Journal of Soft Computing Exploration

Homepage: https://shmpublisher.com/index.php/joscex

p-ISSN: 2746-7686
e-ISSN: 2746-0991

.

A deep learning-based leaf aphid detection approach using YOLOv8

Styawati[1], Heni Sulistiani[2], Ajeng Savitri Puspaningrum[3], Debby Alita[4], S. Samsugi[5], Vanisa Adelia Putri[6]

[1]Department of Information Systems, Universitas Teknokrat Indonesia, Indonesia
[2,4]Department of Informatics, Universitas Teknokrat Indonesia, Indonesia
[3,5,6]Department of Computer Engineering, Universitas Teknokrat Indonesia, Indonesia

Article Info

Article history:

Received Mar 10, 2026
Revised Mar 22, 2026
Accepted Apr 03, 2026

Keywords:

YOLOv8
Aphids
Smart agriculture
IoT
Object detection

ABSTRACT

Aphids pose a serious threat to agricultural productivity due to their rapid reproduction and their role as plant virus vectors. Early manual detection is difficult due to the pests' microscopic size and tendency to hide under leaves. This study aims to develop an accurate and real-time aphid monitoring system using the YOLOv8 algorithm. The model was trained using four epoch scenarios (30, 50, 100, and 200) to identify the best configuration to address the challenges of small, overlapping objects and varying leaf backgrounds. The results showed that increasing the number of epochs positively correlated with model performance, with the 200-epoch scenario providing the most optimal results with 91.5% accuracy, 0.87 recall, 0.89 F1-score, and 0.915 mAP50. The model was then integrated into a smart monitoring dashboard that synchronizes visual detection results with IoT sensor data (temperature, humidity, and nutrients) in real time. This system not only validates the reliability of YOLOv8 under field conditions, but also provides an effective early warning system to support rapid decision-making in crop protection management.

Corresponding Author:

Styawati,
Department of Computer Engineering,
Universitas Teknokrat Indonesia,
Jl. Zainal Abidin Pagaralam No. 9-11, Labuhan Ratu, Bandarlampung.
Email: styawati@teknokrat.ac.id
https://doi.org/10.52465/joscex.v7i1.9

1. INTRODUCTION

Aphid infestations on various agricultural commodities have consistently been reported to cause significant crop losses, reduce product quality, and act as major vectors for various plant viral diseases [1]. Aphids have an exponential reproductive rate and are microscopic in size, often clustering densely on the undersides of leaves [2]. These biological characteristics make accurate early detection crucial and urgent to prevent pest outbreaks that could threaten food security [3].

Current conventional pest monitoring methods still rely heavily on manual visual inspection by farmers or field extension officers. This approach is subjective, very time-consuming, prone to human fatigue (human error), and is not operationally feasible for continuous use over large plantation areas [4]. As an alternative, the development of advanced image acquisition technologies such as hyperspectral sensors, Near-Infrared (NIR), and electronic noses has been explored. While effective, these technologies require very high hardware costs, complex calibration, and heavy computation, making them impractical for widespread implementation, especially at the smallholder level [5].

Currently, many researchers use deep learning technology, especially Convolutional Neural Network (CNN), to classify and detect plant diseases and pests [5]. However, conventional CNNs and two-stage detector models such as Faster R-CNN have several limitations in their application in agriculture. These models require quite high computation and relatively long processing times, making them less suitable for real-time use on devices with limited capabilities (edge devices) [6]. In addition, traditional CNNs often experience decreased accuracy, especially in detecting very small objects, such as aphids, which often overlap and have a color similar to the leaf surface. Uncontrolled environmental conditions also further complicate the detection process, thereby increasing the possibility of detection errors, especially in the form of failure to recognize the target object [7].

The weaknesses of conventional CNNs can be overcome by the YOLO (You Only Look Once) algorithm, which has evolved into an efficient one-stage object detector. The latest development, YOLOv8, offers significant improvements in accuracy and speed compared to previous YOLO versions and traditional CNNs [8]. YOLOv8 uses an anchor-free approach and enhances the feature extraction capabilities of the backbone. These improvements enable the model to detect very small objects more clearly and determine their positions more accurately, while reducing the computational burden, making it suitable for real-time detection [9].

Specifically for aphid management, several recent studies have applied deep learning approaches. Zhang et al. [10] and Li et al. [11] developed YOLOv5-based aphid colony detection and recognition methods on sorghum and bell pepper plants. Furthermore, lightweight models such as Aphid-YOLO have been proposed to support detection in field environments [12]. However, the development of YOLOv8-based models-such as GVC-YOLO on cotton leaves-remains relatively limited and has not been widely explored [13].

A literature review reveals several important research gaps. First, most previous models were developed under controlled environmental conditions or focused on only a single commodity, thus limiting their generalizability when applied to varying field conditions, such as changes in lighting, shading, and background complexity [14]. Second, the challenge of detecting aphids, which are very small and often overlap in dense colonies, remains unaddressed. Furthermore, most research focuses solely on detection and has not yet integrated with field-application monitoring or early warning systems [15].

To address this gap, this study aims to develop an aphid monitoring system that is not only accurate but also easy to implement in the field. The proposed system utilizes the YOLOv8 algorithm to detect aphids, which is trained using data augmentation techniques to improve the model's ability to deal with variations in leaf backgrounds and small, overlapping objects. Furthermore, the developed detection model is integrated with a real-time monitoring dashboard, enabling direct visualization of plant conditions and supporting rapid and accurate decision-making in pest control.

2. METHOD

In the data preparation stage, the dataset used in this study was obtained from the Kaggle platform, consisting of 4,230 training data sets, 695 validation data sets, and 556 test data sets. This dataset is a collection of aphid images collected from various environmental conditions. Each image then underwent an image annotation process by labeling it with a bounding box to indicate the location and type of pest object in the image, resulting in an annotated dataset ready for use in the model training process.

In the pre-processing stage, the images are processed through automatic orientation and resized to 640 × 640 pixels to meet the input specifications of the YOLOv8 model [16], [17]. This process is followed by the application of data augmentation techniques, such as rotation, flipping, brightness and saturation adjustments, cropping, and blurring. This augmentation aims to increase data diversity, strengthen the model's generalization capability, and reduce the risk of overfitting [18]–[21].

The next stage is model training, where the processed dataset is used to train the YOLOv8 algorithm using a transfer learning approach from a previously trained model. This algorithm was chosen because of its ability to detect objects quickly and accurately in one process stage (real-time object detection), making it effective for recognizing visual patterns and localizing pest objects in images [22]–[24].

Next, in the evaluation and iteration stage, the trained model is tested using test data to measure its performance. The evaluation is conducted using object detection metrics, namely mean average precision

.

(mAP), precision, recall, and F1-score, to assess the model's effectiveness in detecting aphids on leaf surfaces. If the evaluation results do not meet the expected targets, improvements are made by repeating the pre-processing stage or retraining the model. Conversely, if the model's performance meets the established criteria, the model is declared ready for implementation in a pest detection system. The workflow for detecting aphid pests using the Yolov8 method is shown in Figure 1.

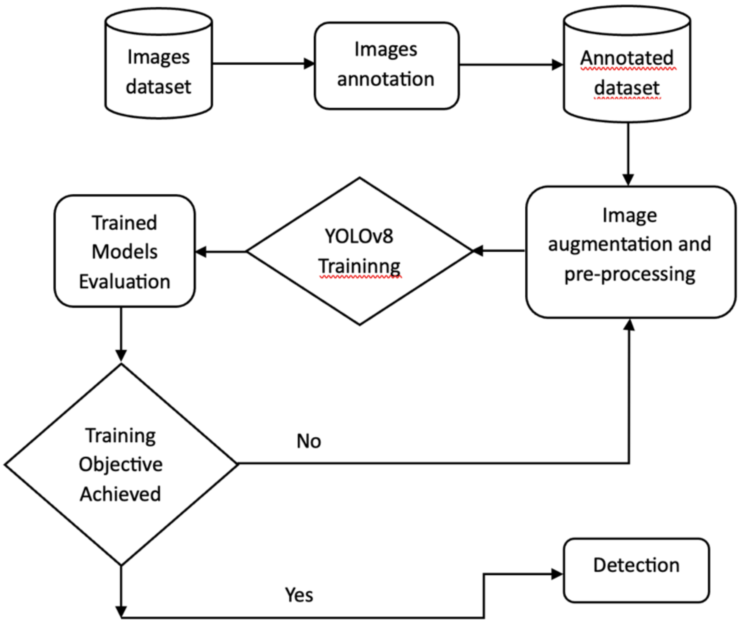


Figure 1. Aphid pest detection workflow using the Yolov8 method

Data Collection

The data used in this study is a secondary dataset obtained from the public repository platform Kaggle. The dataset consists of a total of 4,230 digital images depicting various species of aphids on the surface of plant leaves.

Figure 2. The original input image before the detection process

Preprocessing Data

The data preprocessing stage is a crucial phase in this research to mitigate the high variability in raw images before entering the model training stage [25]. In this phase, an automatic orientation procedure is applied to correct misalignments caused by variations in shooting angles in uncontrolled agricultural environments [26]. This orientation normalization ensures a uniform visual representation of leaves and aphid colonies, thereby increasing the model's effectiveness in learning spatial patterns. Simultaneously, all images are systematically resized to a standard resolution of 640 x 640 pixels. This dimensionality standardization is crucial for maintaining the consistency of the input tensor dimensions, stabilizing the convolution operations in the network, as well as accelerating convergence and improving the model's detection accuracy [4].

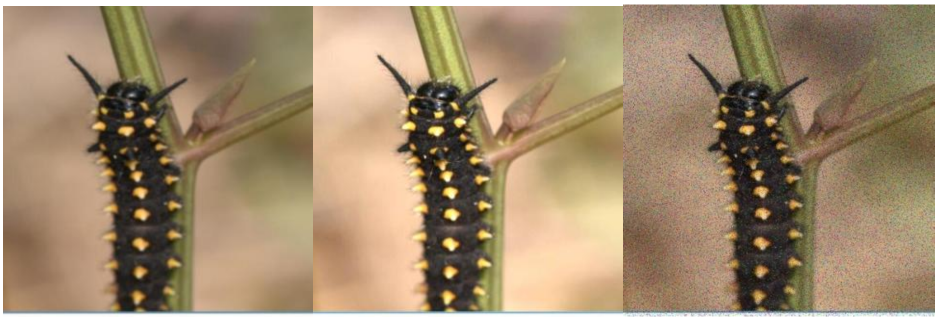
Figure 3. The result of normalizing the image resolution to 640 x 640 pixels

Augmentation Data

Data augmentation was implemented in this study as a strategic approach to increase the diversity and representativeness of the training dataset, which is crucial for improving the robustness and generalization performance of the YOLOv8 detection model [27]. Given that aphids are very small objects that often occur in clusters and often blend into the natural texture and pigmentation patterns of leaf surfaces, a limited or homogeneous dataset would significantly limit the model's ability to accurately distinguish aphids from background noise [28]. Therefore, the augmentation process was designed to artificially introduce a variety of visual scenarios that closely resemble the complex and dynamic conditions encountered in real agricultural environments [29].

The original images were subjected to a series of carefully controlled geometric and photometric transformations to simulate changes that might occur during actual image acquisition in the field. These transformations included horizontal flips to represent different leaf orientations, as well as limited rotation adjustments ranging from -15° to +15° to account for variations in camera angle and perspective. In addition to the geometric variations, random cropping with a maximum zoom factor of 30% was applied to alter the scale and position of aphids within the frame, allowing the model to learn a more spatially adaptive representation of aphids' presence across different regions of the leaf surface.

Next, photometric adjustments were introduced to mimic the effects of changing environmental illumination and atmospheric conditions. Brightness and saturation values were modified within predetermined limits to reproduce exposure differences caused by inconsistent sunlight intensity, shadows, or cloud cover. A slight Gaussian blur, limited to a maximum radius of one pixel, was also introduced to a portion of the images to simulate minor motion blur or focus imperfections that may occur during image capture. Additionally, grayscale conversion was applied to approximately 15% of the dataset to reduce overreliance on color information and encourage the model to focus on shape, contour, and texture-based features. Consequently, each original image yielded multiple transformed variants, significantly expanding the dataset and enabling the YOLOv8 model to develop a more robust and more invariant feature representation for aphid detection.



.

(a) (b) (c)

Figure 4. Application of data augmentation techniques on pest images: (a) Original image, (b) Image after brightness adjustment, (c) Image with the addition of Gaussian noise.

Figure 5. Result of rotating the image of the aphid object

YOLOv8 Model

This study adopted the YOLOv8 architecture as the primary detection model due to its superior balance of computational speed and accuracy in real-time scenarios [30], [31]. As a single-stage detector algorithm, YOLOv8 integrates object localization and classification processes in a single forward pass phase. This mechanism significantly reduces computational overhead without sacrificing detection accuracy. This characteristic is crucial for identifying aphids, which are very small and often occur in dense colonies, enabling the model to quickly and precisely determine the position of each pest on the leaf surface.

During the development phase, a transfer learning strategy was implemented by fine-tuning YOLOv8's pre-trained weights using a specific aphid dataset. This approach allows the model to leverage previously learned fundamental visual knowledge-such as edge, shape, and texture detection-while adapting to the unique visual features of aphids in complex agricultural environments. The model was specifically configured to detect a single target class by generating bounding boxes around individual pests or groups. Through this enhanced architecture, YOLOv8 exhibits high sensitivity to micro-sized objects, enabling it to distinguish aphids from the often visually similar natural patterns of leaf surfaces.

Model Evaluation

The performance of the trained YOLOv8 model is evaluated using standard object detection metrics to comprehensively assess its capability in detecting and localizing aphids on plant leaves. The primary measurement used in this study is mean Average Precision (mAP), which represents the overall detection accuracy by considering both the precision of bounding box localization and the correctness of class prediction. In addition to mAP, precision and recall are employed to evaluate the model's ability to minimize false detections while simultaneously maximizing the detection of actual aphid instances. These metrics provide a balanced perspective on how effectively the model distinguishes aphids from background elements and other non-target objects.

Furthermore, the F1-score is calculated as the harmonic mean of precision and recall to provide a single, comprehensive indicator of model performance. This metric is particularly important in aphid detection tasks, where an imbalance between false positives and false negatives may significantly affect the reliability of the system. A high F1-score indicates that the model is able to maintain both strong detection accuracy and consistency across varied image conditions.

The evaluation process is carried out using a separate testing dataset that was not included in the training phase to ensure an objective and unbiased assessment of the model's generalization ability. Testing images include variations in lighting, leaf texture, aphid density, and background complexity to simulate realistic field conditions. Through this evaluation, the overall effectiveness and robustness of the YOLOv8 model in detecting aphid infestations on plant leaves can be systematically determined.

3. RESULTS AND DISCUSSIONS

In this study, the model was tested through four training scenarios: 30, 50, 100, and 200 epochs.

Model Training with 30 Epochs

In this training scenario, the YOLOv8 model demonstrated a stable learning process. This was demonstrated by a decrease in the training loss and validation loss values for all components: box loss, classification loss, and distribution focal loss. This decrease in loss indicates that the model is improving at performing bounding box regression, object classification, and adjusting object location predictions.

In terms of performance evaluation, the precision and recall values showed an increasing trend until the end of training. In the final epoch, the precision reached approximately 0.836, the recall approximately 0.806, the mAP50 value reached 0.851, and the mAP50-95 reached 0.614. These results indicate that the model has demonstrated sufficient object detection capabilities in the 30-epoch scenario. In general, this scenario indicates that the model has begun to recognize target object patterns well and can be used as a baseline for comparing results over a larger number of epochs. The graphs of the loss metrics, model detection performance, and image results using YOLOv8 at epoch 30 are presented in Figure 6 and 7.

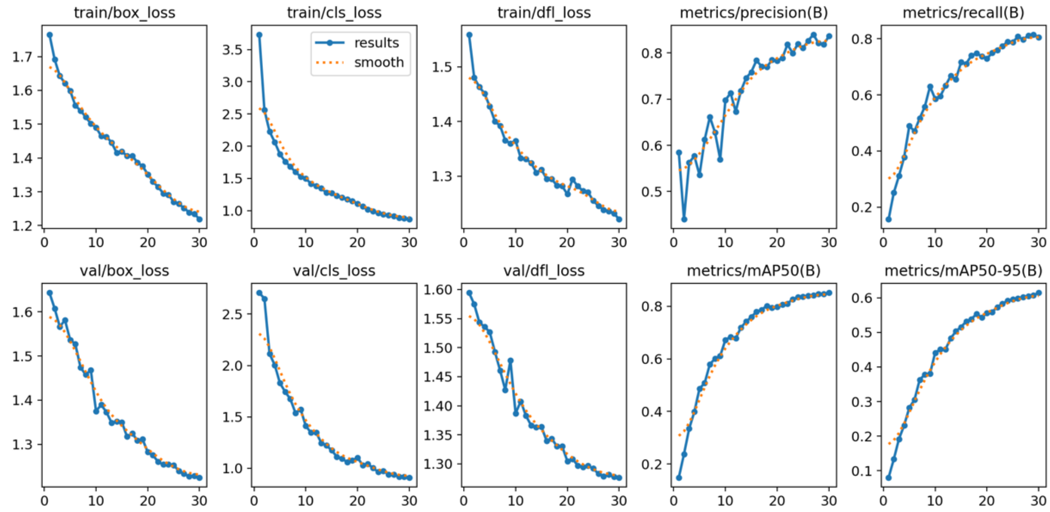


Figure 6. Loss metric graph and detection performance of YOLOv8 model in 30 epoch scenario

Figure 7. Batch validation data visualization with bounding boxes and class labels from YOLOv8 training epoch 30

.

Model Training with 50 Epochs

In the 50-epoch training scenario, the YOLOv8 model demonstrated a steady increase in performance. This was demonstrated by a decrease in the training loss and validation loss values across all components: box loss, classification loss, and distribution focus loss. This decrease indicates that the model is improving at object localization, classification, and box boundary refinement.

In terms of evaluation metrics, the precision, recall, mAP50, and mAP50-95 values increased until the end of training. In the final epoch, the model achieved a precision of around 0.85, a recall of around 0.82, a mAP50 of around 0.87, and a mAP50-95 of around 0.64. These results indicate that the model has good detection capabilities and fairly stable generalization on validation data. In general, training with 50 epochs yields more optimal results than training with 30 epochs, as the model has a longer learning period to recognize object patterns more accurately. The graphs of the loss metrics, model detection performance, and image results using YOLOv8 at epoch 50 are presented in Figure 8 and 9.

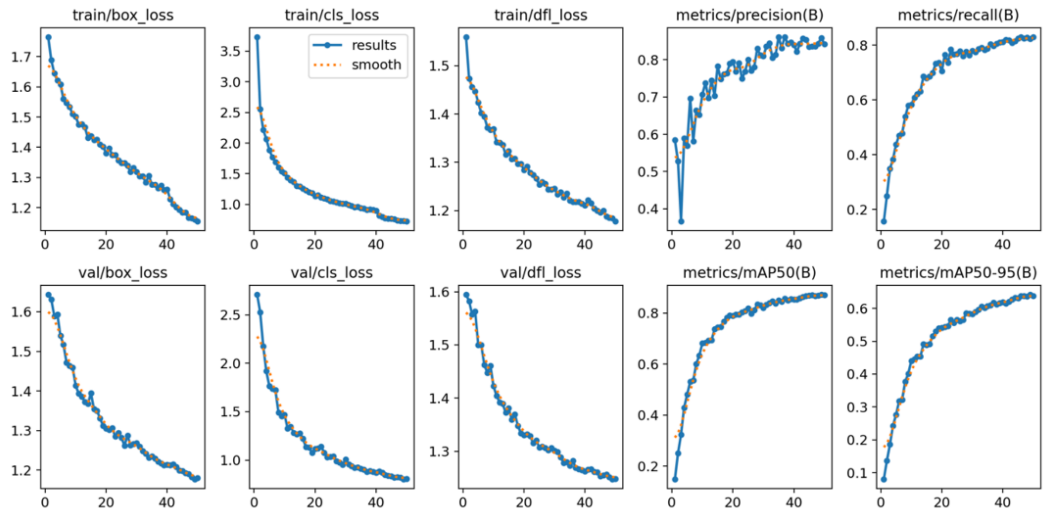


Figure 8. Loss metric graph and detection performance of YOLOv8 model in 50 epoch scenario

Figure 9. Batch validation data visualization with bounding boxes and class labels from YOLOv8 training epoch 50

Model Training with 100 Epochs

After 100 epochs of training, the YOLOv8 model demonstrated good convergence. This was demonstrated by a decrease in the training loss and validation loss values for all components: box loss, classification loss, and distribution focal loss. This decrease indicates that the model is increasingly effective in object localization, classification, and bounding box position refinement. From an evaluation perspective, the precision, recall, mAP50, and mAP50-95 values continued to increase until the end of training. In the final epoch, precision was in the range of 0.88–0.89, recall around 0.85, mAP50 around 0.89–0.90, and mAP50-95 around 0.68-0.69. These results indicate that the model has excellent detection capabilities and stable generalization on the validation data. This is due to the model's long learning period, which allows it to recognize object patterns more accurately and consistently. The graphs of the loss metrics, model detection performance, and image results using YOLOv8 at epoch 100 are presented in Figure 10 and 11.

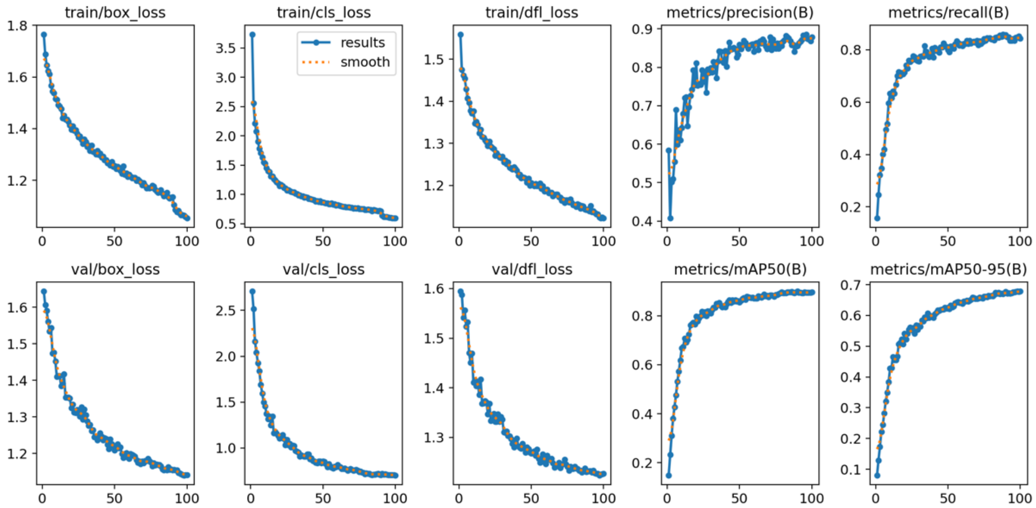


Figure 10. Loss metric graph and detection performance of YOLOv8 model in 100 epoch scenario

Figure 11. Batch validation data visualization with bounding boxes and class labels from YOLOv8 training epoch 100

Model Training with 200 Epochs

After 200 training epochs, the YOLOv8 model demonstrated good convergence and increasingly stable performance. This was demonstrated by a decrease in the training loss and validation loss values for all components: box loss, classification loss, and distribution focal loss. This decrease indicates that the model is increasingly effective in object localization, classification, and bounding box refinement. From the evaluation side, the precision, recall, mAP50, and mAP50-95 values continued to increase until the end of training. In the final epoch, precision was in the range of 0.91–0.92, recall around 0.87–0.88, mAP50 around 0.91–0.92, and mAP50-95 around 0.71–0.72. These results indicate that the model has excellent detection capabilities and stable generalization on the validation data. The graphs of the loss metrics, model detection performance, and image results using YOLOv8 at epoch 200 are presented in Figure 12 and 13.

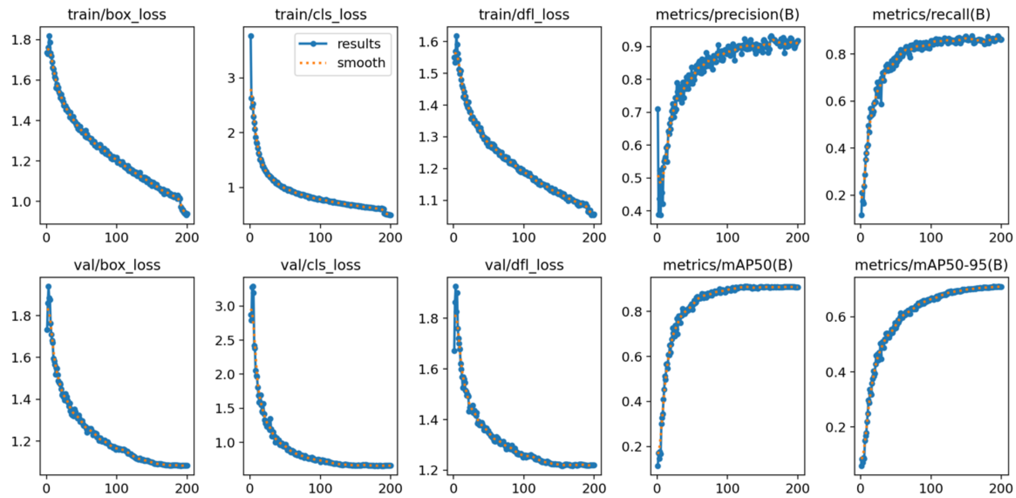


Figure 10. Loss metric graph and detection performance of YOLOv8 model in 200 epoch scenario

Figure 11. Batch validation data visualization with bounding boxes and class labels from YOLOv8 training epoch 200

Comparison of Model Performance Based on Number of Epochs

A comparative analysis of model performance based on the number of epochs aims to evaluate the extent to which additional epochs contribute to improved detection performance. By comparing the evaluation results for each training scenario, this study can identify the most effective number of epochs in producing models with good accuracy and stability.

Table 1. Comparison of model performance based on number of epochs

No	Epoch	Accuracy	Recall	F1 Score	mAP50
1.	30	85.1%	0.806	0.82	0.851
2.	50	87.2%	0.822	0.83	0.872
3.	100	89.5%	0.849	0.85	0.895
4.	200	91.5%	0.87	0.89	0.915

Based on the comparison table, all evaluation metrics show an increasing trend as the number of epochs increases. Accuracy increased from 85.1% at 30 epochs to 87.2% at 50 epochs, then rose again to 89.5% at 100 epochs, and reached 91.5% at 200 epochs. A similar increase was also seen in the recall value, which increased from 0.806 to 0.822, then 0.849, and finally 0.87. The F1-score value also increased consistently from 0.82 at 30 epochs to 0.83 at 50 epochs, 0.85 at 100 epochs, and 0.89 at 200 epochs. Meanwhile, the mAP50 value showed an increase from 0.851 to 0.872, then 0.895, and reached 0.915 at 200 epochs of training. These results indicate that the longer the training process, the better the model's ability to learn data patterns and improve object detection quality. Based on all compared metrics, the 200-epoch scenario performed best and can be considered the most optimal training configuration in this study.

Integration of Detection Model on User Interface Dashboard

After the YOLOv8 model was successfully trained and its performance evaluated, the next step was to integrate it into a prototype smart agriculture monitoring system. This integration aimed to validate the model's application in real-world scenarios, where the algorithm's inference results were visualized through a user interface for easy understanding by end users.

A visualization of the integration results can be seen in Figure 13. This dashboard is designed to monitor crop conditions in real time by combining two crucial pieces of information: pest detection results (computer vision) and environmental data (Internet of Things/IoT). In the "Pest Detection" module, the system provides a visual indicator of the presence of aphids in the monitored area. The indicator changes to "Detected" when the YOLOv8 model identifies pests in the input image, and displays "Not Detected" if the area is safe.

In addition to object detection information, the dashboard also synchronously displays parameters from environmental sensors that influence the pest life cycle and plant growth. These parameters include room temperature (30°C), room humidity (80%), water temperature (26°C), and nutrient concentration (1150 ppm). This environmental data is obtained through the integration of continuously operating physical sensors. Through this comprehensive dashboard, users not only receive early warnings of aphid infestations but also can accurately monitor plant ecology, facilitating faster and more effective decision-making in crop protection management.

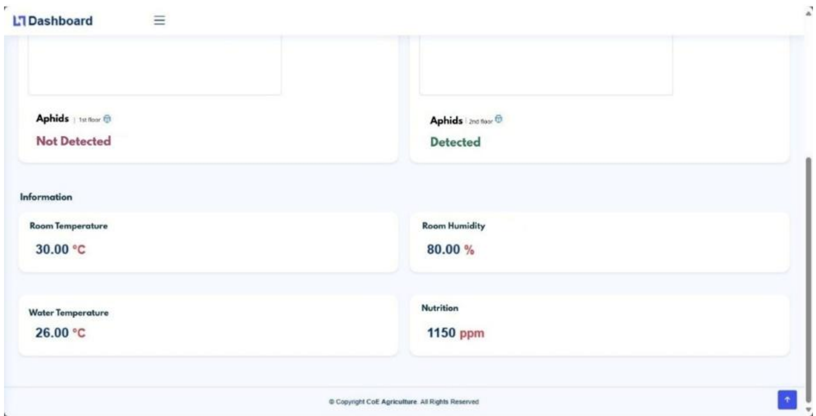


Figure 12. Dashboard interface displaying aphid detection results generated by the YOLOv8 algorithm.

.

4. CONCLUSION

Based on the results of the testing and evaluation conducted, it can be concluded that increasing the number of training iterations (epochs) has a significant positive impact on improving the performance of the YOLOv8 model in detecting aphids. Comparative evaluations of scenarios from 30 to 200 epochs consistently showed an upward trend in all key metrics, with the 200 epoch scenario proving to be the most optimal configuration. At this point, the model achieved its best convergence rate with an accuracy of 91.5%, a recall of 0.87, an F1-score of 0.89, and an mAP50 value of 0.915. Furthermore, this mature detection model was successfully implemented into a prototype interface for a smart agriculture dashboard. The dashboard effectively synchronizes computer vision-based pest detection results with ecological monitoring parameters from IoT sensors, such as temperature and humidity, in real time. This comprehensive integration not only validates the algorithm's reliability in real-world scenarios but also successfully creates a reliable early warning system to facilitate faster and more efficient decision-making in crop protection management.

ACKNOWLEDGEMENTS

The authors would like to thank Universitas Teknokrat Indonesia for the financial support and facilities provided, which made this research possible. This support was instrumental in conducting the research, collecting data, and compiling the results.

CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Styawati: Conceptualization, Methodology, Writing. **Heni Sulistiani**: Implementation & Validation. **Ajeng Savitri Puspaningrum**: Writing - review & editing. **Debby Alita**: Implementation & Validation**. S. Samsugi**: Software. **Vanisa Adelia Putri**: Writing & editing.

DECLARATION OF COMPETING INTERESTS

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

DATA AVAILABILITY

Data will be made available on request. The data that support the findings of this study are available from the corresponding author upon reasonable request. Please contact: styawati@teknokrat.ac.id

REFERENCES

[1] G. Singh and N. K. Joshi, "Insect Pests of Wheat in North India: A Comprehensive Review of Their Bio-Ecology and Integrated Management Strategies," *Agriculture*, 2025.

[2] M. A. Roonjha, R. Roonjho, M. Ali, M. Anas, H. Khalid, and A. Jan, "Aphid-Transmitted Plant Viruses: Epidemiology and Integrated Vector Management," 2025.

[3] M. T. Mallick, S. Banerjee, N. Thakur, H. N. Saha, and A. Chakrabarti, "Evaluation of State-of-the-Art Deep Learning Techniques for Plant Disease and Pest Detection," *CMC*, 2025.

[4] Y. Kaya and E. Gürsoy, "A review of deep learning architectures for plant disease detection," *Turkish J. Biol.*, vol. 49, no. SI-5, pp. 459–497, 2025.

[5] M. Shoaib, A. Sadeghi-Niaraki, F. Ali, I. Hussain, and S. Khalid, "Leveraging deep learning for plant disease and pest detection: a comprehensive review and future directions," *Front. Plant Sci.*, 2025.

[6] S. et al. Wang, "Advances in Deep Learning Applications for Plant Disease and Pest Detection: A Review," *Remote Sens.*, 2025.

[7] C. Yang, Y. Wang, L. Yun, H. Wang, Y. Han, and Z. Chen, "IP-YOLOv8: A Multi-Scale Pest Detection Algorithm for Field-Scale Applications," *Horticulturae*, vol. 11, no. 9, 2025.

[8] M. L. Ali and Z. Zhang, "The YOLO Framework: A Comprehensive Review of Evolution, Applications, and Benchmarks in Object Detection," *Computers*, 2024.

[9] B. et al. Nugroho, "Implementation of Sparrow Pest Detection Using YOLOv8 Method on Raspberry Pi and Google Coral USB Accelerator," *Sebatik*, vol. 29, no. 1, pp. 129–135, 2025.

[10] Z. et al. Zhang, "GVC-YOLO: A Lightweight Real-Time Detection Method for Cotton Aphid-Damaged Leaves Based on Edge Computing," *Remote Sens.*, vol. 16, no. 16, 2024.

[11] X. Li, L. Wang, H. Miao, and S. Zhang, "Aphid Recognition and Counting Based on an Improved YOLOv5 Algorithm in a Climate Chamber Environment," *Insects*, vol. 14, no. 11, 2023.

[12] Y. et al. Zheng, "Aphid-YOLO: A Lightweight Detection Model for Real-Time Identification and Counting of Aphids in Complex Field Environments," *IEEE Trans. AgriFood Electron.*, 2025.

[13] X. Cen, S. Lu, and T. Qian, "YOLO-LCE: A Lightweight YOLOv8 Model for Agricultural Pest Detection," *Agronomy*, vol. 15, no. 9, 2025.

[14] L. et al. Zhang, "YOLOv8 forestry pest recognition based on improved re-parametric convolution," *Front. Plant Sci.*, vol. 16, 2025.

[15] L. Zhou, Y. Zhong, L. Han, Y. Xie, and M. Wan, "Global, regional, and national trends in the burden of melanoma and non-melanoma skin cancer: insights from the global burden of disease study 1990–2021," *Sci. Rep.*, vol. 15, no. 1, p. 5996, 2025.

[16] J. et al. Ryu, "YOLOv8 with Post-Processing for Small Object Detection Enhancement," *Appl. Sci.*, vol. 15, no. 13, 2025.

[17] J. et al. Li, "SRE-YOLOv8: An Improved UAV Object Detection Model Utilizing Swin Transformer and RE-FPN," *Sensors*, vol. 24, no. 12, 2024.

[18] T. M. et al. Omoniyi, "The Effect of Data Augmentation on Performance of Custom and Pre-Trained CNN Models for Crack

Detection," *Appl. Sci.*, vol. 15, no. 22, 2025.
[19] J. Suto, "Using Data Augmentation to Improve the Generalization Capability of an Object Detector on Remote-Sensed Insect Trap Images," *Sensors*, vol. 24, no. 14, 2024.
[20] Y. et al. Zhong, "A Survey of Data Augmentation in Domain Generalization," *Neural Process. Lett.*, vol. 57, no. 2, 2025.
[21] K. et al. Alomar, "Data Augmentation in Classification and Segmentation: A Survey and New Strategies," *J. Imaging*, vol. 9, no. 2, 2023.
[22] F. Hermens, "Automatic object detection for behavioural research using YOLOv8," *Behav. Res. Methods*, vol. 56, no. 7, pp. 7307–7330, 2024.
[23] H. et al. Lou, "DC-YOLOv8: Small-Size Object Detection Algorithm Based on Camera Sensor," *Electronics*, vol. 12, no. 10, 2023.
[24] G. et al. Yue, "GLU-YOLOv8: An Improved Pest and Disease Target Detection Algorithm Based on YOLOv8," *Forests*, vol. 15, no. 9, 2024.
[25] M. O. Ojo and A. Zahid, "Improving Deep Learning Classifiers Performance via Preprocessing and Class Imbalance Approaches in a Plant Disease Detection Pipeline," *Agronomy*, vol. 13, no. 3, 2023.
[26] P. et al. Batz, "From identification to forecasting: the potential of image recognition and artificial intelligence for aphid pest monitoring," *Front. Plant Sci.*, 2023.
[27] K. et al. Joshi, "AgriCure: A web application based layered augmentation-enhanced YOLOv8 for disease and nutrient deficiency detection in bitter gourd leaves," *Curr. Plant Biol.*, vol. 43, 2025.
[28] W. et al. Xu, "RepGhostConv-Subpixel Fusion-Cascading Attention-Detection (RSCDet): A novel lightweight model for real-time aphid detection," *Comput. Electron. Agric.*, vol. 237, 2025.
[29] Y. et al. Shen, "Optimization of Improved YOLOv8 for Precision Tomato Leaf Disease Detection in Sustainable Agriculture," *Sensors*, vol. 25, no. 5, 2025.
[30] J. et al. Zhang, "Accuracy--Efficiency Trade-Off: Optimizing YOLOv8 for Structural Crack Detection," *Sensors*, vol. 25, no. 13, 2025.
[31] Y. et al. Zhu, "An improved YOLOv8n model for in-field detection of pests and diseases in pakchoi," *Front. Plant Sci.*, vol. 16, 2026.

.