment of the model's performance using the test dataset. The dataset employed for this endeavor, sourced from Kaggle Village Datasets, encompasses 2,222 instances representing three distinct types of maize diseases and healthy maize The model design was structured around seven main phases: data acquisition, data preprocessing, segmentation, feature extraction via Gray Level Co-occurrence Matrix and Colour moment, classification models formulation (EBPSO-SVM & ERSA-SVM), disease classification employing the optimised SVM methodology, and final performance evaluation of the classification models. A visual representation of this multistage process is shown in Figure.1.

2.1 Dataset Acquisition and Preprocessing

The dataset comprises 2,222 images categorised into four distinct classes: 574 images of maize with northern leaf blight disease, 574 images of maize with common rust disease, 574 images of maize with gray leaf spot disease, and 500 images of healthy maize leaves. Figure 2 presents samples from the dataset used. A 10-fold cross-validation approach was employed to select the training and test datasets. Consequently, for the maize dataset, 223 datasets were used for testing, while the remaining 1,999 datasets were used for training, as shown in Table 1. As part of the image preprocessing phase, all images were appropriately scaled to dimensions of 128 × 128 pixels to align with the requirements of the model. RGB images were converted to grayscale using the luminosity method, indicated by Equation 2.1, which was put forth by Padmavathi and Thangadurai [17]. To improve the quality of the images, the bi-histogram equalisation technique was used for contrast enhancement and morphological filtering was further used to sharpen the image; this process enhanced the image. Lastly, adaptive median filtering was used to denoise the images before image segmentation processing.

$$Grayscale=0.2989 * Red + 0.587 * Green + 0.114 * Blue \quad (2.1)$$

2.2 Image Segmentation Procedure and Feature Extraction

The suggested models use the Sobel edge detection method to separate lesions from the uninfected part of a leaf. The process of using the Sobel edge detection method involved two stages. Firstly, the gradient of the image along the x-axis and y-axis is determined using the Sobel operator. After finding the image gradient, the next step is to find a threshold value automatically so that edges can be determined [18]. According to Aslam *et al*. [19], equations 2.2, 2.3, 2.4, and 2.5 were used to compute the image gradient along the x and y-axis and equations 2.6, 2.7 and 2.8 were used to calculate the threshold value. Algorithm 2.1 demonstrated an example of the image-dependent threshold computation and edge point Algorithm's pseudo-code.

$$g_x = \frac{\delta f}{\delta x} = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3) \quad (2.2)$$

$$g_y = \frac{\delta f}{\delta y} = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7) \quad (2.3)$$

Then the gradient of the image is defined as:

$$\nabla f(x,y) = \frac{\delta f}{\delta x}\hat{h} + \frac{\delta f}{\delta y} = g_x\hat{h} + g_y\hat{s} \quad (2.4)$$

where f(x,y) is the image, g(x,y) is the gradient image and $\hat{h}$ & $\hat{s}$ are unit vectors along the *x* and *y* axes, respectively. The magnitude of the gradient is given by equation 3.5:

$$g(x,y) = |\nabla f(x,y)| = \sqrt{gx^2 + gy^2} \quad (2.5)$$

Algorithm 2.1: Computation of the Threshold Value of the Image (Source: Adapted from Aslam *et al[19]*)

1: Input: a, height of the image, b, width of the image, and g(x,y): gradient of the image

2: Utilising Equation 3.6, compute the average intensity of the gradient image g(x,y). The initial threshold, Thd^0, is equal to the average intensity of the gradient image *g(x, y)*, as defined in equation 3.6

$$Thd^0 = \frac{\sum_{k=1}^{a} \sum_{l=1}^{b} g(x,y)}{a \times b} \quad (2.6)$$

3: Set iteration index *i* = 0, separate *g(x,y)* into two classes, where the lower class consists of those pixels of *g(x,y)* which have gradient values less than Thd^i, and the upper class contains the rest of the pixels.

4: Compute the average gradient values m_{Low} and m_{High} of the lower and upper classes, respectively.

5: Set iteration *i* = *i*+1 and update threshold value as:

$$Thd^i = \frac{m_{Low} + m_{High}}{2} \quad (2.7)$$

6: Repeat steps 2 to 4 until $|Thd^i - Thd^{i-1}| \leq \epsilon$, where $\epsilon \to 0$ and take Thd^i as the final threshold and denote it by *Thd.*

7: Compute the output utilising equation 3.8

$$E(x,y) = \begin{cases} 255 & g(x,y) \geq Thd, \\ 0 & otherwise \end{cases} \quad (2.8)$$

8: Comparing the value of the gradient image with the threshold value to determine the output:
If g*(x,y) > Thd*
Output: Return edge point (represented as white)
Endif
If *g(x,y) < Thd*
Output: Return background point (represented as black)
Endif

Gray Level Co-occurrence Matrix (GLCM) was used to extract only five texture features, such as energy, contrast, correlation, homogeneity, and entropy and six shape features, such as area, solidity, eccentricity, perimeter, rectangularity, and equidiameter. Colour moment was used to extract four colour features, such as medium, standard deviation, asymmetry, and kurtosis.

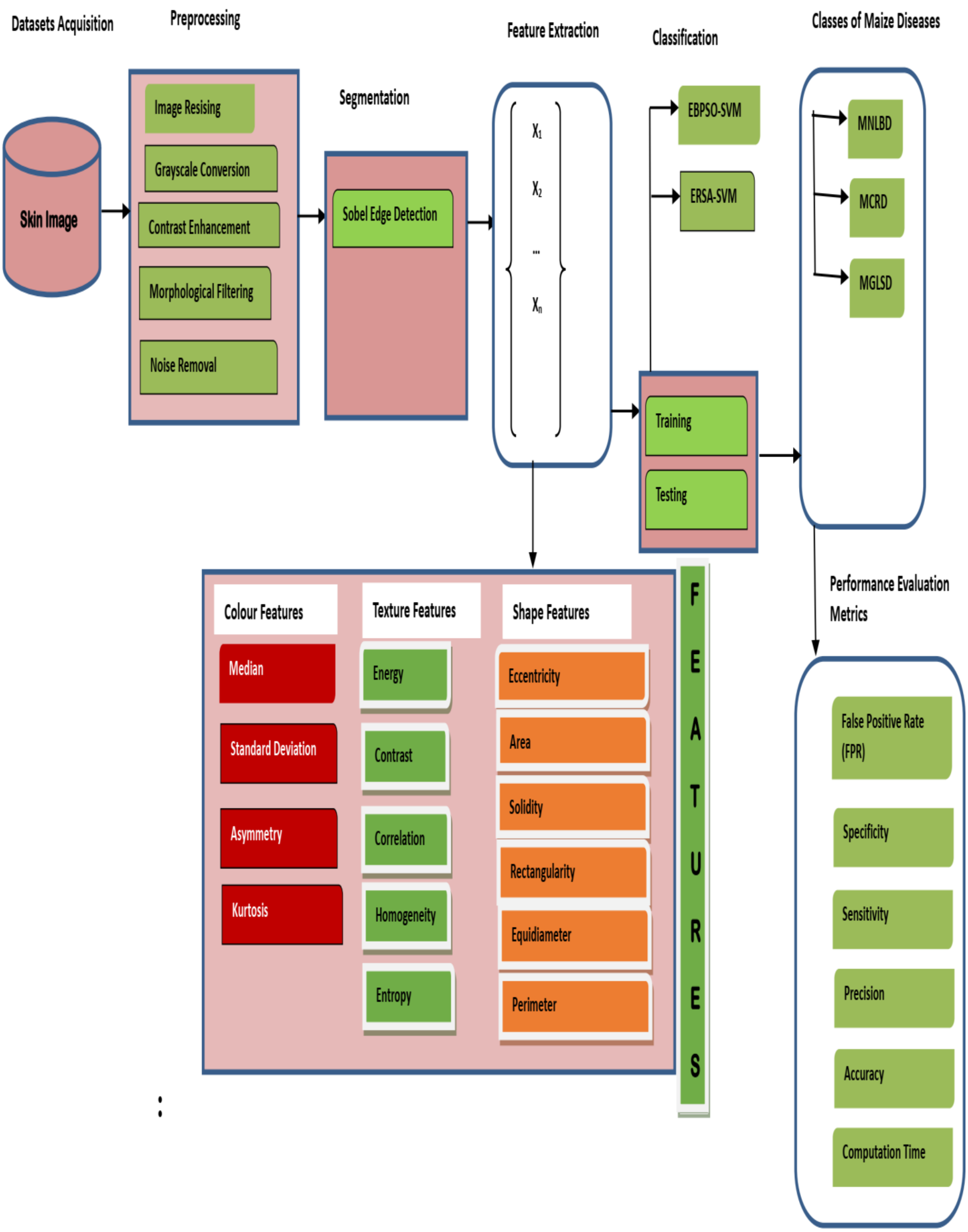

Fig. 1: A Block Diagram of the Developed Cassava Diseases Classification Models

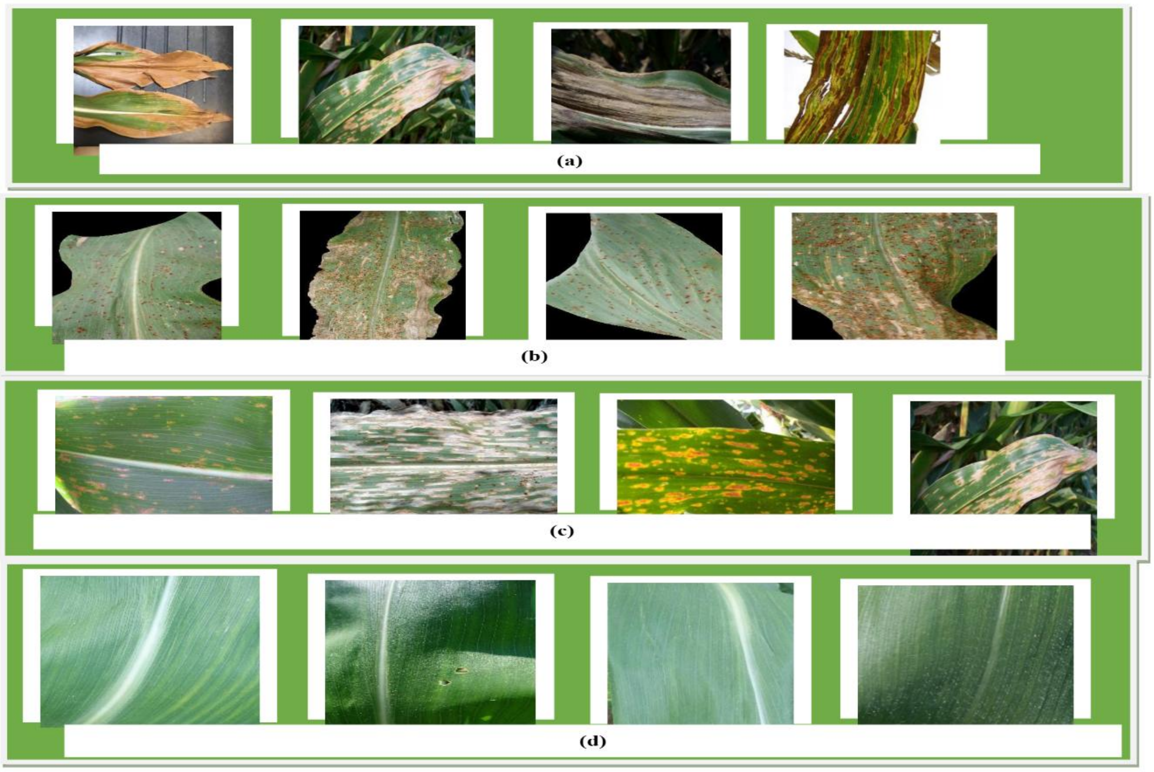

Fig. 2: Samples of the Maize Dataset used for the Study (a) Maize Northern Leaf Blight Disease (MNLBD), (b) Maize Common Rust Disease (MCRD), (c) Maize Cercospora Leaf Spot Disease (MCLSD)/Maize Gray Leaf Spot Disease (MGLSD), (d) Maize Healthy Leaf (Source: [18])

Table 1. Distribution of the Maize Dataset Acquired from the Kaggle Village

Class	Content	Number of Images per Class	Number of Training Datasets	Number of Testing Datasets
1	Maize Northern Leaf Blight Disease (MNLBD)	574	1,999	223
2	Maize Common Rust Disease (MCRD	574		
3	Maize Gray Leaf Spot Disease of Maize Cercospora Leaf Spot Disease (MGLSD/MCLSD)	574		
4	Healthy Maize Leaves	500		

2.3 Formulation of Enhanced Binary Particle Swarm Optimisation (EBPSO) Model and Enhanced Reptile Search Algorithm (ERSA) and Enhanced Reptile Search Algorithm (ERSA)

The algorithms 2.2 and 2.3 depict both the enhanced binary particle swarm optimisation model and the enhanced reptile search model.

Algorithm 2.2: Enhanced Binary Particle Swarm Optimisation (EBPSO) Model

Input: Population Size (N), maximum number of iterations F_{max}

1: Initialise the particles' position ($Z_{i,j}^{f}$), the velocity constant (Vel_c), previous best position ($h_{i,j}^{f}$), acceleration coefficient (C_1: factor of constant cognitive, C_2: factor of social scaling, number of particles (N), set *Iteration*=0, and the maximum number of iterations (F_{max})

Outputs: Solution to the problem

2: **Local Search**

According to Gou *et al.* [20], Chen *et al.* [21], and Lynn *et al.* [22], BPSO is reliable for handling challenging optimisation problems; however, its quick convergence can easily trap the swarm inside certain local optima. Its algorithm also lacks diversity in its search process [23-26]. Therefore, chaotic, Arc Tangent Acceleration Coefficients (AT), and Cosine Map Inertial were used to enhance the BPSO.

Applied Cosine Map Inertial

3: Evaluate inertial weight ω using Equation 2.9 as in Ma *et al.* [27]

$$\omega = \delta \times \cos\left(\left(\frac{F_j}{F_{max}}\right) \times \pi\right) + \vartheta \quad (2.9)$$

Where δ and ϑ are constants (δ=1/3, ϑ= 0.6), F_j and F_{max} denote the number of present iterations and the maximum number of iterations defined by the user, respectively.

4. Evaluate maximum velocity (v_{max}) using

$$Vel_{max} = \left(e^{1-f/F\max}\right) Vel_c$$

Applied Arc Tangent Acceleration Coefficients (AT)

5: Using Equations 2.10 and 2.11 to update the cognitive component C_1 and C_2 the social component as in Ma *et al.* [27].

$$C_1 = -\gamma \times arctan((\frac{F_j}{F_{max}}) \times \beta) + \rho_1 \quad (2.10)$$

$$C_2 = \gamma \times arctan((\frac{F_j}{F_{max}}) \times \beta) + \rho_2 \quad (2.11)$$

where γ, β, ρ_1, and ρ_2 are constant (γ = 1.5, β = 4, ρ_1 = 2.5, and ρ_2 = 0.5)

6: while (iteration < F*max*) do

7: For all particles (i) do

8: Using the current position $z_{i,j}^t$ of the i[th] as a starting point, calculate each particle's performance L (L ($z_{i,j}^f$)) using Equation 2.12

$$Z_{i,j}^{(f+1)} = Z_{i,j}^{f} + vel_{i,j}^{(f+1)} \quad (2.12)$$

9: Evaluate every single individual's performance with their greatest hitherto:
If L ($z_{i,j}^f$) < L ($h_{i,j}^f$) then
$h_{i,j}^f = z_{i,j}^f$
End if

10: Each particle's performance is compared to the universal best particle.
If L ($z_{i,j}^f$) < L ($h_{g,j}^f$) then
$h_{g,j}^f = z_{i,j}^f$
End if

11: Update the new velocity of particle i[th] ($vel_{i,j}^{(f+1)}$) using Equation 2.13

$$vel_{i,j}^{(f+1)} = \omega vel_{i,j}^{f} + C_1 \times r_1 \times (h_{i,j}^{f} - Z_{i,j}^{f}) + C_2 \times r_2 (h_{g,j}^{f} - Z_{i,j}^{f}) \quad (2.13)$$

where ω is the inertia weight; r_1 and r_2 are random numbers within [0,1]; C_1 and C_2 represent learning factors; $C_1 \times r_1 \times (h_{i,j}^{f} - Z_{i,j}^{f})$ is the individual cognition term, which represents the individual cognitive experience of the particles, and it causes the particles to move toward the best position they experience; and $C_2 \times r_2 (h_{g,j}^{f} - Z_{i,j}^{f})$ is the social cognition term, which denotes the influence of the group experience in the flight path of particles. The social cognition term moves the particles toward the best position found by the group, representing the sharing of information between the particles.

12: Update the new position of particle ith ($z_{i,j}^{(t+1)}$) using If rnd() < P($vel_{i,j}$), then $z_{i,j}$ = 1; else $z_{i,j}$ =0;

where

$$P(\upsilon) = \frac{1}{1+e^{-\upsilon el(i,j)}}$$

$$Z_{i,j}^{(f+1)} = Z_{i,j}^{f} + vel_{i,j}^{(f+1)}$$

End for

11: If ($vel_{(i,j)} > vel_{max}$)
$vel_{(i,j)} = vel_{max}$
Endif

12: If ($vel_{(i,j)} < -vel_{max}$)
$vel_{(i,j)} = -vel_{max}$
Endif

13: **Global Search**
Applied Chaotic

14: Chaotic search K times near gbest

15: Use Equation 2.12 to generate a chaotic random sequence D between [0, 1] using the logistic equation, and then pass the carrier map in Equation 2.14, introducing chaos into gbest**,** a nearby area, to achieve local chaotic search as in Ma *et al.* [27]

:

$$D \rightarrow A: Z = gbest + M \times \cos(D) \quad (2.14)$$

16: *E* chaotic search points near $pbest_f$ are obtained

17: for *j=1: E* do

18: if *f* (*Z*) < *f* (*gbest*)

19: Set *f* (*Z*) to be *gbest*

20: End if

21: End for

22: Increase the number of iterations. *Iteration = Iteration +1*

23: End while

24: Return gbest

Algorithm 2.3: Enhanced Reptile Search Algorithm (ERSA) Model

---

Input: Number of Features (N), maximum number of iterations *T*

1. Initialise the RSA parameters, number of features (N), minimum and maximum feature boundaries (LWB, UPB), a sentient specification that governs the global optimum performance (δ), two parameter factors that contribute to the quality of the classification model (θ and μ), the number of population (Po), set iteration t=0, and the highest number of repetitions (T)

Outputs: Solution to the problem

2. Generate an initial population randomly using Equation 2.15

$$Z_{i,j} = random \times (UPB_j - LWB_j) + LWB_j, j = 1,2, \ldots, M \quad (2.15)$$

where random denotes a random number between 0 and 1, UPB_j $and - LWB_j$ denote the upper bound and lower bound of the search space. M denotes the dimension of the optimisation problem. And $Z_{i,j}$ means the generated position of the i-th individual in the j-th dimension.

3: while t< T) do

4: For all populations (Po) do

5: Update RSA parameters using Equations 2.16, 2.17, 2.18, 2.19, and 2.20

$$\Psi_{(i,j)}(t) = B_j(t) \times P_{(i,j)}(t) \quad (2.16)$$

$$R_{(i,j)}(t) = \frac{B_j(t) - Z_{(r1,j)}}{B_j(t) + \epsilon} \quad (2.17)$$

According to Khan *et al.* [28], RSA has certain drawbacks, including highly complex computations, sluggish convergence, and neighbourhood minimum trapping. Also, the parameter ES(t) has a limited range of variation, which makes the RSA show poor global search ability and converge slowly [29]. To enhance the searching capabilities of RSA, an enhanced RSA is developed by using multiple strategies: dynamic evolutionary sense factor, prey approaching strategy and Cauchy mutation strategy.

Dynamic Evolutionary Sense (DES)

$$DES(t) = 2 \times r_2 \times \left(1 - \frac{t}{T}\right)^{(2\times\frac{t}{T})} \times randomn \quad (2.18)$$

where, $B_j(t)$ is the current best position within the population. t is the current iteration number, and T is the maximum iteration number; randomn means a random number satisfying the standard normal distribution. $\Psi_{(i,j)}(t)$ is the hunting operator, which is determined using Equation 2.16, $R_{(i,j)}(t)$ is a reduce function, which is calculated using Equation 2.17, r1 is a random integer between [1, N]. DES(t) is a factor used to improve the ES(t) to avoid early convergence, and it can be obtained by using Equation 2.18. $\in$ is a very small positive number. r2 can be 1 or 1 randomly. $P_{(i,j)}(t)$ is the percentage difference between the current position and the best position, which is calculated using Equations 2.19 and 2.20.

$$P_{(i,j)}(t) = \delta + \frac{Z_{(i,j)} - N(z_1)}{B_j(t) \times (UPB_j - LWB_j) + \in} \quad (2.19)$$

$$N(z_1) = \frac{1}{M}\sum_{j=1}^{M} Z_{(i,j)} \quad (2.20)$$

where represents a sensitive parameter controlling exploration performance, which is set to 0.1 in RSA as mentioned by Zhou *et al.* [29], and $N(z_1)$ signifies the average solution. To balance the exploration and exploitation performance of RSA, DES(t) is applied in the stages of high walking and hunting coordination.

6. **Phase 1: Encircling phase (Exploration Mechanisms, i.e., Global Search)**
   Perform high walking exploration mechanism using Equation 2.21

$$Z_{(i,j)}(t+1) = B_j(t) - (\Psi_{(i,j)}(t) \times \alpha + R_{(i,j)}(t) \times random) \times DES(t), if\ t \leq T \times 0.5 \quad (2.21)$$

   where, α is a constant value, indicating the exploration accuracy, which is set to 0.005 in RSA as mentioned by Zhou *et al*. [29] and r3 is a random integer between [1, N].

7. **Phase 2: Prey Approaching Strategy (PAS)**
   A prey approaching strategy replaced the belly walking of RSA (PAS), which is based on the secretary bird optimisation algorithm (SBOA) proposed by Fu *et al.* [30]. The SBOA helps the RSA explore the search space more effectively. Equation 2.22 was used to mathematically express the SBOA model as in Zhou *et al*. [29]:

$$Z_{(i,j)}(t+1) = B_j(t) + \exp\left(\left(\frac{t}{T}\right)^4\right) \times (randomn - 0.5) \times (B_j(t) - Z_{(i,j)}(t)) \quad (2.22)$$

8. **Phase 3: Hunting phase (Exploitation Mechanisms, i.e., Local Search)**
   To balance the exploration and exploitation performance of RSA, DES(t) is applied in the stages of hunting coordination using Equation 2.23

$$Z_{(i,j)}(t+1) = B_j(t) - (\Psi_{(i,j)}(t) \times \in + R_{(i,j)}(t) \times random) \times DES(t), if\ t > 3 \times T \times 0.25 \quad (2.23)$$

$$Z_{(i,j)}(t+1) = B_j(t) \times P_{(i,j)} \times random, if\ t \leq 3 \times T \times 0.25\ and\ t > T \times 0.5 \quad (2.24)$$

9. **Phase 4: Cauchy Mutation Strategy (CMS)**
   According to Zhao *et al.* [31], Cauchy Mutation Strategy (CMS) provide more potential candidates and enrich the population diversity of the metaheuristic algorithm. Therefore, the CMS was executed after the hunting phase of RSA. The new position after the mutation can be calculated using Equations 2.25 and 2.26

$$Z_{(i,j)} = Z_{(i,j)} \times (1 + CauchyRandom) \quad (2.25)$$

$$CauchyRandom = a \times tan(\pi \times (random - 0.5)) + b \quad (2.26)$$

   where the parameters a and b are set to 1 and 0 in ERSA as in Zhou *et al.* [29], respectively. Cauchy Random is a random number based on the standard Cauchy distribution.

8. End for
9. Increase the iteration t=t+1
10 If t>T, move to step 11; otherwise, step 4
11. End while
12. Return the best position

3. RESULTS AND DISCUSSION OF THE FINDINGS

This experimental exploration assessed the efficacy of the proposed approaches through five performance metrics (i.e., false positive rate, specificity, sensitivity, precision and accuracy). The principal aim of this investigation was to appraise the performance gained from the enhancement of the optimisation techniques with a support vector machine classifier across diverse disease datasets. All experiments were conducted by utilising the Machine Learning (ML) Toolbox programming environment in MATLAB 2020a and executed on a laptop equipped with Intel(R) Core (TM) i3-1005G1 CPU, an X64 version with 8GB RAM, and a 256 SSD HDD, and running Windows 10 Home.

3.1 Performance Evaluation Metrics of the Classification Models on the Maize Datasets

The results presented in Table 2 indicate that All maize diseased datasets exhibit a greater level of misclassification compared with other maize datasets. Specifically, EBPSO-SVM and ERSA-SVM incorrectly classified 38 and 44 diseased maize leaf samples, respectively, as healthy. In addition, 35 samples using EBPSO-SVM and 41 samples using

ERSA-SVM were wrongly identified as diseased instead of healthy. Further analysis across the three disease categories (MNLBD, MCRD, and MGLSD) reveals that the MGLSD dataset recorded the highest misclassification rate. Moreover, a comparative evaluation of the two models shows that ERSA-SVM consistently produced more misclassification errors than the EBPSO-SVM model.

Table 2. Performance Evaluation Metrics of the Classification Models on the Maize Datasets

	All Maize Diseased Dataset	Maize Northern Leaf Blight Disease (MNLBD)	Maize Common Rust Disease (MCRD)	Maize Gray Leaf Spot Disease (MGLSD)
TP				
EBPSO-SVM	1684	559	560	557
ERSA-SVM	1678	553	554	551
FN				
EBPSO-SVM	38	15	14	17
ERSA-SVM	44	21	20	22
FP				
EBPSO-SVM	35	12	11	14
ERSA-SVM	41	18	17	19
TN				
EBPSO-SVM	465	488	489	486
ERSA-SVM	459	482	483	481

Note:
EBPSO-SVM (Enhanced Binary Particle Swarm Optimisation-Support Vector Machine), ERSA-SVM (Enhanced Reptile Search Algorithm-Support Vector Machine), TP (True Positive), FN (False Negative), FP (False Positive), TN (True Negative)

3.2 Performance Evaluation Metrics of the IMLMBOSSVM-EBPSO and IMLMBOSVM-ERSA on Maize Datasets

The EBPSO-SVM model consistently demonstrated superior performance over the ERSA-SVM model across all disease classes. Specifically, EBPSO-SVM achieved lower false positive rates of 2.44%, 2.20%, and 2.80% compared with 3.60%, 3.40%, and 3.80% recorded by ERSA-SVM for MNLBD, MCRD, and MGLSD, respectively. Correspondingly, higher specificity values were obtained by EBPSO-SVM (97.60%, 97.80%, and 97.20%) relative to ERSA-SVM (96.40%, 96.60%, and 96.20%) across the same disease categories (Table 3). In terms of sensitivity, EBPSO-SVM also outperformed ERSA-SVM, achieving 97.39%, 97.56%, and 97.03% compared with 96.34%, 96.52%, and 96.16% for MNLBD, MCRD, and MGLSD, respectively (Table 3). Moreover, EBPSO-SVM yielded higher precision scores of 97.90%, 98.07%, and 97.55%, surpassing the corresponding ERSA-SVM values of 96.85%, 97.02%, and 96.67% (Table 3). Finally, EBPSO-SVM attained overall accuracy rates of 97.49%, 97.67%, and 97.11%, which exceeded the ERSA-SVM accuracies of 96.37%, 96.55%, and 96.17% for the three disease classes, respectively (Table 3).

Furthermore, the results presented in Tables 5 and 7 demonstrate that both the EBPSO-SVM and ERSA-SVM classification models achieve statistical significance, as the p-values associated with all performance evaluation metrics are below the 0.05 threshold.

Table 3. Performance Evaluation Metrics of the IMLMBOSVM-EBPSO and IMLMBOSVM-ERSA on Maize Datasets

	MNLBD	MCRD	MGLSD	Average
FPR (%)				
EBPSO-SVM	2.44	2.20	2.80	2.48
ERSA-SVM	3.60	3.40	3.80	3.60
Specificity (%)				
EBPSO-SVM	97.60	97.80	97.20	97.53
ERSA-SVM	96.40	96.60	96.20	96.40
Sensitivity (%)				
EBPSO-SVM	97.39	97.56	97.03	97.33
ERSA-SVM	96.34	96.52	96.16	96.34
Precision (%)				
EBPSO-SVM	97.90	98.07	97.55	97.84
ERSA-SVM	96.85	97.02	96.67	96.85
Accuracy (%)				
EBPSO-SVM	97.49	97.67	97.11	97.41
ERSA-SVM	96.37	96.55	96.17	96.36

Note:
EBPSOSVM (Enhanced Binary Particle Swarm Optimisation-Support Vector Machine), ERSA-SVM (Enhanced Reptile Search Algorithm-Support Vector Machine), FPR (False Positive Rate), MNLBD (Maize Northern Leaf Blight Disease), MCRD (Maize Common Rust Disease), MGLSD (Maize Gray Leaf Spot Disease)

Table 4. One-Sample Statistics (EBPSO-SVM Model)

Performance Metrics	N	Mean	Std. Deviation	Std. Error Mean
False Positive Rate	16	4.2000	2.13167	.53292
Specificity	16	95.8000	2.13167	.53292
Sensitivity	16	97.6045	.28725	.07181
Precision	16	97.4516	.53669	.13417
Accuracy	16	97.0931	.39132	.09783

Table 5. One-Sample T-test (EBPSO-SVM Model)

Performance Metrics	t	df	Sig. (2-tailed)	Mean Difference	95% Confidence Interval of the Difference	
					Lower	Upper
False Positive Rate	7.881	15	.000	4.20000	3.0641	5.3359
Specificity	179.765	15	.000	95.80000	94.6641	96.9359
Sensitivity	1359.176	15	.000	97.60453	97.4515	97.7576
Precision	726.309	15	.000	97.45160	97.1656	97.7376
Accuracy	992.467	15	.000	97.09310	96.8846	97.3016

Table 6. One-Sample Statistics (ERSA-SVM Model)

Performance Metrics	N	Mean	Std. Deviation	Std. Error Mean
False Positive Rate	16	5.4000	2.12540	.53135
Specificity	16	94.6000	2.12540	.53135
Sensitivity	16	96.7984	.47987	.11997
Precision	16	96.5904	.63782	.15946
Accuracy	16	96.1578	.18802	.04701

Table 7. One-Sample T-test (ERSA-SVM Model)

Performance Metrics	t	df	Sig. (2-tailed)	Mean Difference	95% Confidence Interval of the Difference	
					Lower	Upper
False Positive Rate	10.163	15	.000	5.40000	4.2675	6.5325
Specificity	178.037	15	.000	94.60000	93.4675	95.7325
Sensitivity	806.863	15	.000	96.79836	96.5427	97.0541
Precision	605.749	15	.000	96.59041	96.2505	96.9303
Accuracy	2045.688	15	.000	96.15780	96.0576	96.2580

3.3 Graphical Representation of Performance Evaluation Metrics of the EBPSO-SVM and ERSA-SVM Classification Models on Maize Datasets

The graphical analysis of the evaluation metrics indicates that EBPSO-SVM consistently outperforms ERSA-SVM across all maize disease categories, MNLBD, MCRD, and MGLSD. As illustrated in Figures 3 through 7, EBPSO-SVM achieves superior results for every performance metric considered on the diseased dataset.

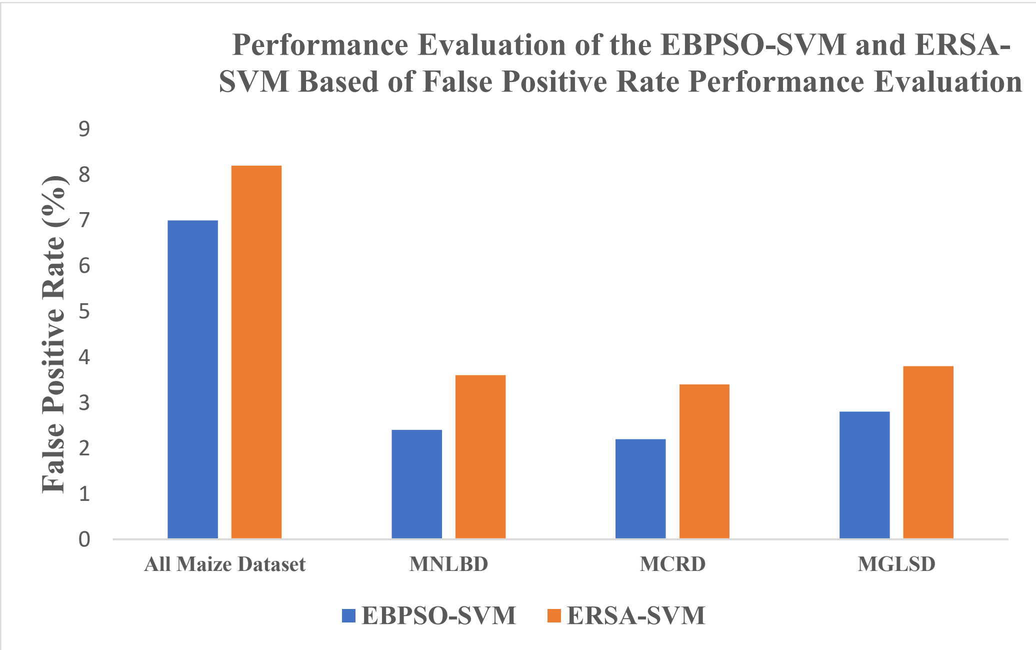

Fig. 3: False Positive Rate of EBPSO-SVM and ERSA-SVM Models on Maize Disease Dataset

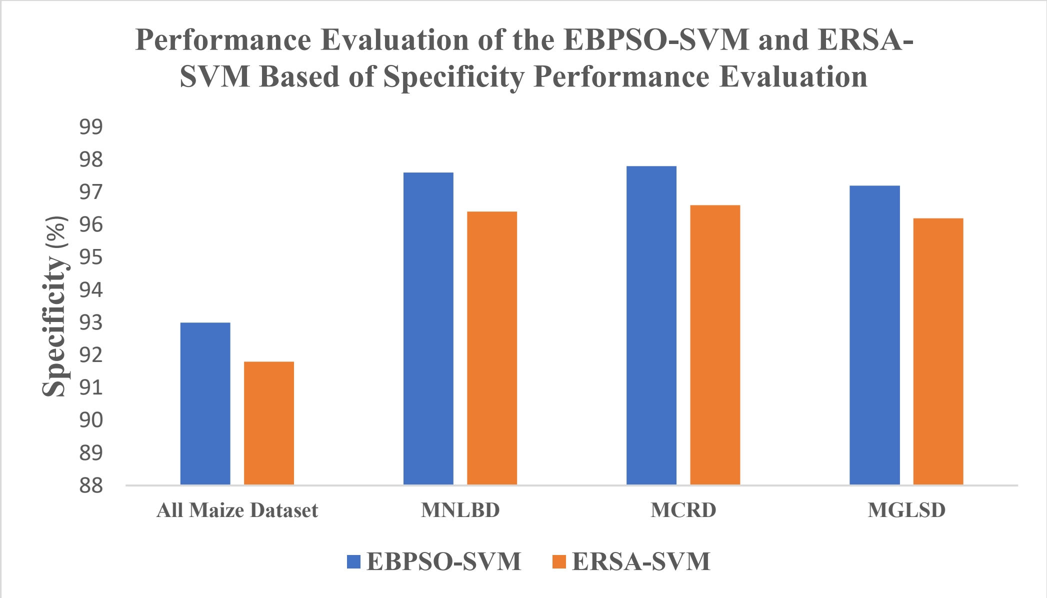

Fig. 4: Specificity of EBPSO-SVM and ERSA-SVM Models on Maize Disease Datasets

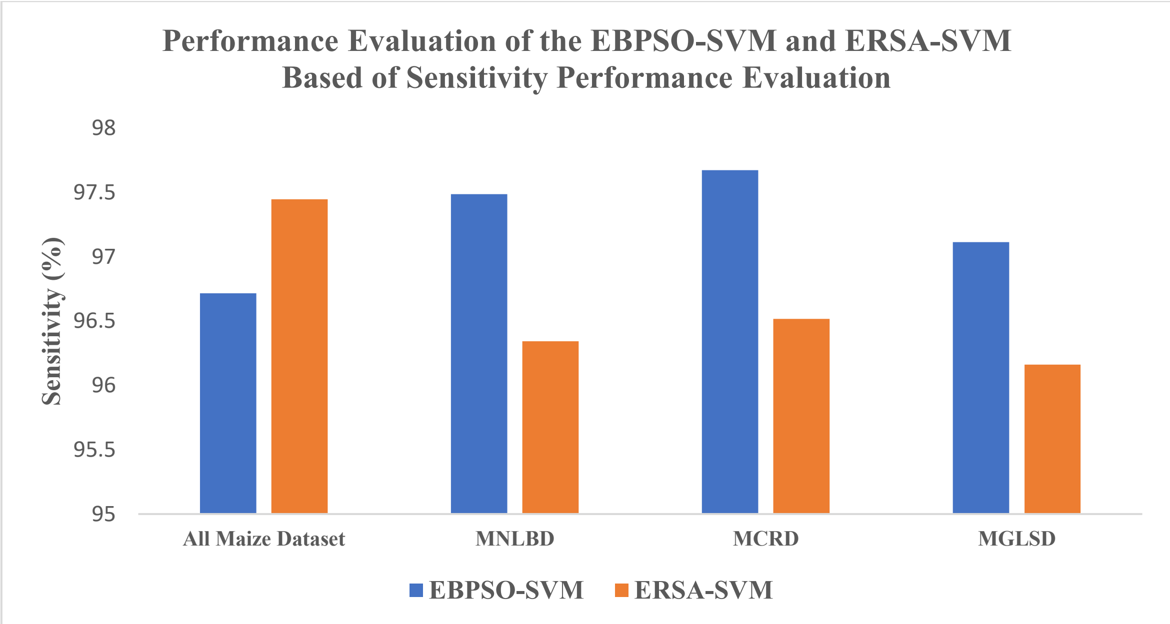

Fig. 5: Sensitivity of EBPSO-SVM and ERSA-SVM Models on Maize Disease Datasets

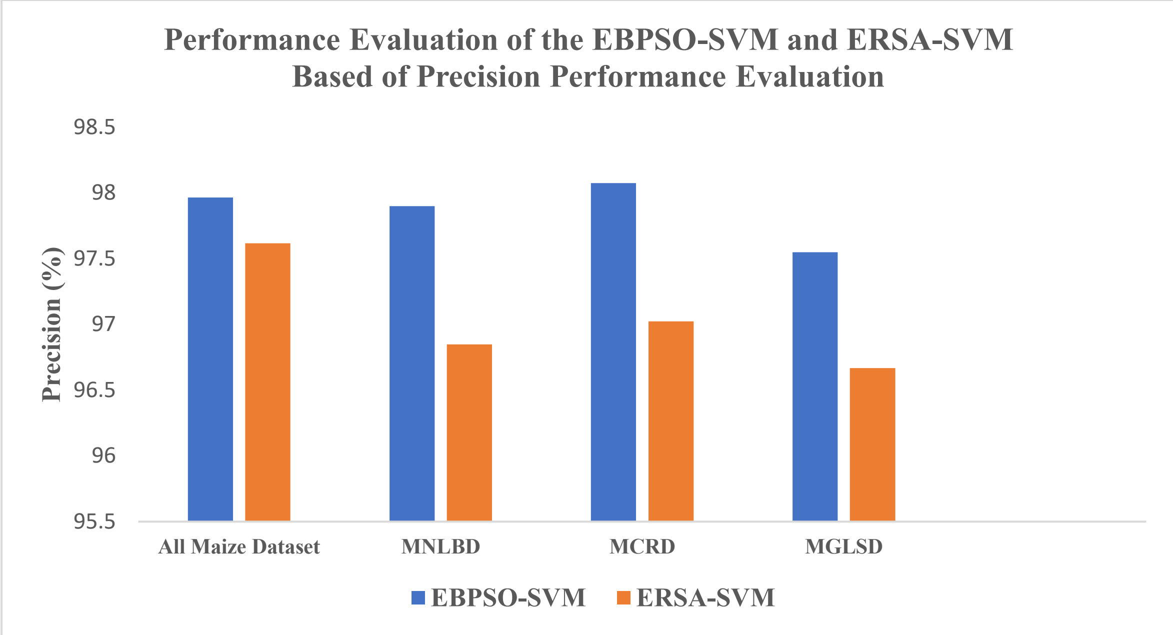

Fig. 6: Precision of EBPSO-SVM and ERSA-SVM Models on Maize Disease Datasets

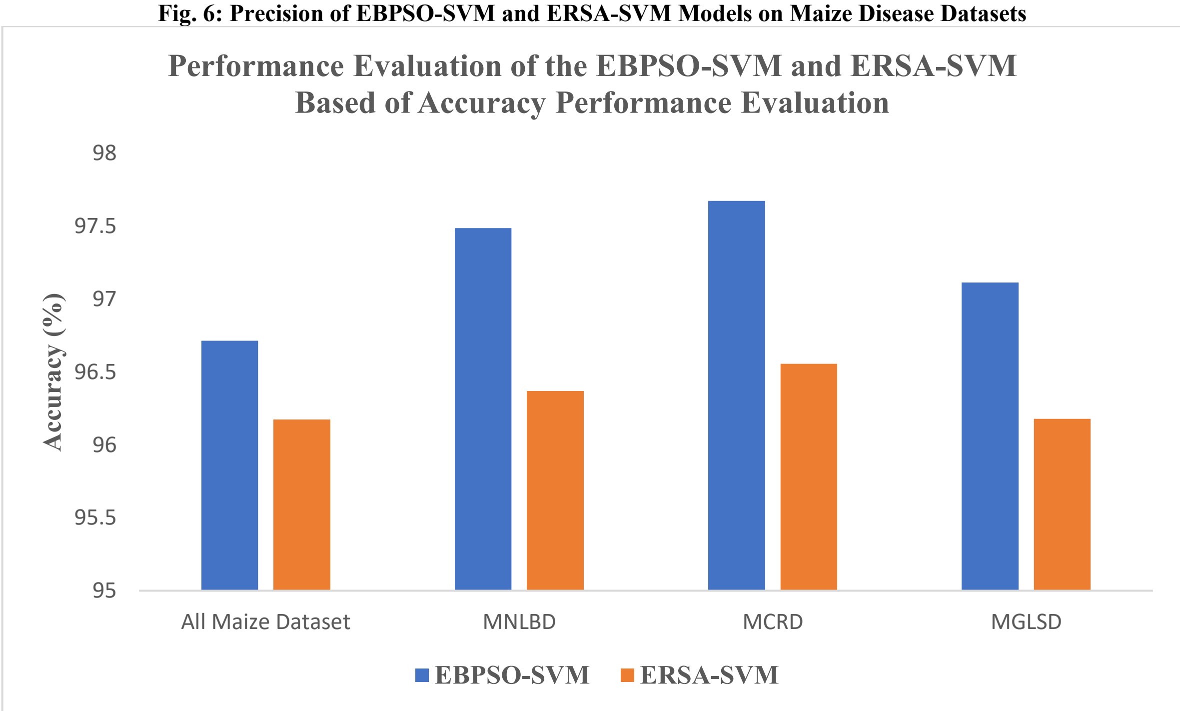

Fig. 7: Accuracy of EBPSO-SVM and ERSA-SVM Models on Maize Disease Datasets

3.4 Performance Evaluation Metrics Comparison of IMLMBOSVM-EBPSO and IMLMBOSVM-ERSA with the Existing Machine Learning Models in Plant Diseases Classification

Table 3 shows that the improved machine learning models based on optimised support vector machine developed (IMLMBOSVM-EBPSO and IMLMBOSVM-EBPSO) outperform some of the existing state-of-the-art machine learning models and deep learning models. For instance, the developed models outperform (BPSO-SVM and RSA-SVM), SVM, LK-SVM, and IMLDD-MRFODL machine learning models developed by Ayoade [18], Islam *et al.* [9], Rajagopal *et al.* [6], and Vimalkumar and Latha [2], respectively, in terms of performance evaluation metrics used in their studies. However, only the IMLMBOSVM-EBPSO proposed model outperforms the random forest machine learning model developed by Ibrahim *et al.* [32] in terms of specificity and accuracy. Similarly, the developed models outperform the IMACDCPSO-ILOAN (VGG16, ResNet34, and ResNet50), and PRF-SVM deep learning models developed by Yang *et al.* [33], Theerthagiri *et al.* [34], and Bachhal *et al.* [35], respectively, in terms of the performance evaluation metrics used in their studies. Finally, the SVM and BPNN machine learning models developed by Ibrahim *et al.* [32] outperform the proposed models in terms of the performance evaluation metrics used in their studies, except for the specificity of the BPNN. Also, the BPSO-RSA-SVM hybrid machine learning model developed by Ayoade [18] outperforms the developed model in terms of all the performance evaluation metrics used in this study. Similarly, the ResNet50v2 and Inceptionv3 CNN, VGG16, and CNN-ViT deep learning models developed by Kauri and Bansal [36], Ashurov *et al.* [37], Rani *et al.* [38], and Aboelenin *et al.* [39], respectively, in terms of the performance evaluation metrics used in their study.

3.5 Discussion of the Findings

It was observed that both the IMLMBOSVM-EBPSO and IMLMBOSVM-ERSA perform better than both the Binary Particle Swarm Optimisation-Support Vector Machine (BPSO-SVM) and Reptile Search Algorithm -Support Vector Machine (RSA-SVM) machine learning models developed by Ayoade [18] in terms of all the performance evaluation metrics used in this study, as shown in Table 3. This is possible because of the improved exploration and exploitation of the search space during hyperparameter tuning of the support vector machine (SVM). The deployment of improved techniques, like Chaotic Map, Arc Tangent Acceleration Coefficient, and Cosine Map Inertia Weight to binary particle swarm optimisation (BPSO) and Dynamic Evolutionary Sense, Prey Approaching Strategy, and Cauchy Mutation Strategy to reptile search algorithm (RSA), can better navigate the complex landscape of SVM parameters, leading to more accurate predictions, better generalisation, and increased robustness. On the contrary, the support vector machine learning model developed by Ibrahim *et al.* [32] performs better than the improved machine learning models based on optimised support vector machine (IMLMBOSVM-EBPSO and IMLMBOSVM-ERSA). Unlike the support vector machine model developed by Ibrahim *et al.* [32], which was not optimised with any optimisation techniques, the proposed models were optimised with an improved optimisation technique. The better performance of Ibrahim et al.'s [32] models could be a result of a combination of factors, including the choice of SVM parameters, the characteristics of the data, the complexity of the model, and the specific problem being addressed.

Furthermore, according to Alzubaidi *et al.* [41], "Deep Learning models have been demonstrated to outperform popular machine learning techniques in several fields, including cybersecurity, natural language processing, bioinformatics, robotics and control, and medical information processing. This assertion is supported by the performance of (ResNet50v2 and Inceptionv3), CNN, VGG16, and CNN-ViT

Table 3. Performance Evaluation Metrics of the IMLMBOSVM-EBPSO and IMLMBOSVM-ERSA with the Existing Machine Learning Models in Plant Diseases Classification

Author(s) and Models	FALSE POSITIVE RATE (%)	SPECIFICITY (%)	SENSITIVITY (%)	PRECISION (%)	ACCURACY (%)
		Ibrahim *et al.* [32]			
"Random Forest"	-	95.27	97.15	-	96.49
"Support Vector Machine (SVM)"	-	98.48	99.74	-	98.74
"BPNN"	-	97.16	98.47	-	97.82
		Nan *et al.* [40]			
"MobileNetV2"	-	-	97.82	97.82	97.53
"EfficientNetB3	-	-	98.14	97.47	97.69
"ResNet50"	-	-	98.22	98.24	98.02
		Khan *et al.* [28]			
"YOLOv3-tiny	-	-	43.00	81.00	69.40
"YOLOv4"	-	-	98.00	89.00	97.50
"YOLOv5s"	-	-	89.00	95.00	88.23
"YOLOv7s"	-	-	95.00	100.00	93.30
"YOLOv8n"	-	-	87.86	88.00	99.04
		Islam *et al.* [9]			
"SVM"	-	-	81.00	91.00	82.00
		Rajagopal *et al.* [6]			
"LK-PSVM"	-	93.33	96.43	91.15	96.12
		Vimalkumar and Latha [2]			
"IMLDD-MRFODL"	-	-	96.27	96.27	98.14
		Kauri and Bansal [36]			
"MobileNetv2"	-	-	97.00	97.00	99.42
"ResNet50v2"	-	-	99.00	99.00	99.42
"Inceptionv3"	-	-	99.00	98.00	98.49
		Yang *et al.* [33]			
"IMACDCPSO-ILOAN (ICPNet)"	-	-	87.35	87.99	88.46
		Theerthagiri *et al.* [34]			
"SqueezeNet"	-	-	96.50	97.00	97.00
"VGG16"	-	-	90.50	89.25	92.00
"ResNet34"	-	-	94.75	93.75	95.00
"ResNet50"	-	-	92.75	93.00	94.00
		Bachhal *et al.* [35]			
"PRF-SVM"	-	-	75.97	84.45	94.83
		Ashurov *et al.* [37]			
"CNN"	-	-	99.00	97.00	98.00
		Rani *et al.* [38]			
"VGG16"	-	-	98.43	98.27	98.25
		Aboelenin *et al.* [39]			
"CNN-ViT"	-	-	98.00	98.00	98.00

		Ayoade [18]			
BPSO-SVM	3.27	96.73	96.63	97.13	96.68
RSA-SVM	4.60	95.40	95.47	95.97	95.44
BPSO-RSA-SVM	2.80	97.20	97.04	97.55	97.11
		Developed Model			
IMLMBOSVM-EBPSO	**2.48**	**97.53**	**97.33**	**97.84**	**97.41**
IMLMBOSVM-ERSA	**3.60**	**96.40**	**96.34**	**96.85**	**96.36**

Note

LK-PSVM (Legion Kernels-Parallel Support Vector Machine), IMLDD-MRFODL (Intelligent Maize Leaf Disease Detection -Manta-Ray Foraging Optimisation with a Deep Learning), IMACDCPSO-ILOAN (Integrated Multidimensional Attention Coordinate Depthwise Convolutional Particle Swarm Optimisation-Integrated Lion Optimisation Algorithm Network), PRF-SVM (PSPNet ResNet50 Fuzzy Support Vector Machine), BPNN (Back Propagation Neural Network), CNN (Convolutional Neural Network), CNN-ViT (Convolutional Neural Network-Vision Transformer)

deep learning models developed by Kauri and Bansal [36], Ashurov *et al.* [37], Rani *et al.* [38], and Aboelenin *et al.* [39], respectively. This finding demonstrates that deep learning models offer a more data-driven and automated approach to feature extraction and model building, making them have the ability to automatically learn complex hierarchical features from data, while SVMs rely on hand-crafted or pre-defined features. However, the IMLMBOSVM-EBPSO and IMLMBOSVM-ERSA models perform better than some deep learning models, such as IMLDD-MRFODL, IMACDCPSO-ILOAN, and (VGG16, ResNet34, and ResNet50), developed by Vimalkumar and Latha [2], Yang *et al.* [33], and Theerrthagiri *et al.* [34], respectively. These could be factors like superior generalisation in high-dimensional spaces, better handling of limited data, and less susceptibility to overfitting. SVMs excel in areas where deep learning's reliance on large datasets and computational resources is a disadvantage. While deep learning models have made significant advancements, SVMs, when optimised with enhanced techniques, can still provide superior performance in specific contexts, particularly when dealing with limited data, high-dimensional spaces, or computational constraints. However, the choice between SVM and deep learning depends on the specific application and its requirements.

Finally, the hybridised machine learning model (BPSO-RSA-SVM) developed by Ayoade [18] outperforms the developed models despite improving the BPSO with Chaotic Map, Arc Tangent Acceleration Coefficient, and Cosine Map Inertia Weight and RSA with Dynamic Evolutionary Sense, Prey Approaching Strategy, and Cauchy Mutation Strategy. Individual or improved algorithms can increase the diversity of initial solutions and potentially improve exploration in algorithms such as RSA, but they do not provide the same level of exploitation as RSA and BPSO combined. Hybridisation of two or more optimisation techniques allows for a more comprehensive search of the parameter space, resulting in a balance of exploitation and exploration search behaviours, as well as a potentially more robust solution for scaling SVM input features, resolving SVM challenges of hyperparameter tuning, datasets with noise, outliers, and overlapping classes than using a single algorithm with or without these strategies. Hybridised machine learning models can leverage diverse approaches to tackle complex problems more effectively, leading to enhanced performance, increased accuracy, and robust predictions.

4. CONCLUSION AND FUTURE WORK

This study investigated the development of an improved machine learning model based on an optimised support vector machine for maize disease detection and classification, aimed at advancing sustainable agricultural practices. By enhancing binary particle swarm optimisation (BPSO) and reptile search algorithm (RSA) with different strategies to alleviate issues affecting their performances in optimisation problems and integrating each of the enhanced optimisation techniques with Support Vector Machine (SVM) and leveraging the Kaggle Village datasets, this study has demonstrated highly effective approaches for diagnosing and classifying maize leaf diseases. The gray level co-occurrence matrix and colour moment were utilised to extract essential image features from maize plant images, while the SVM, fine-tuned using enhanced binary particle swarm optimisation (EBPSO) and enhanced reptile search algorithm (ERSA), made precise classification decisions based on these features. The IMLMBOSVM-EBPSO model achieved an impressive accuracy rate of 97.41%, a precision rate of 97.84%, a sensitivity rate of 97.33%, a specificity rate of 97.53%, and a false positive rate of 2.48%. Also, the IMLMBOSVM-ERSA model achieved an impressive accuracy rate of 96.36%, a precision rate of 96.85%, a sensitivity rate of 96.343%, a specificity rate of 96.40%, and a false positive rate of 3.60%. These results showcase the significant potential of the proposed models of their capability in detecting and categorising the maize diseases. This technology stands to benefit farmers and agricultural experts immensely, enhancing crop productivity and reducing the need for labour-intensive manual inspections. Looking ahead, our plans involve expanding the application of our model to encompass additional categories of maize diseases. Simultaneously, we aim to explore optimised deep learning classifiers, enhancing the accuracy of early disease detection in maize plants. Importantly, our proposed model has surpassed existing state-of-the-art methods, underlining its effectiveness in the field of agricultural disease classification.

5. ACKNOWLEDGMENTS

Our thanks to the reviewers and the editorial team for all their efforts put into making this paper a reality.

6. REFERENCES

[1] Ashwini, C., Sellam, V. 2024. An optimal model for identification and classification of corn leaf disease using hybrid 3D-CNN and LSTM. Biomedical Signal Processing and Control, 9, 1-18.

[2] Vimalkumar, S., Latha, R. 2024. Maize leaf disease detection using manta-ray foraging optimization with deep learning model. Engineering, Technology & Applied Science *Research*, 14(5), 17068-17074. https://etasr.com/index.php/ETASR/article/view/7821/4089

[3] Haque, A., Marwaha, S., Deb, C. K., Nigam, S., Arora, A. 2023. Recognition of diseases of maize crop using deep learning models. Neural Computing and Applications, 35(10), 7407–7421.

[4] Rai, C.K., Pahuja, R. 2024. Northern maize leaf blight disease detection and segmentation using deep convolution neural networks. Multimedia Tools and Applications, 83(7), 19415-19432.

[5] Qian, X., Zhang, C., Chen, L., Li, K. 2022. Deep learning-based identification of maize leaf diseases is improved by an attention mechanism: Self-attention. Frontiers in plant science, 13(864486), 1-15. https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2022.864486/pdf

[6] Rajagopal, M., Kayikci, S., Abbas, M., Sivasakthivel, R. 2024. A novel technique for leaf disease classification using legion kernels with parallel support vector machine (LK-PSVM) and fuzzy C means image segmentation. Heliyon, 10(32707), 1-17. https://pmc.ncbi.nlm.nih.gov/articles/PMC11237940/pdf/main.pdf

[7] Reddy, Y.A., Adimoolam, M. 2024. Efficient plant leaf disease detection using support vector machine algorithm and compare its features with Naive Bayes classification. AIP Conference Proceedings, 2729(1), 1-15.

[8] Raju, T., Supriya, V., Swathi, P., Shamshuddin, S.H., Mohan, V. 2024. Support vector machine based diseases detection of various plant leaf using image processing techniques. The International Journal of Analytical and Experimental Modal Analysis, 15(3), 1366-1372.

[9] Islam, S., Samsuzzaman, Reza, N., Lee, K., Ahmed, S., Cho, Y.J., Noh, D.H., Chung, S. 2024. Image processing and support vector machine (SVM) for classifying environmental stress symptoms of pepper seedlings grown in a plant factory. Agronomy, 14(9), 1-28. https://www.mdpi.com/2073-4395/14/9/2043/pdf?version=1725853039

[10] Pannakkong, W., Thiwa-Anont, K., Singthong, K., Parthanadee, P., Buddhakulsomsiri, J. 2022. Hyperparameter tuning of machine learning algorithms using response surface methodology: A case study of ANN, SVM, and DBN. Hindawi Mathematical Problems in Engineering, 2022(8513719), 1-17.

[11] Yang, N., Li, S., Liu, J., Bian, F. 2014. Sensitivity of support vector machine classification to various training features. TELKOMNIKA Indonesian Journal of Electrical Engineering, 12(1), 286-291.

[12] Gad, A. G. 2022. Particle swarm optimization algorithm and its applications: A systematic review. Archives of Computational Methods in Engineering, 29(12), 2531–2561. https://www.researchgate.net/publication/360057862_Particle_Swarm_Optimization_Algorithm_and_Its_Applications_A_Systematic_Review

[13] Zhang, D., Liu, J., Jiang, L., Bu, G., Hu, R., Luo, Y. 2020. The improvement of v-shaped transfer function of binary particle swarm optimisation. Advances in Swarm Intelligence, 12145, 202-211.

[14] Isiet, M., Gadala, M. 2020. Sensitivity analysis of control parameters in particle swarm optimization. *Journal of Computational* Science, 41(1), 1-13. https://dspace.adu.ac.ae/bitstream/handle/1/1850/gadala%20Sensitivity%20analysis%20of%20control%20parameters%20in%20particle%20swarmoptimization.pdf?sequence=1&isAllowed=y

[15] Yuan, Q., Zhang, Y., Dai, X., Zhang, S. 2022. A modified reptile search algorithm for numerical optimisation problems. Computational Intelligence and Neuroscience, 2022(975200), 1-20.

[16] Kailasam, J.k., Nalliah, R., Muthusamy, S.N., Manoharan, P. 2023. MLBRSA: Multi-learning-based reptile search algorithm for global optimisation and software requirements prioritisation problems. Biomimetics (Basel), 8(8), 1-49.

[17] Padmavathi, K., Thangadurai, K. 2016. Implementation of RGB and gray scale images in plant leaves disease detection: Comparative study. *Indian Journal of Science and Technology*, 9(6), 1-6. https://indjst.org/downloadarticle.php?Article_Unique_Id = INDJST 5373&Full_Text Pdf_Download=True

[18] Ayoade, O.B. 2025 Development of an optimised support vector machine for classification of cassava and maize diseases. A Ph.D. Thesis Submitted to the Department of Computer Sciences, Faculty of Natural Sciences, Ajayi Crowther University, Oyo, Nigeria. Unpublished Dissertation.

[19] Aslam, A., Khan, E., Beg, M. M. S. 2015. Improved edge detection algorithm for brain tumor segmentation. *Procedia Computer Science*, 58(2015), 430-437.

[20] Gou, J., Lei, Y. X., Guo, W. P., Wang, C., Cai, Y. Q., Luo, W. 2017. A novel improved particle swarm optimization algorithm based on individual difference evolution. Applied Soft Computing Journal, 57, 468–481.

[21] Chen, Y., Li, L., Peng, H., Xiao, J., Wu, Q. 2018. Dynamic multi-swarm differential learning particle swarm optimizer. Swarm Evolutionary Computation Journal, 39, 209-221.

[22] Lynn, N., Ali, M. Z, Suganthan, P. N. 2018. Population topologies for particle swarm optimization and differential evolution. Swarm Evolutionary Computation Journal, 39, 24-35.

[23] Khatami, A., Mirghasemi, S., Khosravi, A., Lim, C.P., Nahavandi, S. (2017). A new PSO-based approach to fire flame detection using K-Medoids clustering. Expert System Application, 68, 69-80.

[24] Lin, C.W., Yang, L., Fournier-Viger, P., Hong, T.P., Voznak, M. 2017. A binary PSO approach to mine high-utility item sets. Soft computing, 21, 5103-5121.

[25] Zhou, Y., Wang, N., Xiang, W. 2017. Clustering hierarchy protocol in wireless sensor networks using an improved PSO algorithm. IEEE Access, 5, 2241-2253.

[26] Chouikhi, N., Ammar, B., Rokbani, N., Alimi, A.M. 2017. PSO-based analysis of echo state network parameters for time series forecasting. Applied Soft Computing, 55, 211-225.

[27] Ma, Z., Yuan, X., Han, S., Sun, D., Ma, Y. 2019. Improved chaotic particle swarm optimisation algorithm with more symmetric distribution for numerical function optimisation. Symmetry, 11(876), 1-19.

[28] Khan, M. K., Zafar, M. H., Rashid, S., Mansoor, M., Raza Moosavi, S. K., Sanfilippo, F. 2023. Improved reptile search optimization algorithm: Application on Regression and classification problems. Applied Sciences, 13(945), 1-29. https://www.mdpi.com/2076-3417/13/2/945

[29] Zhou, L., Liu, X., Tian, R., Wang, W., Jin, G. 2025. A multi-strategy enhanced reptile search algorithm for global optimisation and engineering optimization design problems. Cluster Computing, 28(141), 1-41.

[30] Fu, Y., Liu, D., Chen, J., He, L. 2024. Secretary bird optimisation algorithm: a new metaheuristic for solving global optimisation problems. Artificial Intelligence Review, 57(123), 1-102. https://link.springer.com/content/pdf/10.1007/s10462-024-10729-y.pdf

[31] Zhao, S., Zhang, T., Ma, S., Chen, M. 2022. Dandelion optimizer: a nature-inspired metaheuristic algorithm for engineering applications. Engineering Application of Artificial Intelligence, 114(2), 1-28.

[32] Ibrahim, M. A., Ayotunde, O. O., Abeke, A. A. Samushudeen, B. O., Obiyemi, O. O. 2022. Development of hybrid learning technique for detection and classification of plant diseases. Adeleke University Journal of Engineering and Technology, 5(1), 87-102. http://aujet.adelekeuniversity.edu.ng/index.php/aujet/article/view/227/155

[33] Yang, J., Zhu, W., Liu, G., Dai, W., Xu, Z., Wan, L, Zhou, G. 2024. ICPNet: Advanced maize leaf disease detection with multidimensional attention and coordinate depthwise convolution. Plants, 13(2277), 1-21. https://www.mdpi.com/2223-7747/13/16/2277/pdf?version=1723725644

[34] Theerthagiri, P., Ruby, A.U., Chandran, J.G.C., Sardar, T.H., Ahamed S, B.M. 2024. Deep SqueezeNet learning model for diagnosis and prediction of maize leaf diseases. Journal of Big Data, 11(112), 1-16. https://journalofbigdata.springeropen.com/counter/pdf/10.1186/s40537-024-00972-z.pdf

[35] Bachhal, P., Kukreja, V., Ahuja, S., Kumar, Lilhore, U.K., Simaiya, S., Bijalwan, A., Alroobaea, R. Algarni, S. 2024. Maize leaf disease recognition using PRF-SVM integration: A breakthrough technique. Scientific Reports, 14(10219), 1-20. https://pmc.ncbi.nlm.nih.gov/articles/PMC11068775/pdf/41598_2024_Article_60506.pdf

[36] Kauri, K., Bansal, K. 2024. Enhancing Plant Disease Detection using Advanced Deep Learning Models. *Indian Journal of* Science and Technology, 17(17), 1755-1766.https://indjst.org/download-article.php?Article_Unique_Id=INDJST13524&Full_Text_Pdf_Download=True

[37] Ashurov, A.Y., Mehdhar S. A., Al-Gaashani , M., Samee, N.A., Alkanhel, R., Atteia, G., Abdallah, H.A., Muthanna, M.S. A. 2025. Enhancing plant disease detection through deep learning: A Depthwise CNN with squeeze and excitation integration and residual skip connections. Frontiers in Plant Science, 15, 1-16. https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2024.1505857/pdf

[38] Rani, R., Sahoo, J., Bellamkonda, S., Kumar, S. 2025. Attention-enhanced corn disease diagnosis using few-shot learning and VGG16, MethodsX, 2025(103172), 1-12. https://pmc.ncbi.nlm.nih.gov/articles/PMC11795141/pdf/main.pdf

[39] Aboelenin, S., Elbasheer, F.A., Eltoukhy, M.M., El-Hady, W.M., Hosny, K, M. (2025). A hybrid framework for plant leaf disease detection and classification using convolutional neural networks and vision transformer. Complex & Intelligent Systems, 11(142), 1-17. https://link.springer.com/content/pdf/10.1007/s40747-024-01764-x.pdf

[40] Nan, F., Song, Y., Yu, X., Nie, C., Liu, Y., Bai, Y., Zou, D., Wang, C., Yin, D., Yang, W., Jin, X. 2023. A novel method for maize leaf disease classification using the RGB-D post-segmentation image data. Frontiers in Plant Science, 14, 1-14. https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2023.1268015/pdf

[41] Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., Duan, Y. Al-Shamma, O., Santamaria, J., Fadhel, M.A. 2021. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. Journal of Big Data, 8(53), 1-74. https://journalofbigdata.springeropen.com/counter/pdf/10.1186/s40537-021-00444-8.pdf