Journal of Deep Learning, Computer Vision and Digital Image Processing
E-ISSN: 2986-8939 P-ISSN: 2986-8920
Vol. 4, No. 2, June 2026
DOI. https://doi.org/10.61255/decoding.v4i2.1447

YOLOv8-Based IoT System for Oil Palm Harvest Readiness Identification through Loose Fruit Detection with Real-Time Web Monitoring

(Wira Jhohan Simatupang
Institut Teknologi Sawit Indonesia, Medan, Indonesia)

(Andi Prayogi✉
Institut Teknologi Sawit Indonesia, Medan, Indonesia)

(Muhammad Akbar Syahbana Pane
Institut Teknologi Sawit Indonesia, Medan, Indonesia)

OPEN ACCESS

ARTICLE HISTORY
Received: May 1, 2026
Revised: June 17, 2026
Accepted: June 30, 2026

KEYWORDS
ESP32-CAM;
Harvest Readiness;
Internet of Things (IoT);
Loose Fruit Detection;
YOLOv8

ABSTRACT

Purpose - This study develops and evaluates an Internet of Things (IoT)-based monitoring system for automatically identifying oil palm harvest readiness through loose fruit detection using the YOLOv8 algorithm. The system addresses the subjectivity, labor intensity, and inefficiency of conventional manual observation in large-scale plantations.

Methods - An experimental design was applied using 2,000 images of oil palm loose fruits collected from the Indonesian Institute of Palm Oil Technology, North Sumatra. Images were captured using an ESP32-CAM, annotated through Roboflow, and used to train the YOLOv8x model for 120 epochs at a resolution of 640 × 640 pixels.

Findings - The prototype performed image acquisition, loose fruit detection, and dashboard-based monitoring in near real time. On the independent testing dataset, the model achieved a precision of 0.933, recall of 0.954, F1-score of 0.944, mAP@0.5 of 0.943, and mAP@0.5:0.95 of 0.495. These results demonstrate the system's feasibility as a prototype for supporting harvest-readiness monitoring, although broader field validation is still required.

Research Implications - The dataset was obtained from a single plantation site and may not represent highly variable environmental conditions. Future studies should use larger and more diverse datasets and assess cloud-based deployment for improved scalability.

Originality - This study integrates ESP32-CAM, YOLOv8x, IoT communication, and web-based monitoring into a prototype architecture for data-driven oil palm harvest-readiness assessment through loose fruit detection.

Correspondence Author: ✉ andiprayogi@itsi.ac.id

To cite this article : Simatupang, W. J., Prayogi, A., & Pane, M. A. S. (2026). YOLOv8-based IoT system for oil palm harvest readiness identification through loose fruit detection with real-time web monitoring. Journal of Deep Learning, Computer Vision and Digital Image Processing, 4(2), 236–249. https://doi.org/10.61255/decoding.v4i2.1447

INTRODUCTION

Oil palm (*Elaeis guineensis*) is a plantation commodity that plays a strategic role in Indonesia's economy. As the world's largest palm oil producer, Indonesia relies on the oil palm plantation sector to generate foreign exchange, provide employment, and drive the growth of the vegetable oil-based industry. Rising global demand for palm oil compels plantation companies to continuously enhance productivity, operational efficiency, and harvest quality through the adoption of modern technology. Consequently, effective, technology-driven plantation management has become a crucial factor in maintaining the competitiveness of Indonesia's palm oil industry [1], [2]. Harvesting is one of the most critical stages in the palm oil production process. The timing of the harvest significantly influences oil yield and the quality of the resulting Crude Palm Oil (CPO). Fresh Fruit Bunches (FFB) harvested before reaching optimal maturity have low oil content, whereas delayed harvesting can lead to increased yield losses due to fruit detaching from the bunches. Therefore, accurately identifying fruit maturity levels is essential for supporting productivity and efficiency in the harvesting process [3], [4].

In field operations, a primary indicator used to determine harvest readiness is the number of loose fruits (*brondolan)* found in the cleared circle around the tree base. These loose fruits are oil palm fruits that have naturally detached from the bunch due to physiological ripening. A higher count of loose fruits in the circle indicates a more advanced stage of bunch maturity, signaling that the bunch is ready for harvest. Consequently, the loose fruit count serves as a practical parameter widely used by harvesters to determine the optimal time for harvesting [3], [4]. However, in most plantations, the process of identifying harvest readiness is still carried out manually through visual observation by field workers. This method has several drawbacks: it relies heavily on worker experience, is subjective, and is prone to errors caused by lighting conditions, weather, or surrounding vegetation. Furthermore, the vast size of plantation areas makes comprehensive and continuous monitoring impossible, potentially leading to delays or inaccuracies in determining harvest timing.

Another common issue is yield loss resulting from loose fruits that go undetected or are not collected promptly. Loose fruits have a relatively high oil content, so any fruit left in the field represents an economic loss. The manual process of searching for and collecting these fruits is time-consuming and labor intensive. If this situation persists, the cumulative loss of loose fruits can significantly impact total palm oil production [3], [4]. Various studies have been conducted regarding the application of the Internet of Things (IoT), computer vision, and deep learning in the agriculture and plantation sectors. IoT research generally focuses on developing environmental monitoring systems, crop surveillance, and automating cultivation processes through the use of sensors, cameras, and internet-based communication. Research findings indicate that IoT systems enhance monitoring efficiency while providing real-time information [5], [6].

In the field of computer vision, the YOLO algorithm has emerged as one of the most widely used object detection methods due to its ability to identify objects rapidly without compromising accuracy. The latest version, YOLOv8, offers improved performance through a lighter network architecture, more efficient training processes, and enhanced capabilities for detecting small objects. These characteristics make YOLOv8 highly suitable for camera-based systems requiring real-time processing [7], [8]. Research on oil palm fruit ripeness detection using YOLOv8 has also been extensive. Most studies utilize images of fresh fruit bunches to classify ripeness levels based on color and visual characteristics. These studies demonstrate that YOLOv8 achieves high precision, recall, and mean Average Precision (mAP) scores, making it a viable candidate for implementation in automated detection systems [9].

Furthermore, some studies have integrated object detection results with web dashboards, enabling users to monitor operations online. This approach has proven effective in improving information accessibility and supporting more efficient plantation surveillance [10], [11]. While extensive research has been conducted on oil palm fruit maturity detection, most studies still focus on classifying the maturity level of fresh fruit bunches based on visual characteristics. Research specifically utilizing the detection of detached fruitlets as an indicator of harvest readiness remains relatively limited. Furthermore, most studies concentrate solely on developing object detection models without integrating them into an Internet of Things (IoT) system capable of automated, real-time monitoring. The use of low-cost devices like the ESP32-CAM for image acquisition linked with the YOLOv8 algorithm and a web dashboard is also rarely encountered [12], [13].

Another limitation is the scarcity of research developing automated alarm systems based on the number of detected fruitlets to inform harvesting decisions. Such a system has the potential to enhance monitoring efficiency and reduce reliance on manual observation. Addressing these research gaps requires a system that integrates IoT technology, computer vision, and deep learning into a single, automated platform. Fruitlet detection was selected because it aligns with the indicators field workers use to determine harvest readiness; consequently, the research findings are expected to have a higher likelihood of practical implementation in actual plantation settings [3], [14]. Using the ESP32-CAM for image acquisition offers an economical solution that is easy to deploy in the field. Meanwhile, the YOLOv8 algorithm was chosen for its optimal balance between processing speed and accuracy. Integration with a web dashboard allows detection results to be displayed automatically through a near-real-time prototype interface, while an automated alarm system assists workers and plantation managers in determining harvest timing more quickly, objectively, and efficiently [12], [15].

Thus, this research is expected to contribute to the development of an Internet of Things (IoT)-based smart plantation system capable of increasing productivity, reducing harvest losses, and supporting digital transformation in the oil palm plantation sector. This research aims to design and build an IoT system capable of detecting the presence of loose oil palm fruits an indicator of harvest readiness using an ESP32-CAM camera and the YOLOv8 algorithm. The developed system was designed to perform automatic loose fruit detection, display monitoring results through a web dashboard, and provide detection-count information that can support the development of a rule-based harvest-readiness alarm. Based on these objectives, the proposed research hypotheses are as follows : (1) The YOLOv8 algorithm is capable of detecting loose oil palm fruit objects with a high level of accuracy based on images captured by the ESP32-CAM. (2) The integration of the ESP32-CAM, the YOLOv8 algorithm, IoT technology, and a web dashboard can support a near-real-time harvest readiness monitoring prototype. (3) A rule-based alarm mechanism using the detected loose fruit count has the potential to support more objective harvest-readiness decisions, provided that the decision threshold is calibrated and validated using agronomic or field-based criteria.

METHOD

Research Design

This study employed an experimental research design to develop and evaluate an Internet of Things (IoT)-based oil palm loose fruit detection system using the YOLOv8x object detection algorithm. The design was structured as a sequential development process consisting of device preparation, field image acquisition, dataset development, model construction, system integration, and prototype testing. The ESP32-CAM was installed in an oil palm plantation environment to capture images of loose fruits under naturally occurring conditions. The collected images were then prepared and annotated to support the development of the object detection model. After the model development stage, YOLOv8x was integrated with the ESP32-CAM, server, and web-based monitoring platform to form a complete IoT prototype. This experimental design enabled the researchers to systematically examine whether the proposed system could perform image acquisition, loose fruit detection, data transmission, and monitoring within a near-real-time plantation setting.

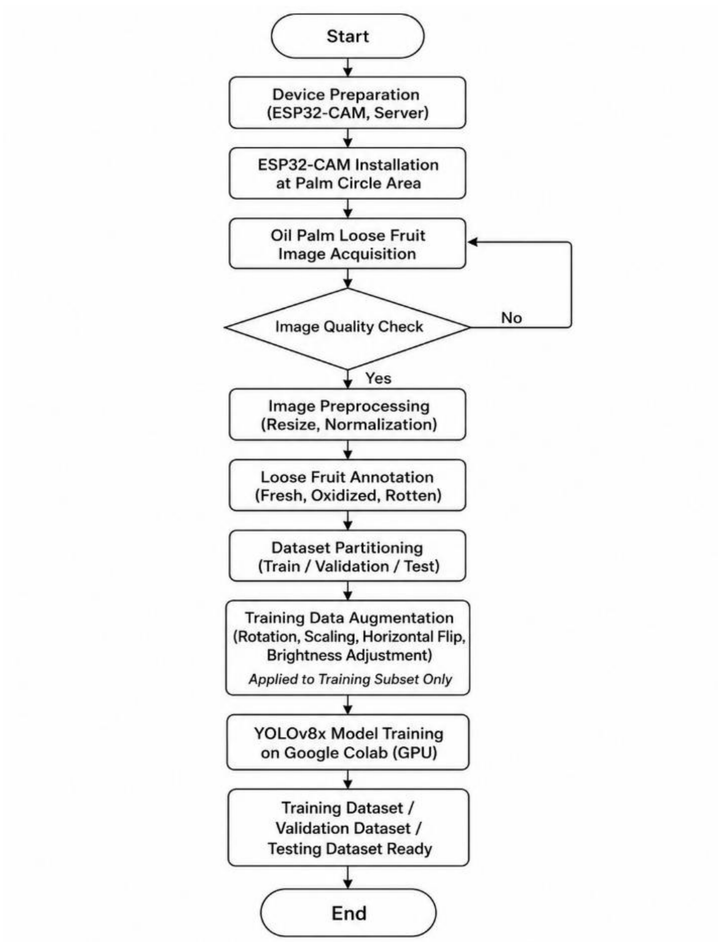

Figure 1. Flowchart for Oil Palm Loose Fruit Dataset Collection

Participant and Population

As this study did not involve human participants, the research population consisted of digital images of palm oil loose fruits collected from the plantation area of the Indonesian Institute of Palm Oil Technology (ITSI), North Sumatra, Indonesia. A total of 2,000 images were acquired using an ESP32-CAM module. Purposive sampling was applied to ensure that the collected images represented various environmental conditions, including different lighting levels, camera angles, and loose fruit distributions. Images with poor quality, excessive blur, or inadequate visibility were excluded from the dataset.

Instrumentation

The primary instruments used in this study were an ESP32-CAM module for image acquisition, the Roboflow platform for image annotation, and the YOLOv8x framework for object detection model development. Each loose fruit object was manually annotated using the bounding box technique and categorized into three visual-condition classes: fresh, oxidized, and rotten. These classes represent different visual conditions of detached oil palm fruits, while background regions were treated as non-object areas during model evaluation. For harvest-readiness monitoring, detections from the three classes were aggregated as the total loose fruit count.

The original dataset consisted of 2,000 JPEG images with filenames ranging from Brondolan_1.jpg to Brondolan_2000.jpg. To reduce the risk of data leakage, dataset partitioning was performed before augmentation. The original images were first divided into training, validation, and testing subsets using an 80:10:10 ratio, resulting in 1,600 original images for training, 200 images for validation, and 200 images for testing. Data augmentation was then applied only to the training subset using rotation, scaling, horizontal flipping, and brightness adjustment. This procedure increased the training subset from 1,600 to 4,800 images, while the validation and testing subsets remained unaugmented. All images were resized to 640 × 640 pixels to match the YOLOv8 input requirement. The reliability of the dataset was supported through manual verification of annotations, while validity was maintained by collecting images directly from plantation environments.

Procedures

The study was conducted from January to June 2026. The research procedure consisted of five stages. First, image data were collected using the ESP32-CAM installed near the palm tree circle area. Second, image preprocessing and annotation were performed using Roboflow, where loose fruits were labeled into three visual-condition classes: fresh, oxidized, and rotten. Third, the 2,000 original annotated images were divided into training, validation, and testing subsets before augmentation using an 80:10:10 ratio. This split produced 1,600 original training images, 200 validation images, and 200 testing images. Data augmentation was then applied only to the training subset, increasing the number of training images to 4,800. The validation and testing subsets were not augmented and were kept independent from the augmented training data. Fourth, the YOLOv8x model was trained on Google Colaboratory using GPU acceleration with an input image size of 640 × 640 pixels, a batch size of 16, and 120 training epochs. Finally, the best-performing model (best.pt) was deployed into an IoT-based monitoring prototype consisting of an ESP32-CAM, a local processing server, and a web dashboard.

Table 1. Dataset Partitioning and Augmentation Workflow

Subset	Original Images	Augmentation Applied	Final Images Used
Training	1,600	Yes	4,800
Validation	200	No	200
Testing	200	No	200
Total	**2,000**	**Training only**	**5,200**

This workflow ensured that augmented variants of the same original image did not appear across different subsets, thereby reducing the risk of data leakage during model evaluation. During operation, images captured by the ESP32-CAM were transmitted to the server through a Wi-Fi network. The server processed each image using the trained YOLOv8 model and displayed the detection results, including the number of detected loose fruits and harvest recommendations, on the monitoring dashboard.

For the harvest-readiness alarm, the system aggregated detections from the fresh, oxidized, and rotten classes into a total loose fruit count. This count was intended to serve as the input for a rule-based alarm module. However, the current prototype did not embed a fixed agronomic threshold for determining harvest readiness, and no field validation of alarm decisions was conducted in this study. Therefore, the alarm component is treated as a decision-support concept that requires threshold calibration based on plantation practice, expert judgment, or agronomic criteria before operational deployment.

Analysis Plan

Model performance was evaluated using standard object detection metrics, including Precision, Recall, F1-score, mean Average Precision at an IoU threshold of 0.5 (mAP@0.5), and mean Average Precision across IoU thresholds from 0.5 to 0.95 (mAP@0.5:0.95). The validation subset was used during training to monitor model convergence and select the best-performing model checkpoint (best.pt). Final model performance was then reported using the independent testing subset, which was not used during model training or model selection. Confusion matrices, Precision-Recall curves, and F1-score curves were analyzed by comparing predicted bounding boxes with ground-truth annotations from the independent testing subset. Because the model categorized loose fruits into fresh, oxidized, and rotten classes, evaluation was reported both at the class level and as an aggregated loose-fruit detection result for harvest-readiness monitoring.

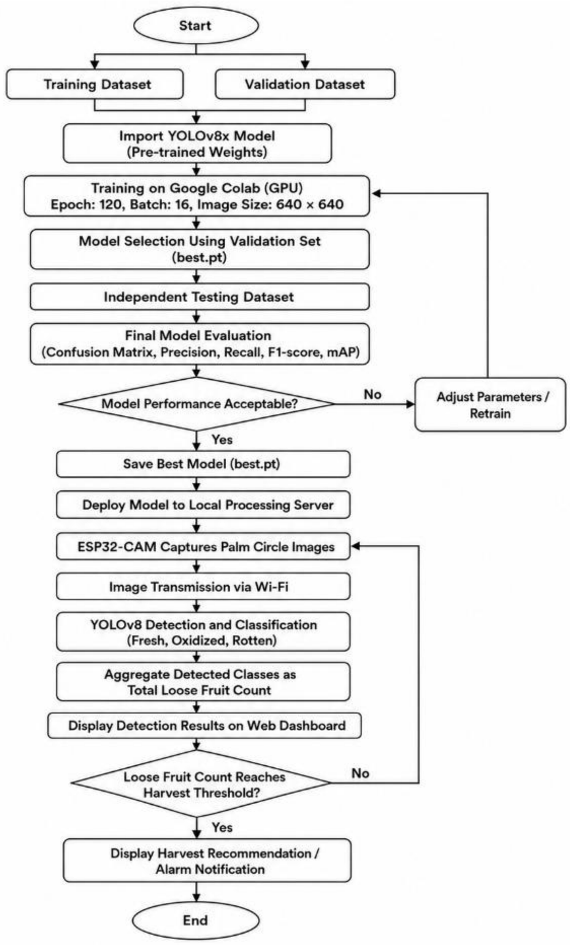

Figure 2. Flowchart of the IoT Detection and Integration System

Scope and Limitations

This study was limited to detecting palm oil loose fruits within the plantation environment where the dataset was collected. The model was trained using 2,000 images obtained from a single geographical location, which may limit its generalizability to plantations with significantly different environmental characteristics. Furthermore, detection performance may be affected by extreme lighting conditions, severe occlusions, or variations in camera positioning. Future studies may expand the dataset size, include multiple plantation locations, and evaluate the system under a wider range of environmental conditions.

RESULTS AND DISCUSSION

The objective of this study was to develop and evaluate an Internet of Things (IoT)-based monitoring system capable of detecting palm oil loose fruits as an indicator of harvest readiness. The proposed system integrates an ESP32-CAM image acquisition device, a YOLOv8 object detection model, a local processing server, and a web-based monitoring dashboard. The results demonstrate that the developed system successfully performed image acquisition, object detection, data transmission, and visualization of detection outcomes in a near-real-time prototype monitoring environment.

The successful integration of these components demonstrates the feasibility of combining IoT technology with deep learning-based computer vision for intelligent agricultural monitoring applications. Throughout prototype testing, the system functioned as intended, enabling communication between the hardware and software components while maintaining detection output consistency. The web-based dashboard provided users with direct access to detection results, allowing remote observation of harvest readiness without the need for direct field inspection. These findings indicate that the proposed architecture can support more efficient plantation management by reducing manual monitoring efforts, improving the quality of decision-making, and providing objective information to support harvesting activities on oil palm plantations.

System Implementation Results

The developed architecture consists of four main components.

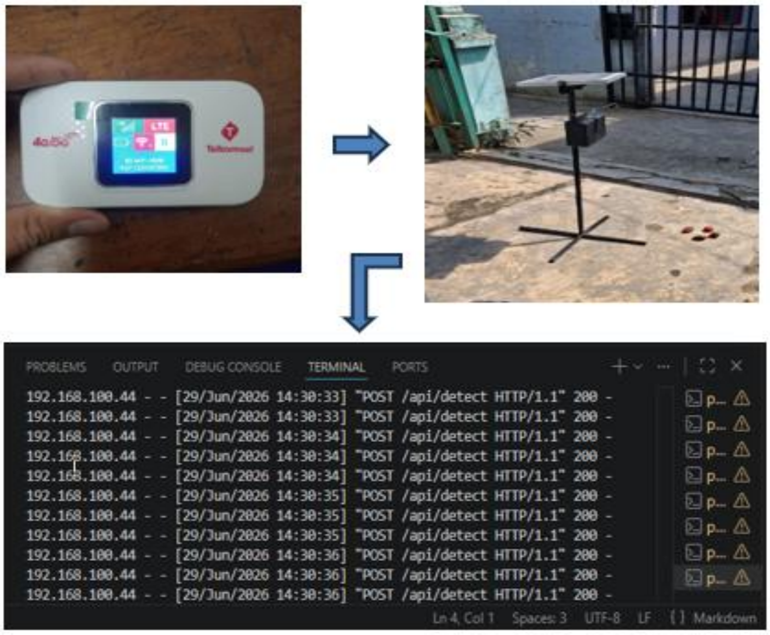

Figure 3. ESP32-CAM Device Connectivity Testing

The ESP32-CAM module functioned as the image acquisition device, periodically capturing images from the palm tree circle area, where loose fruits typically accumulate naturally. The captured images were then transmitted over a wireless network to the local processing server. The server executed the YOLOv8 model to detect loose fruits and calculate the number of detected objects. The detection results were subsequently displayed via a web dashboard, allowing users to remotely monitor plantation conditions.

The implementation process demonstrated successful integration between the hardware and software components. The ESP32-CAM firmware was configured to continuously capture JPEG frames and push them to the local Flask server through an HTTP POST request. No fixed capture interval was defined in the firmware; instead, frames were transmitted repeatedly according to the capture, encoding, and network transmission cycle of the device. The server received each incoming frame, placed the latest frame into an asynchronous queue, and performed YOLOv8x inference on the local processing machine. The measured server-side inference time was recorded as inference_ms and forwarded to the web dashboard through a WebSocket event. However, this study did not record separate timestamps for image capture, HTTP transmission, dashboard rendering delay, or total end-to-end response time. Therefore, the monitoring capability is reported as a near-real-time prototype response rather than a fully benchmarked real-time system.

Connectivity and Image Transmission Performance

Connectivity testing was conducted to verify the ESP32-CAM's ability to communicate with the local server over a Wi-Fi network.

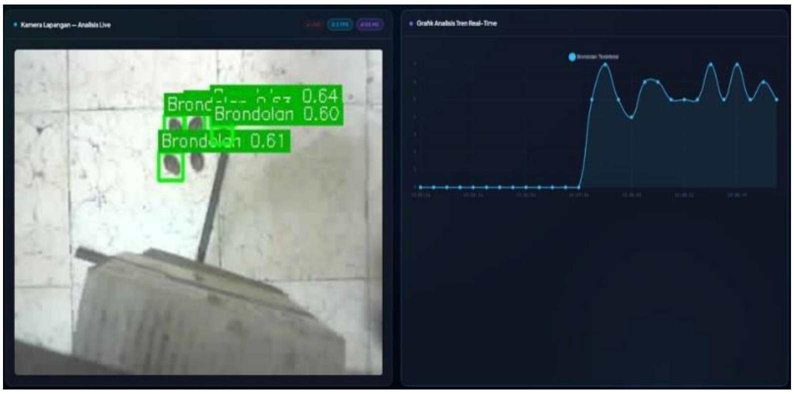

Figure 4. Testing of image transmission captured by the ESP32-CAM

The device successfully obtained an IP address and maintained stable communication throughout the testing period. No significant disruptions were observed during data transmission. Image transmission testing further confirmed that images captured by the ESP32-CAM could be correctly received by the server. A visual comparison between the transmitted and received images showed no meaningful degradation in image quality. In addition, no indication of image corruption or data loss was found during transmission. These results indicate that the communication pipeline between the ESP32-CAM, local server, and dashboard was functionally verified during prototype testing. However, a separate quantitative reliability benchmark, including transmission success rate, failed upload count, corrupted image rate, and total operating duration, was not conducted in this study.

YOLOv8 Detection Performance

The YOLOv8x model demonstrated strong performance in detecting loose fruits within plantation images. Using the independent testing subset, detected loose fruits were localized using bounding boxes and classified into three visual-condition classes: fresh, oxidized, and rotten. Visual inspection showed that the model was able to distinguish loose fruits from complex plantation backgrounds, including soil surfaces, fallen leaves, and surrounding plant materials. For harvest-readiness monitoring, detections from the three visual-condition classes were aggregated as the total loose fruit count.

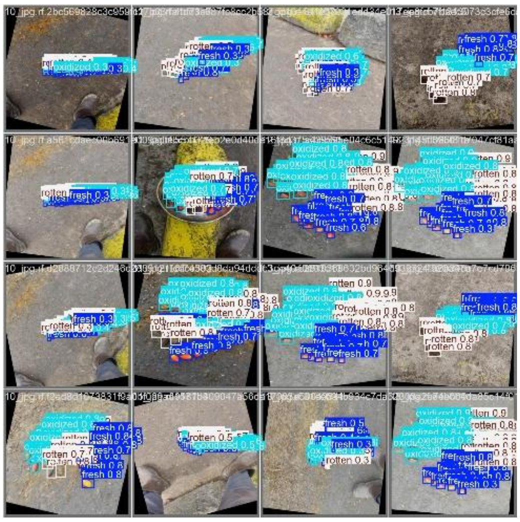

Figure 5. YOLOv8 Loose Fruit Detection Results with Multi-Class Classification

The detection and class-assignment results were further examined using the confusion matrix. As shown in Figure 6, most predictions were concentrated along the main diagonal, indicating that the model generally assigned detected loose fruits to the correct visual-condition classes. The oxidized class produced the highest number of correct predictions, followed by the fresh and rotten classes. The remaining errors were mainly associated with visually similar fruit conditions and background regions that resembled loose fruit objects.

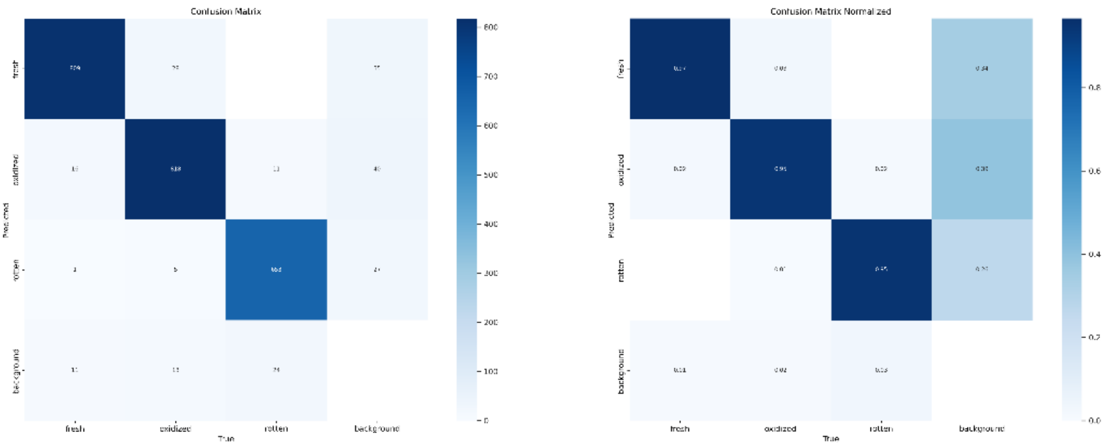

Figure 6. Confusion Matrix and Normalized Confusion Matrix of YOLOv8 Loose Fruit Condition Classification

The detailed class-level classification performance is summarized in Table 2. The table presents the number of true positives (TP), false positives (FP), false negatives (FN), precision, recall, and F1-score for each loose fruit condition class. Precision, recall, and F1-score were calculated from the reported TP, FP, and FN values to ensure consistency between the confusion matrix summary and the final performance metrics.

Table 2. Class-Level Confusion Matrix Summary

Class	TP	FP	FN	Precision	Recall	F1-score
Fresh	809	63	28	0.928	0.967	0.947
Oxidized	818	67	46	0.924	0.947	0.935
Rotten	653	33	35	0.952	0.949	0.951
Micro-average	**2,280**	**163**	**109**	**0.933**	**0.954**	**0.944**

Based on the micro-average values, the model achieved a precision of 0.933, a recall of 0.954, and an F1-score of 0.944. These results indicate that the model maintained a balanced ability to detect loose fruits while minimizing false detections and missed objects across the three condition classes. In addition to class-level evaluation, the detections from fresh, oxidized, and rotten classes were aggregated during system operation to generate the total loose fruit count used for harvest-readiness monitoring.

Training Performance Evaluation

The training process showed a substantial improvement in model performance across successive epochs.

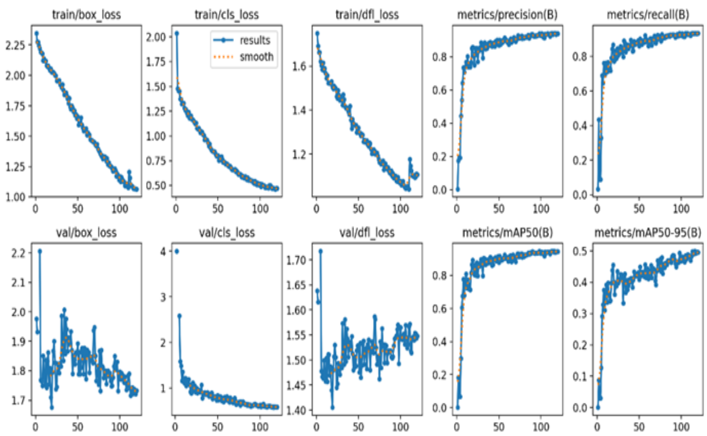

Figure 7. YOLOv8 Model Training Results Graph

The training box loss decreased from approximately 2.3 to 1.1, while the classification loss decreased from approximately 2.0 to 0.5. Similarly, the distribution focal loss (DFL) decreased from approximately 1.7 to 1.1. These reductions indicate that the model progressively learned meaningful feature representations from the training dataset.

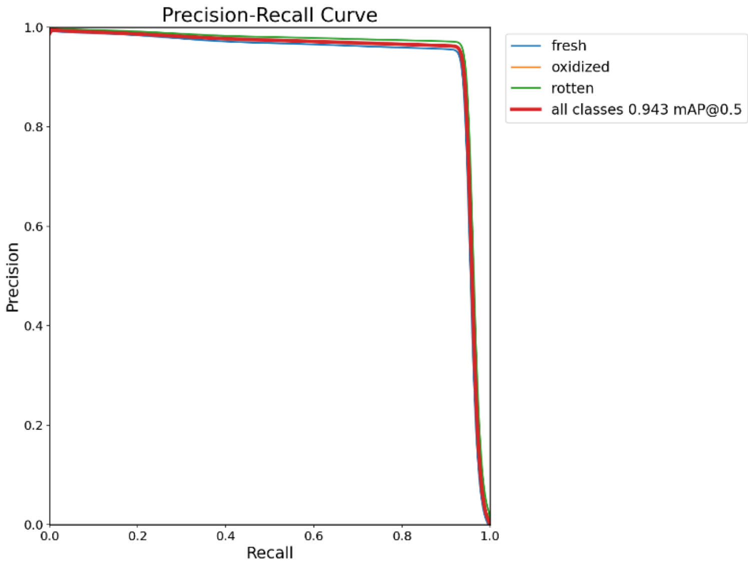

Figure 8. Precision-Recall Curve

Performance metrics also improved consistently throughout training. Precision and recall values increased steadily, exceeding 0.90 by the end of the training process. The mAP@0.5 reached 0.943, while the mAP@0.5:0.95 reached 0.495.

Table 3. Summary of Model Evaluation Metrics

Metric	Value
Precision	0.933
Recall	0.954
F1-score	0.944
mAP@0.5	0.943
mAP@0.5:0.95	0.495

The table above summarizes the final values of all evaluation metrics discussed in this subsection, allowing them to be referenced quickly without needing to consult each curve separately. The evaluation results indicate that the YOLOv8x model achieved strong detection performance across the three loose fruit condition classes. The precision, recall, and F1-score values show that the model was able to detect and classify loose fruits with relatively balanced performance. The mAP@0.5 value of 0.943 indicates strong detection performance at the standard IoU threshold. However, the lower mAP@0.5:0.95 value of 0.495 suggests that bounding-box localization became less stable under stricter IoU thresholds. This pattern is understandable because loose fruits are relatively small, irregularly scattered, and may partially overlap with debris, shadows, or vegetation in plantation environments.

Precision Curve

The Precision-Recall curve further confirmed the model's detection capability across the three loose fruit condition classes. Overall, the model achieved an mAP@0.5 of 0.943, indicating strong detection performance at the standard IoU threshold. The curve also shows that model confidence influenced the balance between precision and recall, where stricter confidence thresholds reduced false detections but could also increase the possibility of missed objects.

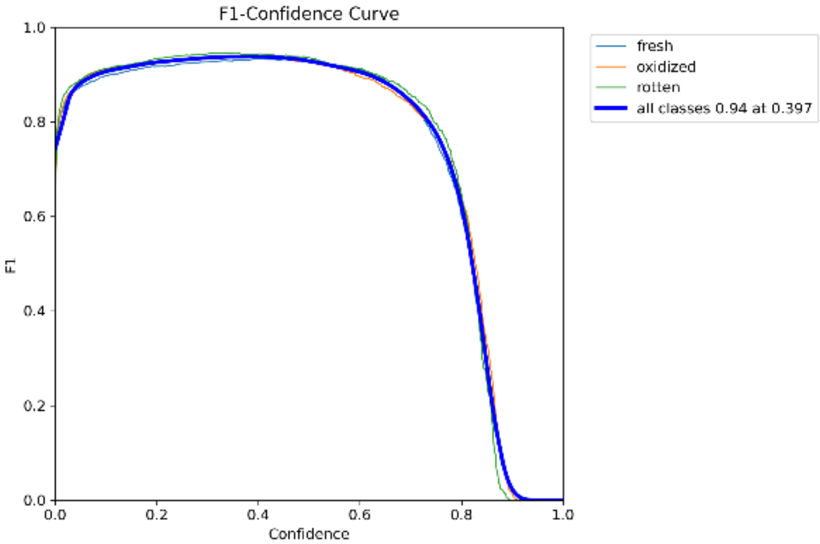

Figure 9. F1-Score Curve

The F1-score curve reached a maximum value of approximately 0.94 at a confidence threshold of 0.397. This result reflects an optimal balance between precision and recall, suggesting that the model is able to simultaneously minimize false detections and missed objects.

Dashboard Monitoring Results

The web dashboard successfully displayed detection results in a near-real-time monitoring interface. Information presented on the dashboard included annotated images, detected object counts, timestamps, and harvest status indicators. The harvest status indicator should be interpreted as a prototype display derived from the detection-count output rather than as a validated agronomic alarm. The present study did not evaluate alarm accuracy, false alarm rate, or missed alarm rate because the harvest-readiness threshold was not calibrated against field-based harvesting decisions. During prototype testing, the displayed information corresponded to the output generated by the detection model.

Although the system was able to display detection outputs automatically through the dashboard, the present study did not perform a detailed latency decomposition covering image capture interval, Wi-Fi transmission delay, model inference time, dashboard refresh delay, and total end-to-end response time. Consequently, the term near-real-time in this study refers to the prototype's ability to update detection results automatically after image acquisition and server-side processing, rather than to a formally benchmarked real-time performance standard. The dashboard provided users with a centralized platform for monitoring plantation conditions without requiring direct field inspection. This functionality demonstrates the practical feasibility of integrating computer vision and IoT technologies into plantation management operations.

Discussion

The results indicate that the integration of YOLOv8 and IoT technologies provides an effective solution for automated harvest readiness monitoring on oil palm plantations. Traditionally, harvest decisions rely heavily on manual observations carried out by field workers. Such approaches tend to be time-consuming, labor-intensive, and susceptible to subjective judgment. The proposed system addresses these limitations by providing objective, data driven assessments based on loose fruit detection. One of the primary strengths of the developed system lies in the object detection capability of YOLOv8. The precision, recall, and mAP values obtained during evaluation indicate that the model was able to identify loose fruits within the visual conditions represented in the testing subset. This capability is particularly important, as plantation environments present numerous challenges, including variable illumination, background clutter, shadows, and object occlusion.

To position the findings of this study, the model performance was compared contextually with previous YOLOv8-based studies related to oil palm object detection. Farhan et al. [14] reported an mAP@0.5 of

0.98 for oil palm tree detection and counting using UAV imagery and YOLOv8n deployed on an edge device. In comparison, the present study achieved an mAP@0.5 of 0.943, with a precision of 0.933 and a recall of 0.954, for loose fruit detection using images acquired from an ESP32-CAM placed near the palm tree circle area. This comparison should be interpreted cautiously because the target objects and acquisition settings are different. Oil palm trees captured from a UAV perspective are larger and more structurally consistent, whereas loose fruits are smaller, irregularly scattered, and often affected by soil background, shadows, vegetation, and partial occlusion.

A contextual comparison can also be made with Marchiningrum et al. [9], who evaluated YOLOv8 variants for oil palm fruit ripeness classification using three maturity categories. Their study focused on fresh fruit bunch ripeness classification, while the present study focused on detecting detached loose fruits and classifying them into three visual-condition classes: fresh, oxidized, and rotten. Although the evaluation results are within the performance range reported in related YOLOv8-based oil palm studies, direct superiority cannot be claimed because the datasets, target objects, class definitions, image acquisition conditions, and deployment settings differ. The main contribution of the present study lies in integrating YOLOv8x-based loose fruit detection with ESP32-CAM image acquisition, server-side inference, and a web-based monitoring dashboard for harvest-readiness observation.

The observed reduction in training losses throughout the learning process suggests that the model successfully learned discriminative features associated with loose fruits. Furthermore, the absence of major fluctuations in the validation metrics indicates stable convergence with limited overfitting. These findings confirm that the dataset preparation strategy, including annotation quality and augmentation techniques, contributed significantly to the model's final performance. Another important finding relates to the effectiveness of the IoT integration. The ESP32-CAM module successfully functioned as a low-cost image acquisition platform capable of reliably transmitting images to the processing server. This demonstrates the practicality of deploying affordable hardware solutions in agricultural environments. Compared with more expensive industrial imaging systems, the proposed configuration offers a cost-efficient alternative suitable for small and medium-sized plantations.

The use of YOLOv8x should be interpreted within the server-side inference architecture adopted in this prototype. The ESP32-CAM was used only as a low-cost image acquisition and transmission device, while object detection was performed on a local processing server. YOLOv8x was selected to prioritize detection accuracy for small, scattered, and visually variable loose fruit objects. However, this choice increases computational requirements compared with lighter YOLOv8 variants such as YOLOv8n or YOLOv8s. As a result, the proposed configuration may require a more capable local server, may increase deployment cost, and may limit scalability if many ESP32-CAM nodes are operated simultaneously. Future implementation should therefore evaluate lightweight YOLOv8 variants or model optimization techniques for edge deployment when lower cost, lower latency, and wider scalability are required.

The web dashboard also played a crucial role in enhancing the system's usability. By presenting detection results in a visual and accessible format, the dashboard enables plantation managers to remotely monitor field conditions. This capability aligns with the principles of precision agriculture, which emphasize data-driven decision-making and continuous monitoring. The high recall value achieved by the model is particularly valuable for harvest readiness assessment. Missed loose fruits could potentially lead to incorrect harvest recommendations. Maintaining high recall therefore ensures that most relevant objects are detected and included in the decision-making process. At the same time, the high precision value minimizes false alarms that could otherwise reduce user trust in the system. Overall, the results demonstrate that combining computer vision with IoT infrastructure can significantly improve operational efficiency in oil palm plantation management. The developed system successfully transforms visual observations into actionable information, supporting more consistent and objective harvest decisions.

CONCLUSION

This study developed and evaluated an Internet of Things (IoT)-based prototype monitoring system for oil palm harvest-readiness assessment using YOLOv8x-based loose fruit detection. The proposed system integrated an ESP32-CAM image acquisition device, a wireless communication network, a local processing server, and a web-based dashboard to support automated detection-count visualization in a near-real-time prototype environment. Model evaluation on the independent testing subset showed that the YOLOv8x model achieved a precision of 0.933, a recall of 0.954, an F1-score of 0.944, an mAP@0.5 of 0.943, and an mAP@0.5:0.95 of 0.495. These results indicate that the model was able to detect and classify loose fruits into fresh, oxidized, and rotten visual-condition classes with balanced performance. The lower mAP@0.5:0.95 value also shows that bounding-box localization became less stable under stricter IoU thresholds, which is relevant because loose fruits are small, scattered, and often affected by background clutter or partial occlusion.

The system implementation demonstrated the feasibility of integrating low-cost ESP32-CAM image acquisition with server-side YOLOv8x inference and a web-based dashboard. During prototype testing, the dashboard displayed annotated images, detected loose fruit counts, timestamps, and harvest-status indicators derived from detection-count outputs. However, the harvest-readiness alarm should be interpreted as a rule-based decision-support concept rather than a validated agronomic alarm, because the decision threshold was not calibrated against field-based harvesting decisions. Overall, this study demonstrates prototype feasibility for applying computer vision and IoT technologies to support oil palm loose fruit monitoring. The system may assist future development of data-driven harvest-readiness observation, but its operational effectiveness in improving harvest scheduling, reducing loose fruit losses, or increasing CPO production has not yet been confirmed through large-scale field validation.

Future research should evaluate the system across multiple plantation locations, longer operating periods, different lighting conditions, camera angles, object densities, and occlusion levels. Further work should also quantify IoT reliability, measure end-to-end latency, calibrate the harvest-readiness alarm threshold using agronomic or expert-based criteria, and compare YOLOv8x with lighter model variants for more scalable edge or low-cost deployment.

ACKNOWLEDGMENT

The authors would like to express their gratitude to the Institut Teknologi Sawit Indonesia, North Sumatra, Indonesia, for the institutional support, plantation access, and facilities provided during image data collection and prototype testing. The authors also acknowledge the plantation personnel who assisted with the installation of the ESP32-CAM and the implementation of the IoT-based monitoring system.

AUTHOR CONTRIBUTION STATEMENT

WJS conceived the study, provided conceptual guidance, supervised the research, validated the methodology, and reviewed the manuscript. AP collected and annotated the image data, performed data preprocessing, trained and evaluated the YOLOv8x model, developed and integrated the IoT monitoring system, analyzed the results, and wrote the original manuscript. MASP contributed to dataset preparation, hardware installation, field testing, system validation, research documentation, and manuscript review. All authors reviewed and approved the final version of the manuscript. The authors and institutional affiliation are based on the submitted manuscript

AI DISCLOSURE STATEMENT

The authors used ChatGPT, developed by OpenAI, during the preparation of this manuscript for language editing, grammar improvement, and script refinement. After using this tool, the author carefully reviewed, revised, and validated all content to ensure its accuracy, originality, and scientific integrity. The author is solely responsible for the content of this manuscript.

REFERENCES

[1] Badan Pusat Statistik Indonesia, Statistik Indonesia 2025. Jakarta, Indonesia: BPS, 2025. [Online]. Available: https://www.bps.go.id

[2] H. K. Adli, M. A. Remli, K. N. S. Wan Salihin Wong, N. A. Ismail, A. González-Briones, J. M. Corchado, and M. S. Mohamad, "Recent Advancements and Challenges of AIoT Application in Smart Agriculture: A Review," Sensors, vol. 23, no. 7, pp. 1–22, 2023, doi: 10.3390/s23073752.

[3] F. Firnanda, "Manajemen Pemanenan Kelapa Sawit Tanaman Menghasilkan pada Perkebunan Senagan PT. Socfindo Afdeling II," J. Pertan. Agros, vol. 25, no. 3, pp. 2028–2037, 2023.

[4] H. Manurung and E. Wiraguna, "Analisis Efisiensi Metode Panen Manual dan Mekanisasi pada Produksi Kelapa Sawit," Botani: Publ. Ilmu Tanam. Agribisnis, vol. 2, no. 2, pp. 1–11, 2025, doi: 10.62951/botani.v2i2.288.

[5] T. Alahmad, M. Neményi, and A. Nyéki, "Applying IoT Sensors and Big Data to Improve Precision Crop Production: A Review," Agronomy, vol. 13, no. 10, 2023, doi: 10.3390/agronomy13102603.

[6] A. Pawar and S. B. Deosarkar, "IoT-Based Smart Agriculture: An Exhaustive Study," Wireless Netw., vol. 29, no. 6, pp. 2457–2470, 2023, doi: 10.1007/s11276-023-03315-7.

[7] C. M. Badgujar, A. Poulose, and H. Gan, "Agricultural Object Detection with You Only Look Once (YOLO) Algorithm: A Bibliometric and Systematic Literature Review," Comput. Electron. Agric., vol. 223, 2024, doi: 10.1016/j.compag.2024.109090.

[8] A. Reis, J. Kupec, J. Hong, and A. Daoudi, "Real-Time Flying Object Detection with YOLOv8," arXiv preprint arXiv:2305.09972, 2024.

[9] A. U. Marchiningrum et al., "Automatic Detection of Palm Oil Fruit Ripeness Using YOLOv8 for Selection of the Best Detection Model," vol. 6, no. 2, pp. 299–306, 2026.

[10] B. E. Mayowela and J. S. Ally, "Real-Time Monitoring of Parameters Contributing to Soil Quality in Palm Oil Plantation," East African J. Inf. Technol., vol. 6, no. 1, pp. 231–242, 2023, doi: 10.37284/eajit.6.1.1557.

[11] M. N. Prayoga, A. Hamdani, and M. L. Hakim, "Implementasi Sistem Absensi Online Berbasis Web Menggunakan Laravel dan Ngrok sebagai Server," J. Sist. Inf. Univ. Suryadarma, vol. 13, no. 1, pp. 97–107, 2025, doi: 10.35968/jsi.v13i1.1710.

[12] S. A. Arrahma and R. Mukhaiyar, "Pengujian ESP32-CAM Berbasis Mikrokontroler," vol. 4, no. 1, pp. 60–66, 2023.

[13] D. Hercog, T. Lerher, M. Truntič, and O. Težak, "Design and Implementation of ESP32-Based IoT Devices," Sensors, vol. 23, no. 15, 2023, doi: 10.3390/s23156739.

[14] M. Farhan, M. N. Akhtar, and E. A. Bakar, "Efficient Real-Time Palm Oil Tree Detection and Counting Using YOLOv8 Deployed on Edge Devices," J. Umm Al-Qura Univ. Eng. Archit., vol. 16, no. 4, pp. 1293–1308, 2025, doi: 10.1007/s43995-025-00164-7.

[15] K. A. Al-Zihairy and A. E. Abdelkareem, "Enhanced YOLOv8 for Edge Computing: A Deep Learning Approach for Memory Optimization," Iraqi J. Inf. Commun. Technol., vol. 8, no. 1, pp. 23–37, 2025, doi: 10.31987/ijict.8.1.283.