

Research Article

Parrot Optimization Algorithm Integrated with YOLOv8 for Improving the Detection of Diseases in the Cotton Plant

Armaneesa Naaman Hasoon ¹ , Salwa Khalid Abdulateef ^{*,1} , Raya N. Ismail ¹ , Israa Rafaa Abdulqader ¹ 

¹ Department of Computer Science, Computer Science and Mathematics College, Tikrit University, Tikrit, 34001, Iraq

* Corresponding Author: Salwa Khalid Abdulateef, E-mail: khalid.salwa@tu.edu.iq

Article Info	Abstract
Article History	Agricultural crops are often susceptible to various diseases, which can significantly impact yield and economic viability. Cotton, a vital cash crop, is particularly vulnerable in this regard. Timely identification and diagnosis of cotton crop infections are crucial for protecting the crops and minimizing losses. This study proposes an innovative approach that integrates the YOLOv8 model with the Parrot Optimization Algorithm (POA) to optimize disease detection in cotton plants. The POA serves as an effective optimizer in enhancing weight selection of the YOLOv8 model and thereby improving its performance accuracy. A dataset of 4760 images encompassing four classes—diseased plant, leaf plant, fresh plant, and leaf—were used to fine-tune and evaluate the model. The results demonstrate high accuracy, with an mAP@50 of 96.3%, an mAP@95 of 96.2%, and F1-scores, precisions, and recalls of 93.3%, 93.2%, and 93.4%, respectively, confirming the model's improved detection capabilities. Compared to the base YOLOv8 model and other optimization techniques, the proposed model exhibits superior performance across various measures by effectively identifying infected areas on cotton leaves.
Received Apr 06, 2025	
Revised Jun 12, 2025	
Accepted Jul 05, 2025	
Keywords	
YOLOv8	
Cotton Plant	
Parrot Optimization	
Detection	
Cotton Diseases	
 Copyright: © 2025 Armaneesa Naaman Hasoon, Salwa Khalid Abdulateef, Raya N. Ismail, and Israa Rafaa Abdulqader. This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY 4.0) license.	

1. Introduction

Agriculture is a crucial sector that contributes significantly to revenue generation and fulfills the dietary needs of the population [1, 2]. In recent times, this sector is facing increasing pressure due to the rising demand for food. Growing global population has intensified the demand for food. Nations with fertile agricultural lands are better positioned to meet their food demands and bolster their economy through agricultural exports. Crop diseases pose a serious threat to agricultural productivity, as they lead to substantial reductions in yield and quality [3].

Cotton, often referred to as "white gold," is a leading fiber crop cultivated on approximately 2.5% of the world's agricultural land across nearly 100 countries [4]. It is a vital cash crop that contributes significantly to the textile industry. Therefore, effective plant protection is crucial in cotton production, as the crop is susceptible to various bacterial, fungal, and viral diseases, which can lead to a considerable

reduction in crop yield. Diseases affecting cotton plants include Anthracnose, alternaria leaf spot, fusarium wilt, bacterial blight, and boll rot. Approximately 80–90% of cotton plant diseases primarily affect the leaves [5].

The traditional approach of identifying cotton crop diseases involves deploying experts to visually inspect crops—a process that is labor-intensive and time-consuming. Continuous monitoring is necessary, and access to specialists can be limited in certain regions. The reliability of optical tracking for disease detection is often questionable, and conventional diagnostic methods may not yield accurate results. Phytopathology can lead to both qualitative and quantitative losses [6]. In recent times, researchers have been focusing on developing automated techniques to identify crop diseases, utilizing neural networks capable of learning and making complex decisions based on extensive datasets [7]. Computer vision and Artificial intelligence are two contemporary technologies that facilitate the independent identification of agricultural diseases without the need for human intervention [8-10]. Plant diseases incur significant financial losses for the global agriculture business [11, 12]. Therefore, utilizing an automated approach for disease detection would be more accurate, cost-effective, and efficient.

Optimization is a crucial procedure used to enhance the performance of various algorithms. Meta-heuristic search optimization is one category of optimization algorithms [13, 14]. This study introduces a technique, YOLO v8, which is one of the object detection techniques to improve the performance of cotton plant disease detection models. In addition, it proposes a Parrot Optimization Algorithm to optimize the weights of YOLOv8 models, which can aid in improving the accuracy and reliability of crop disease detection, thus saving money and time. Additionally, the model (YOLOv8-POA) utilizes gradient descent for practical training and convergence. The objectives of the study are:

1. To integrate YOLOv8 with Parrot Optimization Algorithm (POA).
2. To propose YOLOv8-POA as a new model that updates the weights of the YOLO technique for detecting cotton plant diseases.
3. To conduct comparative evaluations to assess the benefits and disadvantages of the current methods.

2. Literature Review

In this area of study, several deep learning models have been employed. Over the past few years, researchers have been working to develop autonomous disease detection systems for cotton. In 2020, researchers introduced a disease detection framework that utilizes a long short-term memory unit and a bidirectional recurrent neural network. This framework created a time series prediction dataset to forecast disease and pest conditions, achieving an average precision of 90.95%, an F1-score of 87.84%, and an accuracy of 87.84%, with an AUC value of 95.45% [15].

Another study proposed an effective system for detecting cotton plant diseases by capturing images

of diseased leaves and extracting their color features [16]. The authors compiled a dataset of plant leaf images, removed the RGB components, and converted them first to HSV and then to grayscale. They then trained an artificial neural network to detect the health status of the leaves and achieved an accuracy of 85%.

In another study, the researchers introduced prominent transfer learning techniques for classifying cotton diseases on fresh cotton plants and leaves using a small dataset obtained from Akash Zada [17]. This model, known as ResNet152V2, achieved an accuracy of 98.36%.

In 2022, researchers proposed a set of deep learning techniques, including convolutional neural network (CNN), VGG16, and ResNet50, as a meta learning technique to identify seven categories of cotton plant diseases [18]. The researchers compiled a dataset comprising 4,770 images collected from Matiari, in Sindh Province, Pakistan, and then pre-processed them by removing noise and applying data augmentation. This model achieved an accuracy of 98.53%.

Noon, et al. [19] presented an improved YOLOX model by enhancing spatial pyramid pooling. This model aggregated multiscale features using diverse pooling rates, achieving a mAP-50 of 72.31% across five classes of 1,112 images. Another group of researchers improved YOLOX for real-time detection, addressing the issue of sample imbalance and enhancing image feature extraction through focal loss function, channel attention, and hard-swish activation function [20]. They collected 5,760 images of cotton pests and diseases for this purpose, achieving accuracy, precision, and F1-scores of 94.60%, 94.04%, and 0.90, respectively.

In 2022, the YOLOv3 algorithm was employed in a study that utilized neural network techniques on 400 images to determine the health status of cotton leaves and plants. The model achieved 92.26% validation accuracy, 96.79% training accuracy, and a mAP of 95.09% [21]. Another study employed several transfer learning techniques, including InceptionV3, ResNet152V2, VGG16, InceptionResNetV2, Xception, MobileNetV2, and DenseNet121, to detect seven categories of cotton diseases and one healthy condition. It used 6,158 images, with VGG16 achieving the highest performance at an accuracy of 95.02% [22]. A study introduced a cutting-edge object detection, integrating the YOLO paradigm with ResNet-101 to identify diseases on cotton leaves, achieving an accuracy of 93%, with a low error rate of 6% [23].

Other studies have focused on real-time detection of cotton diseases. The authors of one such study proposed GVC-YOLO, a real-time detection model specifically designed for detecting aphid-damaged cotton leaves in an edge computing environment. By utilizing lightweight GSConv and VoVGSCSP modules, the backbone and neck networks of YOLOv8n were restructured to achieve optimized speed and accuracy, making them suitable for edge devices with limited computing resources. This model achieved mAP@0.5 and mAP@ 0.95 values of 90.3% and 97.9%, respectively [24]. The CDDLite-YOLO8 model was proposed for detecting cotton plant diseases in natural conditions [25]. This study explored real-time

detection, introducing the MPDIoU loss function and designing the PCDetect detection head, which integrates the PCD module and replaces some CBS modules with PCDetect. The dataset was private, comprising video durations of 15 to 30 seconds at a frame rate of 15 frames per second (FPS), achieving an mAP of 90.6%, 222.22 FPS, and an mAP@0.5 of 73.7%. This reduced accuracy may be attributed to background interference, as the leaf discoloration associated with the cotton verticillium wilt disease often blends with the soil due to similar coloration. Furthermore, cotton Fusarium wilt and Verticillium wilt diseases share similar colors, occasionally leading to misdetection.

Yang, et al. [26] presented an SRNet-YOLO framework for identifying "tiny pests" and "very tiny pests" in wild cotton crops. The SRNet-YOLO model incorporates a fusion mechanism based on BiFormer attention, a feature map super-resolution reconstruction module, and YOLOv8 for feature extraction, achieving mAP, precision, recall, and F1-score of 92.8%, 86.6%, 92.3%, and, 0.89% respectively.

In another study, the authors proposed an improved YOLOv9 model, replacing the RepLanLsk module with RepNCSPELAN4 to enhance feature extraction [27]. The dataset contained 4,000 image samples, divided into training and validation sets in the ratio of 80:20. The results of the study demonstrated a 93% accuracy rate, a 92.4% recall rate, an average precision of 96.4%, and mAP0.5 and mAP0.5:0.95 rates of 96.4% and 67.4%, respectively. However, the YOLOv9-LSBN model employed in this study utilized a limited number of pest and disease samples, resulting in poor diversity. Therefore, the dataset requires expansion and the implementation of slower detection methods.

Pantazi, et al. [1] classified cotton leaf diseases into three significant categories. Their proposed model for cotton leaf disease identification, utilizing convolutional neural networks, achieved a testing accuracy of 90%.

Although the recent methods yield promising results, they also present some challenges. Some models struggle to differentiate between diseases when their symptoms are similar, and others face limitations related to dataset size or computational requirements. Additionally, several models exhibit reduced accuracy at high intersection-over-union (IoU) thresholds. These limitations highlight the need for more effective methods that yield precise results while being computationally efficient. The current study proposes the use of the POA in conjunction with YOLOv8 to address these gaps identified in previous research. This model aims to enhance weight optimization, resulting in sharper and more reliable disease detection. The researchers anticipate that POA's exploratory capabilities, combined with YOLOv8's innovative detection methods, will facilitate faster, more accurate, and cost-effective identification of cotton plant diseases in the field.

3. Overview of Algorithms

This section will illustrate the details of the algorithms that are used in this paper.

3.1. The YOLOv8 Model

Studies by Xie, et al. [28] and Sohan, et al. [29] selected YOLOv8 to detect the disease of a cotton crop. YOLOv8 is efficient because it eliminates time-consuming processes by directly detecting objects, reduces inference times, and optimizes resources. The architecture of this model consists of three components: the backbone, the neck, and the head, as illustrated in Figure 1 [30]. The backbone extracts features from images using CNN, the neck combines feature maps from various stages of the backbone, and the head makes the final detection decisions [31].

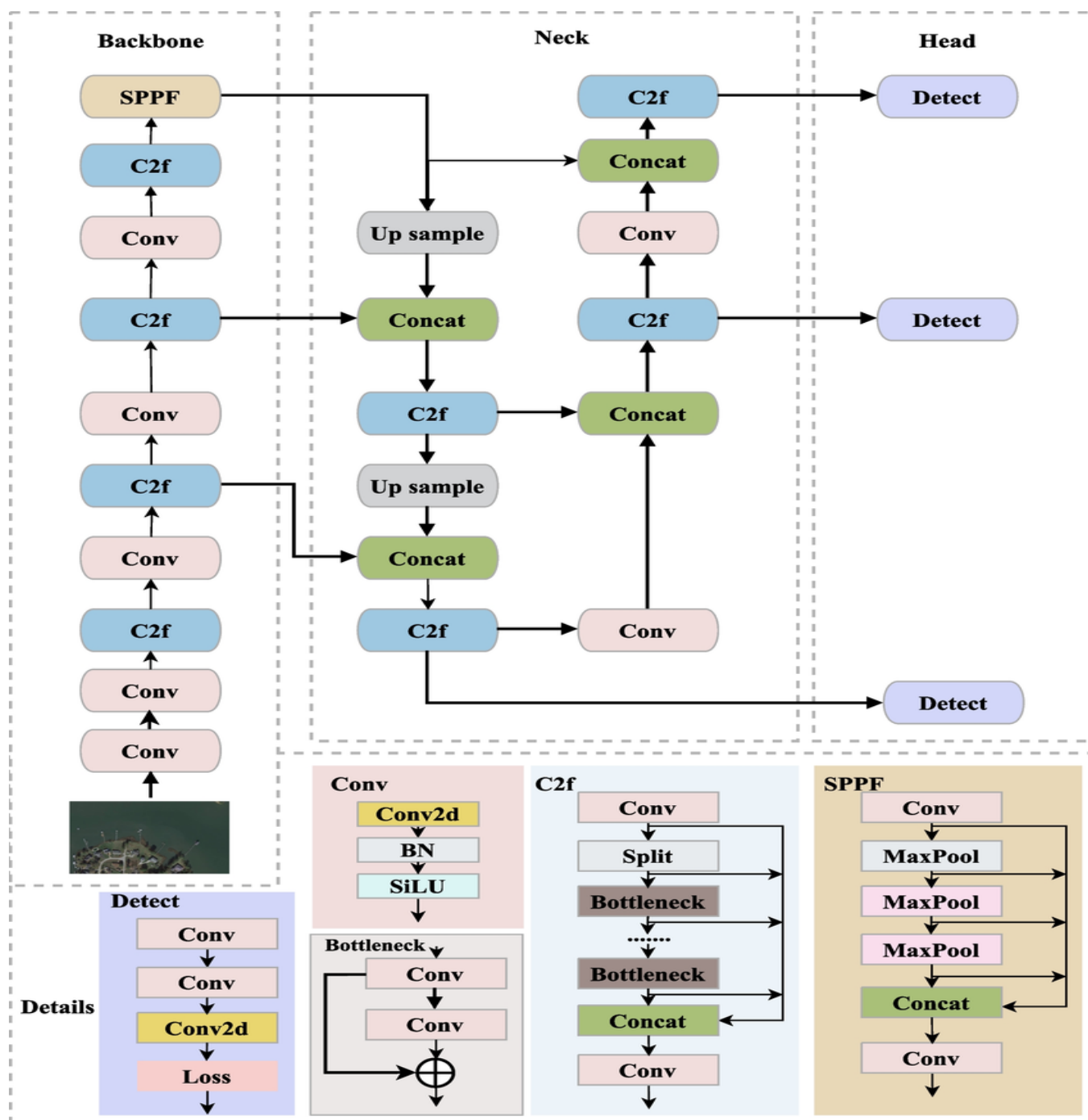


Figure 1. The YOLOv8 model

3.2. Parrot Optimization Algorithm

Lian, et al. [32] proposed an optimization algorithm based on *Pyrrhura molinae* parrot behavior (POA). POA seeks to tackle complicated optimization problems by replicating parrot behaviors such as foraging, staying, communication, and fear of strangers. This approach enables independent group member migration, which can be executed in parallel on hardware, thereby significantly reducing algorithm complexity. In addition, POA effectively navigates the search space while converging toward optimal solutions by leveraging both exploration and exploitation strategies. The algorithm's optimization process is partitioned into four segments based on parrot behavior models.

1. Foraging Behavior: A parrot's foraging activity involves locating food by observing its position or considering the owner's location and flying towards it. This behavior is applied in Equation (1) of the PO model, which governs positional movements.

$$x_i^{t+1} = (x_i^t - x_{best}).levy_{dim} + rand(0,1). \left(1 - \frac{t}{Max_{iter}}\right)^{\frac{2t}{Max_{iter}}} . x_{mean}^t \quad (1)$$

In this equation, x_{best} represents the ideal solution from the current iteration, x_i^t represents the position of the i^{th} particle at the t^{th} iteration, where x_i^{t+1} represents the position in the next iteration. x_{mean}^t indicates the average position of the population at the t^{th} iteration, and $levy_{dim}$ is the Lévy distribution of the problem's dimensionality, with t and Max_{iter} representing the current and maximum iterations, respectively.

2. Staying Behavior: A parrot lands on its owner's spot and remains stationary in that position. This behavior is applied in Equation (2), which governs progress.

$$x_i^{t+1} = x_i^t + x_{best}.levy_{dim} + rand(0,1).ones(1, dim) \quad (2)$$

3. Communicating Behavior: Parrots constantly share information within the flock, and can fly in and out of the flock at any time. The PO model mimics both these actions. The average position of the current population represents the center of the flock, as illustrated in Equation (3).

$$x_i^{t+1} = \begin{cases} 0.2 \cdot rand(0,1). \left(1 - \frac{t}{Max_{iter}}\right). (x_i^t - x_{mean}^t) & p \leq 0.5 \\ 0.2 \cdot rand(0,1).exp\left(-\frac{t}{rand(0,1).Max_{iter}}\right) & p > 0.5 \end{cases} \quad (3)$$

This formula uses $p \leq 0.5$ to align individuals with the population's center, and $p > 0.5$ to generate random placements that allow individuals to migrate away from the population.

4. Avoiding Strangers Behavior: Parrots avoid strangers and look for a secure environment with the owner.

The PO model imitates this behavior, as shown in Equation (4).

$$x_i^{t+1} = x_i^t + \text{rand}(0,1) \cdot \cos\left(0.5\pi \cdot \frac{t}{\text{Max}_{iter}}\right) \cdot (x_{best} - x_i^t) - \cos\left(\text{rand}(0,1) \cdot \pi \cdot \left(\frac{t}{\text{Max}_{iter}}\right)^{\frac{2}{\text{Max}_{iter}}}\right) \cdot (x_i^t + x_{best}). \quad (4)$$

Equation (5) constructs a collection of predetermined solutions that form the initial population during the algorithm's startup phase.

$$x_i^0 = 1b + \text{rand}(0,1) \cdot (ub - 1b) \quad (5)$$

where ub denotes upper boundaries, $1b$ denotes lower boundaries, and x_i^0 represents the initial population created. Iterative changes commence thereafter. The algorithm updates the individuals in the population by randomly selecting from the four stages. The PO algorithm dynamically updates each solution based on the current best solution, iterating until the maximum number of iterations is reached. Figure 2 presents a flowchart for the PO [33].

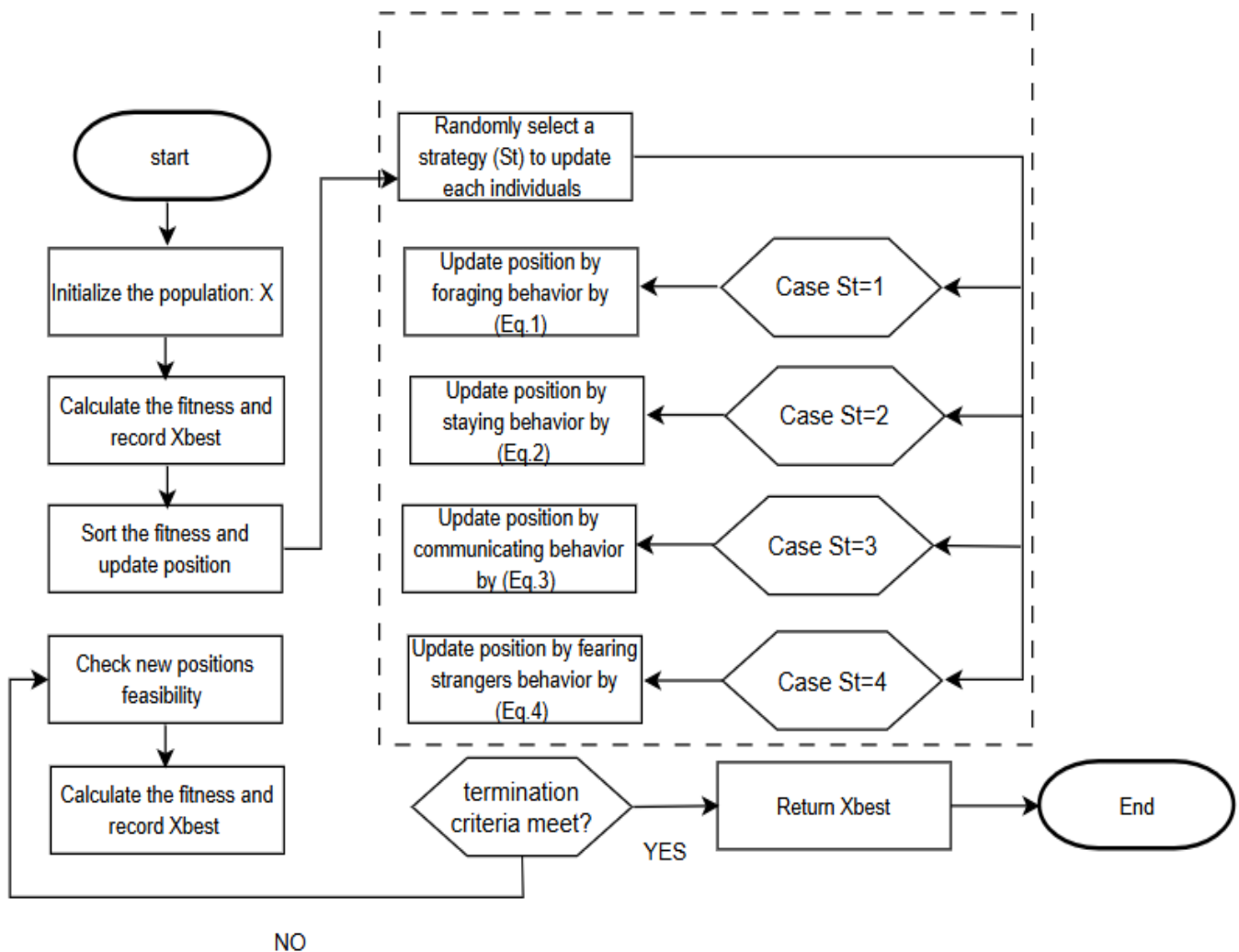


Figure 2. Flow Chart of POA

4. Methodology

This section outlines the steps used in the proposed YOLOv8-POA model for detecting cotton leaf disease. This new model utilizes a large dataset of images to train the base YOLOv8 model, and then implements a POA to optimize weights of the YOLOv8 for efficient disease detection. This methodology is illustrated in Figure 3.

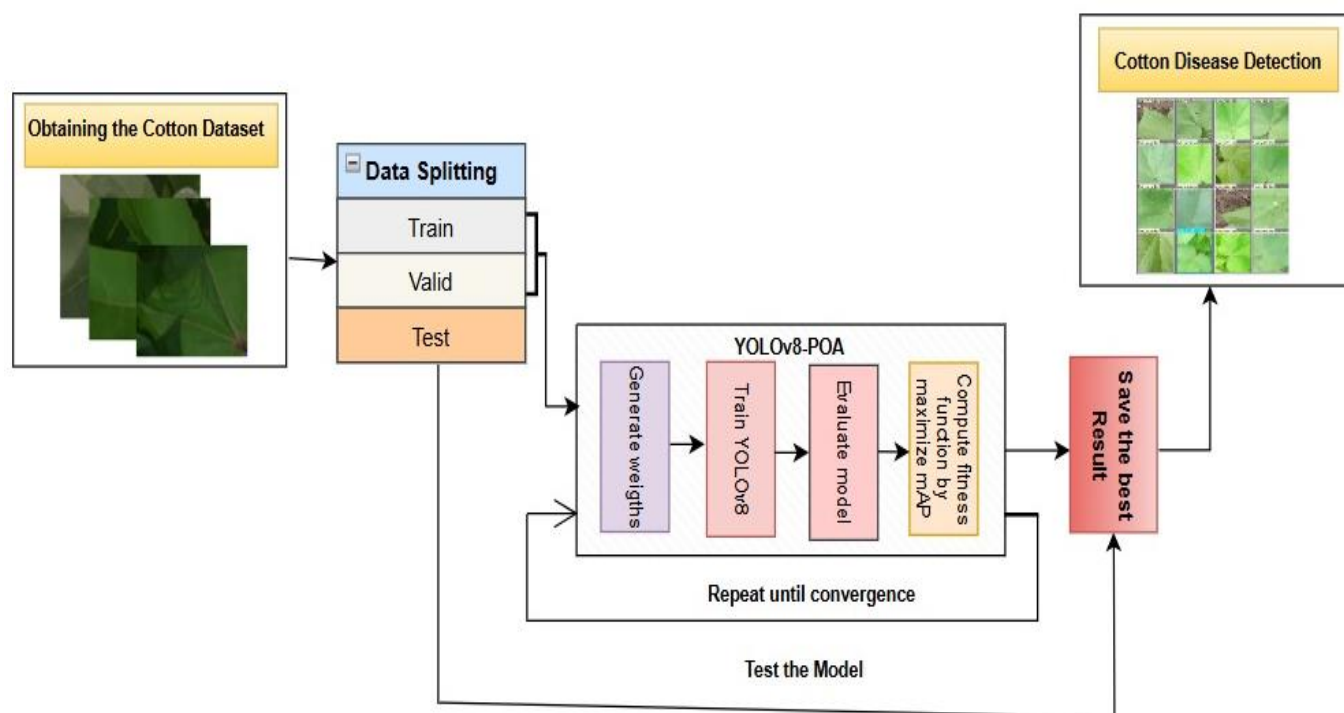


Figure 3. Methodology

4.1. Obtaining and Splitting the Dataset

The dataset used in the experiment was obtained from Roboflow [34]. The Pune Institute of Computer Technology has produced a dataset that serves as a comprehensive resource suitable for object detection tasks related to cotton crop disease identification. The images in this dataset have high resolution and include healthy samples, images of diseased leaves, and other plant sections. Each image is annotated to determine diseases and contains four classes: diseased plant, leaf plant, fresh plant, and leaf. Due to this careful labeling, the dataset enables models to accurately distinguish between diseased and healthy cotton plants in everyday indoor and outdoor settings. Therefore, the dataset is organized and the data are evenly distributed, which can help train reliable deep learning models for precision agriculture and plant pathology. Additionally, the images have a size of 640x640, and the applied augmentation is a horizontal flip. The total number of annotated images is 4,760, with the training set comprising 4,068 images, the validation set containing 454 images, and the test set comprising 238 images. Figure 4. Shows the samples of dataset used in this study.



Figure 4. Some Samples of the Dataset: a) Diseased cotton leaf, b) Fresh cotton plant, c) Diseased cotton leaf, and d) Fresh cotton leaf

4.2. Proposed YOLOv8-POA Object Detection Model

This study employs YOLOv8, which is used to perform real-time object detection tasks. Initially, the YOLOv8 model was applied to detect cotton leaf diseases using an annotated bounding box dataset. After obtaining and splitting the cotton plant dataset, the YOLOv8 model was implemented, utilizing the Albumentation library to preprocess images and enhance the model's performance. The model was then trained using the hyperparameters specified in Table 1.

Table 1. Hyperparameters of the trained YOLOv8 model

Hyperparameter	Value
Number of Epochs	100
Batch size	32
Image size	416*416
Learning rate	0.00125

YOLOv8 can scale effectively to handle more classes and a larger set of images by simply increasing annotations and retraining. However, the inclusion of the POA component requires additional effort, which increases with larger datasets. Enhancing the speed of particle evaluations or tuning the POA's hyperparameters (population size and iteration count) can help maintain good training times when working with more disease types or larger datasets.

To optimize the results obtained from training the base YOLOv8 model, we integrated the POA with the YOLOv8 model to generate optimal weights. As illustrated in Figure 3, the YOLOv8-POA process begins with loading the training and validation sets to train the model. The POA generates multiple particles (a set of candidate weights), which are then used to train the YOLOv8 model. After training, the model is evaluated to determine the mAP value. Then, the fitness function corresponding to the mAP value is computed as shown in Equation (6).

$$fitness_k = mAP_k \quad (6)$$

where k represents all potential weight vectors included in each candidate weight vector.

These steps involve iteratively adjusting the weight vectors until a termination condition is reached, ultimately obtaining the best weights that achieve a high mAP , thereby enhancing object detection capabilities. In the final step, we saved the best result and used it to test the model with the test set images to detect diseases in cotton plants.

4.3. Evaluation Metrics

This research utilized four metrics to assess the performance of the proposed model. Equation (7) calculates precision (P), which is the ratio of correctly predicted observations to the total predicted positives [13, 35]. Recall (R), computed from Equation (8), represents the ratio of correctly predicted objects to all ground truth objects [36]. Equations (9) and (10) detail the computation of average precision (AP), which implements average accuracy, and mean average precision (mAP), which implements the average value of AP [37]. The equations for P , R , AP , and mAP are as follows:

$$P = \frac{TP}{TP + FP} \quad (7)$$

$$R = \frac{TP}{TP + FN} \quad (8)$$

$$AP = \frac{\sum_i^N AP_i}{N} \quad (9)$$

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (10)$$

where TP denotes true positive predictions, TN denotes true negative predictions, N denotes the number of classes, FP denotes false positives, and FN denotes false negatives.

5. Results and Discussions

This section provides an overview of the performance of the YOLOv8-POA model in detecting diseases in cotton plants. The results were evaluated using various metrics, including $mAP@50$, $mAP@95$, precision, recall, and loss, for both training and validation. All experiments were conducted using Python 3.10.12 and an NVIDIA Tesla L4 GPU in the Google Colab Pro environment. Section 5.1 explains the quantitative results obtained from the proposed model using different optimization algorithms. Additionally, visual results for the proposed model are presented. Section 5.2 presents the inference time results. The results of the proposed model based on loss versus epoch during training are illustrated in Section 5.3. Section 5.4 presents the performance analysis of the proposed model (YOLOv8-POA) in comparison to previous studies.

5.1. Performance Results of YOLOv8 when Compared to Other Optimizers

Using a dataset of 4,760 images of cotton plant diseases, all models, baseline YOLOv8 (with Adam), YOLOv8 with particle swarm optimization (YOLOv8-PSO), YOLOv8 with genetic algorithm (YOLOv8-

GA), and YOLOv8 with POA (YOLOv8-POA), were given 100 epochs of training and fine-tuned at a learning rate of 0.00125. Their performance was tested using the same test set. Multiple metrics were computed after training; the values of these metrics are detailed in Table 2. Baseline YOLOv8 with Adam achieved 96.2% mAP@50 and 96.1% mAP@95. Switching to PSO improved the accuracy to 95.6% mAP@50 and 95.4% mAP@95. GA provided a slight accuracy increase ($\sim 0.3\%$), demonstrating its operational impact. POA achieved the highest mAP@50 (96.3%) and mAP@95 (96.2%). Its ability to distribute tasks across multiple channels allowed it to identify the most accurate weight combinations, resulting in significant accuracy gains.

From the results presented in Table 2, it is evident that applying YOLOv8-POA enhances every evaluation metric in the detection model. Consequently, the performance of the YOLOv8-POA model surpasses that of the base YOLOv8, as well as both YOLOv8-GA and YOLOv8-PSO. Figure 5 illustrates the visual identification of the four types of cotton leaf diseases, demonstrating that the proposed model yields superior results compared to the standard algorithm.



Figure 5. Visual results of YOLOv8-POA for cotton leaf disease detection

Table 2. Quantitative results proposed a model compared to other models

Model	mAP @50	mAP @95	Precision	Recall	F1-score
YOLOv8	96.2	96.1	93.1	92.4	92.7
YOLOv8-POA	96.3	96.2	93.2	93.4	93.3
YOLOv8-GA	95.9	95.8	92.9	93	92.9
YOLOv8-PSO	95.6	95.4	92.7	92.9	92.8

5.2. Analysis Based on Inference Time

With regard to the inference time, YOLOv8-POA exhibits a preprocessing time of 0.4 ms, an

inference time of 5.4 ms, a loss time of 0.0 ms, and a postprocessing time of 4.1 ms per image. YOLOv8-POA was rigorously tested to assess its suitability for agricultural applications requiring continuous object recognition. The experiments indicate that the total processing time for a single image is approximately 9.9 milliseconds, allowing for nearly 100 frames to be processed every second, significantly exceeding typical real-time standards (30 FPS). This result confirms that YOLOv8-POA can efficiently manage real-time tasks, including fruit detection, weed identification, and crop monitoring, on edge AI devices used in agriculture.

5.3. Loss vs. Epoch

Figure 6 presents three curves representing training box losses, which calculate the differences between the ground truth boxes and the expected bounding box coordinates. These curves begin at 2.5 and decrease with the increasing number of epochs, approaching zero, indicating the model's effective performance in detecting objects. The second curve named cls-loss represents classification loss, indicating how accurately class probabilities were predicted. This curve starts at 3 and gradually decreases, ultimately achieving a good classification accuracy. The third curve, Distribution Focal Loss (DFL), is utilized to enhance the distribution of bounding boxes to improve prediction quality, starting at 3.5 and decreasing over the course of training. The final values of all three curves indicate strong performance of the detection model. Figure 7 shows the effective convergence of the three loss curves for the YOLOv8-POA model, with all curves decreasing during early training, signifying improved efficiency on the training data and effective minimization of loss functions.

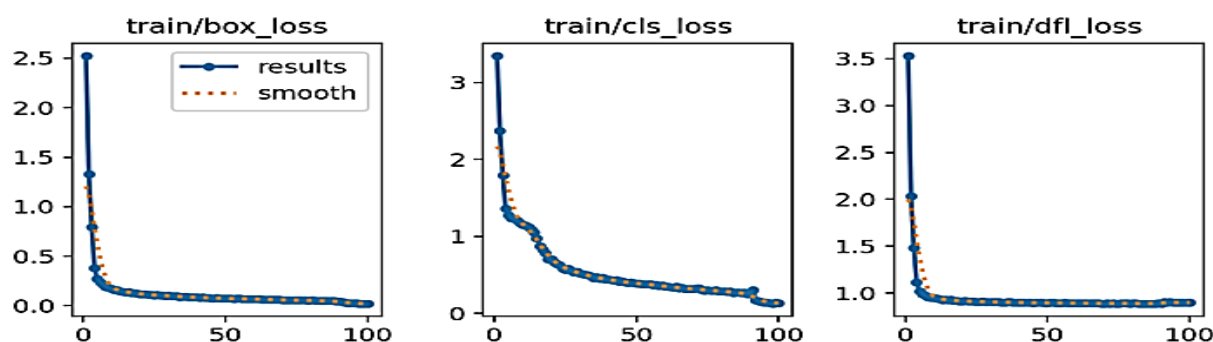


Figure 6. The curve of the loss in training YOLO8

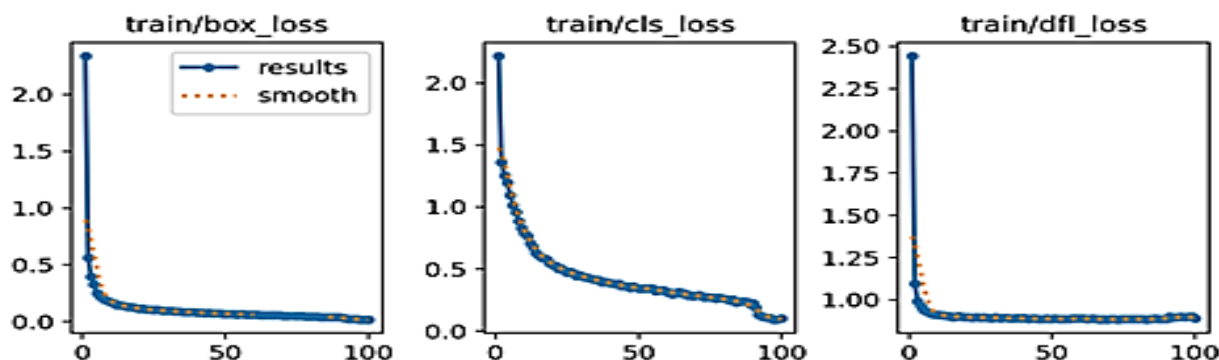


Figure 7. The curve of the loss in training POA-YOLO8

5.4. Comparative Analysis with the Most Recent Detection Models

We compared the YOLOv8-POA model with state-of-the-art models from literature. Table 3 provides a detailed overview of studies related to cotton leaf detection that use deep learning and computer vision models. Improvements in mAP, precision, recall, and computational resource consumption are highlighted in these studies. The comparative analysis underscores the effective performance of models in detecting cotton leaf diseases. The comparison of these studies reveals several significant advancements and compromises in YOLO-based models for cotton disease and pest identification. Zhang, et al. [20] successfully improved the YOLOX model using Focal Loss and the hard-Swish activation function, yielding strong results on a large dataset. Nevertheless, these models had clear limitations in differentiating between closely related disease symptoms, despite excellent overall metrics. Although Susa, et al. [21] achieved outstanding accuracy with the YOLOv3, its use of a much smaller dataset may limit its generalizability. Conversely, Pan, et al. [25] prioritized efficiency and resource constraints with the CDDLite-YOLO8 model, albeit at a notable cost of accuracy. While precision issues were evident, Yang, et al. [26] creatively addressed the challenge of detecting small pests using super-resolution reconstruction with the SRNet-YOLO model. He, et al. [27] demonstrated exceptional accuracy with the YOLOv9-LSBN model at standard thresholds but experienced a notable decline in performance at higher IoU thresholds. With consistent detection accuracy and localization across various thresholds, the proposed YOLOv8-POA model stands out as the most reliable and well-balanced solution, showcasing exceptional potential and utility in precision agriculture applications.

Table 3. Comparison with related research

Reference	Model	Size of the dataset	Evaluation Metrics				
			mAP@50	Precision	Recall	F1-score	mAP@95
Zhang, et al. [20]	Improve YOLOX	5760	94.04%	94.60%	-	0.90%	-
Susa, et al. [21]	YOLOv3	400	-	-	-	-	95.09%
Pan, et al. [25]	CDDLite-YOLO8	1530	90.6%	89.0%	86.6%	-	73.7%
Yang, et al. [26]	SRNet-YOLOv8	3759	-	86.6%	92.3%	0.89%	92.8 %
He, et al. [27]	YOLOv9-LSBN	4000	96.4%	96.4%	92.4%	-	67.4%
Proposed model	YOLOv8-POA	4760	96.3%	93.2%	93.4%	93.3%	96.2%

6. Limitations of the Study

The POA relies on multiple hyperparameters, such as population size and iteration count. Therefore, careful adjustment is essential, as imbalances can slow down the process without guaranteeing improvement. Given that the cotton-disease sample set is uniform (4,760 images classified into training, validation, and test sets labeled with four classes), the learned model may not be suitable for other crops, diseases, or imaging conditions, and may overfit when tested outside this scope.

7. Conclusions

This study proposes a new model, YOLOv8-POA, for detecting diseases in cotton plants. The experiment began with the implementation of the standard YOLOv8 model on a cotton plant dataset for disease detection. Subsequently, the weights of the base YOLOv8 were optimized using the Parrot Optimization Algorithm (POA). The YOLOv8-POA model achieved the best detection evaluation metrics, with mAP@50 values of 97.1, 97.0, and 96.3 for the training, validation, and test sets, respectively. The experimental findings indicate that the model proposed in this research is viable for rapid and precise detection of cotton diseases. In the future, we plan to use the YOLOv8-POA model to detect additional types of cotton diseases in real time through a smartphone application that prompts farmers to capture images of leaves, allowing prompt disease identification. Additionally, we plan to optimize other structures within YOLO algorithms, gather more images of tomatoes, potatoes, and citrus fruits from various regions and seasons, incorporate GAN-based style transfer in POA to reduce the need for manual annotation in new fields, and maximize accuracy while maintaining efficiency. We will also conduct ablation tests to isolate lighting and occlusion levels and outline planned experiments to measure robustness under controlled variations.

Acknowledgment: The authors gratefully acknowledge the constructive comments and guidance provided by the anonymous reviewers and the Editor-in-Chief.

Data Availability Statement: The image dataset used in this study is publicly available from the “Cotton Crop Health Monitoring – Disease” project on Roboflow [34]. Processed data and trained YOLOv8-POA model weights are available from the corresponding author upon reasonable request.

Funding: The authors declare that no funding was received for this research.

Declaration of Competing Interest: The authors declare they have no known competing interests.

Research Involving Human/Animal Participants: This article does not involve any studies conducted by authors on animals or human participants.

References

- [1] X. E. Pantazi, D. Moshou, and A. A. Tamouridou, "Automated leaf disease detection in different crop species through image features analysis and One Class Classifiers," *Computers and electronics in agriculture*, vol. 156, pp. 96-104, 2019, doi: <http://dx.doi.org/10.1016/j.compag.2018.11.005>.
- [2] B. D. Jadhav, P. M. Ghate, P. B. Chopade, P. N. Kota, and S. S. Kulkarni, "An Improved Model for Plant Disease Analysis Using Attention Based Capsule Network," *International Journal of Intelligent Engineering & Systems*, vol. 17, no. 5, pp. 12-21, 2024, doi: <https://doi.org/10.22266/ijies2024.1031.02>.
- [3] Y. Toda and F. Okura, "How convolutional neural networks diagnose plant disease," *Plant Phenomics*, 2019, doi: <http://dx.doi.org/10.34133/2019/9237136>.
- [4] B. Tugrul, E. Elfatimi, and R. Eryigit, "Convolutional neural networks in detection of plant leaf diseases: A review,"

- Agriculture*, vol. 12, no. 8, p. 1192, 2022, doi: <http://dx.doi.org/10.3390/agriculture12081192>.
- [5] L. Satyanarayana and D. C. Rao, "Applications of CNN in leaf diseases: A critical survey," in *ITM Web of Conferences*, 2023, vol. 54: EDP Sciences, p. 01014, doi: <http://dx.doi.org/10.1051/itmconf/20235401014>.
- [6] R. Manavalan, "Towards an intelligent approaches for cotton diseases detection: A review," *Computers and Electronics in Agriculture*, vol. 200, p. 107255, 2022, doi: <http://dx.doi.org/10.1016/j.compag.2022.107255>.
- [7] L. C. Ngugi, M. Abelwahab, and M. Abo-Zahhad, "Recent advances in image processing techniques for automated leaf pest and disease recognition—A review," *Information Processing in Agriculture*, vol. 8, no. 1, pp. 27-51, 2021, doi: <http://dx.doi.org/10.1016/j.inpa.2020.04.004>.
- [8] Z. Iqbal, M. A. Khan, M. Sharif, J. H. Shah, M. H. ur Rehman, and K. Javed, "An automated detection and classification of citrus plant diseases using image processing techniques: A review," *Computers and electronics in agriculture*, vol. 153, pp. 12-32, 2018, doi: <http://dx.doi.org/10.1016/j.compag.2018.07.032>.
- [9] A. Kalyanasundaram and M. K. Nair, "Straight through Estimator Based Convolution Neural Network with Parameterized Clipping Activation Function for Plant Disease Detection and Classification," *International Journal of Intelligent Engineering & Systems*, vol. 18, no. 3, pp. 1-13, 2025, doi: <https://doi.org/10.22266/ijies2025.0430.01>.
- [10] S. Sonawane and N. N. Patil, "Crop-Weed Segmentation and Classification Using YOLOv8 Approach for Smart Farming," *Journal of Studies in Science and Engineering*, vol. 4, no. 2, pp. 136-158, 2024, doi: <https://doi.org/10.53898/josse2024428>.
- [11] M. H. Najim, S. K. Abdulateef, and A. H. Alasadi, "Early detection of tomato leaf diseases based on deep learning techniques," *IAES International Journal of Artificial Intelligence*, vol. 13, no. 1, pp. 509-515, 2024, doi: <http://dx.doi.org/10.11591/ijai.v13.i1.pp509-515>.
- [12] M. Zekiwo and A. Bruck, "Deep Learning-Based Image Processing for Cotton Leaf Disease and Pest Diagnosis," *Journal of Electrical & Computer Engineering*, vol 2021, p. 9981437, 2021. <https://doi.org/10.1155/2021/9981437>
- [13] S. K. Abdulateef, "Evolutionary Optimization of Geometrical Image Contour Detection," *International Journal of Intelligent Engineering & Systems*, vol. 15, no. 2, pp. 287-297, 2022 doi: <https://doi.org/10.22266/ijies2022.0430.26>
- [14] H. M. Fadhil and M. I. Younis, "A Hybrid Meta-Heuristic Approach for Test Case Prioritization and Optimization," *Fusion: Practice & Applications*, vol. 14, no. 2, pp. 261-271, 2024, doi: <https://doi.org/10.54216/FPA.140221>
- [15] P. Chen, Q. Xiao, J. Zhang, C. Xie, and B. Wang, "Occurrence prediction of cotton pests and diseases by bidirectional long short-term memory networks with climate and atmosphere circulation," *Computers and Electronics in Agriculture*, vol. 176, p. 105612, 2020, doi: <http://dx.doi.org/10.1016/j.compag.2020.105612>.
- [16] Y. GSR, B. VK, and H. DA, "Cotton Leaf Disease Detection Using Artificial Neural Network," *International Journal of Advanced Research in Computer Science*, vol. 12, 2021.
- [17] P. Sivakumar, N. S. R. Mohan, P. Kavya, and P. V. S. Teja, "Leaf disease identification: Enhanced cotton leaf disease identification using deep CNN models," in *2021 IEEE International Conference on Intelligent Systems, Smart and Green Technologies (ICISSGT)*, 2021: IEEE, pp. 22-26, doi: <http://dx.doi.org/10.1109/ICISSGT52025.2021.00016>.
- [18] M. S. Memon, P. Kumar, and R. Iqbal, "Meta deep learn leaf disease identification model for cotton crop," *Computers*, vol. 11, no. 7, p. 102, 2022, doi: <http://dx.doi.org/10.3390/computers11070102>.
- [19] S. K. Noon, M. Amjad, M. A. Qureshi, and A. Mannan, "Handling severity levels of multiple co-occurring cotton plant diseases using improved YOLOX model," *IEEE Access*, vol. 10, pp. 134811-134825, 2022, doi: <http://dx.doi.org/10.1109/ACCESS.2022.3232751>.
- [20] Y. Zhang, B. Ma, Y. Hu, C. Li, and Y. Li, "Accurate cotton diseases and pests detection in complex background based on an improved YOLOX model," *Computers and Electronics in Agriculture*, vol. 203, p. 107484, 2022, doi: <http://dx.doi.org/10.1016/j.compag.2022.107484>.
- [21] J. A. B. Susa, W. C. Nombrefia, A. S. Abustan, J. Macalisang, and R. R. Maaliw, "Deep learning technique detection for cotton and leaf classification using the YOLO algorithm," in *2022 International Conference on Smart Information Systems*

- and Technologies (SIST), 2022: IEEE, pp. 1-6, doi: <http://dx.doi.org/10.1109/SIST54437.2022.9945757>.
- [22] A. Herok and S. Ahmed, "Cotton leaf disease identification using transfer learning," in *2023 International conference on information and communication technology for sustainable development (ICICT4SD)*, 2023: IEEE, pp. 158-162, doi: <http://dx.doi.org/10.1109/ICICT4SD59951.2023.10303323>.
- [23] M. S. Maqbool, R. Nazeer, A. Basit, and K. Zahra, "Automated detection and localization of fungal infections on cotton leaves using YOLO-based object detection model," *Machines and Algorithms*, vol. 2, no. 2, pp. 121-136, 2023.
- [24] Z. Zhang et al., "GVC-YOLO: A Lightweight Real-Time Detection Method for Cotton Aphid-Damaged Leaves Based on Edge Computing," *Remote Sensing*, vol. 16, no. 16, p. 3046, 2024, doi: <http://dx.doi.org/10.3390/rs16163046>.
- [25] P. Pan et al., "Lightweight cotton diseases real-time detection model for resource-constrained devices in natural environments," *Frontiers in Plant Science*, vol. 15, p. 1383863, 2024, doi: <http://dx.doi.org/10.3389/fpls.2024.1383863>.
- [26] S. Yang, G. Zhou, Y. Feng, J. Zhang, and Z. Jia, "SRNet-YOLO: A model for detecting tiny and very tiny pests in cotton fields based on super-resolution reconstruction," *Frontiers in Plant Science*, vol. 15, p. 1416940, 2024, doi: <http://dx.doi.org/10.3389/fpls.2024.1416940>.
- [27] R. He et al., "YOLOv9-LSBN: An improved YOLOv9 model for cotton pest and disease," in *IEEE Access*, vol. 13, pp. 136536-136553, 2025, doi: <https://doi.org/10.1109/ACCESS.2025.3578967>.
- [28] D. Xie, W. Yao, W. Sun, and Z. Song, "Real-time identification of strawberry pests and diseases using an improved YOLOv8 Algorithm," *Symmetry*, vol. 16, no. 10, p. 1280, 2024, doi: <http://dx.doi.org/10.3390/sym16101280>.
- [29] M. Sohan, T. Sai Ram, and C. V. Rami Reddy, "A review on yolov8 and its advancements," in *International Conference on Data Intelligence and Cognitive Informatics*, 2024: Springer, pp. 529-545, doi: http://dx.doi.org/10.1007/978-981-99-7962-2_39.
- [30] G. Yao, S. Zhu, L. Zhang, and M. Qi, "Hp-yolov8: high-precision small object detection algorithm for remote sensing images," *Sensors*, vol. 24, no. 15, p. 4858, 2024, doi: <http://dx.doi.org/10.3390/s24154858>.
- [31] E. Casas, L. Ramos, C. Romero, and F. Rivas-Echeverria, "A comparative study of YOLOv5 and YOLOv8 for corrosion segmentation tasks in metal surfaces," *Array*, vol. 22, p. 100351, 2024, doi: <http://dx.doi.org/10.1016/j.array.2024.100351>.
- [32] J. Lian et al., "Parrot optimizer: Algorithm and applications to medical problems," *Computers in Biology and Medicine*, vol. 172, p. 108064, 2024, doi: <http://dx.doi.org/10.1016/j.combiomed.2024.108064>.
- [33] T. Chen and Y. Yi, "Multi-Strategy Enhanced Parrot Optimizer: Global Optimization and Feature Selection," *Biomimetics*, vol. 9, no. 11, p. 662, 2024, doi: <http://dx.doi.org/10.3390/biomimetics9110662>.
- [34] Roboflow. Cotton Crop Health Monitoring - Disease Computer Vision Model [Online] Available: <https://universe.roboflow.com/pune-institute-of-computer-technology/cotton-crop-health-monitoring-disease/dataset/8/download>
- [35] M. S. Z. Abid, B. Jahan, A. Al Mamun, M. J. Hossen, and S. H. Mazumder, "Bangladeshi crops leaf disease detection using YOLOv8," *Heliyon*, vol. 10, no. 18, p. e36694, 2024, doi: <https://doi.org/10.1016/j.heliyon.2024.e36694>
- [36] A. T. Khalaf and S. K. Abdulateef, "Ophthalmic diseases classification based on YOLOv8," *Journal of Robotics and Control (JRC)*, vol. 5, no. 2, pp. 408-415, 2024, doi: <http://dx.doi.org/10.18196/jrc.v5i2.21208>.
- [37] H. Lou et al., "DC-YOLOv8: small-size object detection algorithm based on camera sensor," *Electronics*, vol. 12, no. 10, p. 2323, 2023, doi: <http://dx.doi.org/10.3390/electronics12102323>.