Journal of Computer Networks, Architecture and High Performance Computing
Volume 08, Number 3, July 2026
https://doi.org/ 10.47709/cnahpc.v8i3.8780

Submitted : 2026-06-25
Accepted : 2026-07-06
Published : 2026-07-06

Real-Time Pest Monitoring System for Chili Plants Based on Internet of Things Using Image Sensors and Soil Moisture Sensors

Hari Haran[1], Jose Given Amanro Silaen[2], Muhammad Fakhruddin Alrazi[3], Ertina Sabarita Barus[4], Muhammad Akbar Raihansyah[5]
[1,2,3,4,5)] Universitas Prima Indonesia, Medan 20118, Indonesia
[1)]jayahari508@gmail.com, [2)] josesilaen2003@gmail.com, [3)]sihazinamanya@gmail.com, [4)] ertinasabaritabarus@unprimdn.ac.id, [5)] akbar33949@gmail.com

ABSTRACT

Pest infestation is one of the main causes of declining productivity in chili (Capsicum annuum L.) cultivation, while conventional monitoring still relies on manual visual inspection that is subjective, time-consuming, and prone to delayed detection. This study aims to design and implement an Internet of Things (IoT) based real-time pest monitoring system that integrates an image sensor and a soil moisture sensor on chili plants. The system is built around an ESP32 microcontroller and an ESP32-CAM module that captures leaf images, which are analyzed on a server using the You Only Look Once version 8 (YOLOv8) object detection model, while a capacitive soil moisture sensor monitors the growing-media condition. The detection model was trained on a Pest Detection dataset from Roboflow Universe consisting of 38,449 images across 28 pest classes, using 40 epochs and an input size of 480 pixels. Evaluation on 1,612 test images produced a precision of 0.853, a recall of 0.735, an mAP@0.5 of 0.778, an mAP@0.5:0.95 of 0.606, and an F1-score of approximately 0.79. A dedicated eight-stage image pre-processing pipeline was applied to reduce the domain gap between the high-quality training images and the lower-quality ESP32-CAM production images, and all sensor data were transmitted to the server with an average latency below two seconds. The results show that the integration of visual detection and environmental sensing produces an accurate and responsive early-warning system that is suitable for low-cost smart-farming deployment.
Keywords: *chili plant; ESP32-CAM; Internet of Things; pest detection; YOLOv8*

INTRODUCTION

Chili (Capsicum annuum L.) is a high-value horticultural commodity that plays a crucial role in the stability of the national food industry. Indonesia is recorded as one of the largest chili producers in the world, which makes this commodity economically strategic (Tripodi et al., 2021). However, the productivity potential of this commodity is constantly threatened by yield fluctuations, which are predominantly caused by leaf- and fruit-damaging pests. At a global scale, pests and pathogens are estimated to cause yield losses of roughly 20–40% on major food crops every year, representing one of the most persistent threats to food security (Savary et al., 2019). Specifically, the greatest threat during the growth phase often comes from microscopic and flying insects, such as whiteflies (Bemisia tabaci), which act as a vector of the yellow (begomovirus) virus (Czosnek et al., 2017), and fruit flies (Bactrocera spp.), which can significantly reduce crop yields (Clarke et al., 2005).

The vulnerability of the growth phase to rapid pest invasion demands precise early detection, because a delay in identifying an attack in its early stage can trigger fatal physical damage that culminates in total crop failure (Bock et al., 2010). The fundamental problem that currently occurs in chili crop-protection practices is the high dependence on manual visual inspection. This conventional observation method has significant weaknesses, namely high vulnerability to subjective observer variation and human error, as well as a need for intensive time and labor when continuous monitoring is required (Barbedo, 2016; Singh &

* Corresponding author

Journal of Computer Networks, Architecture and High Performance Computing
Volume 08, Number 3, July 2026
https://doi.org/ 10.47709/cnahpc.v8i3.8780

Submitted : 2026-06-25
Accepted : 2026-07-06
Published : 2026-07-06

Misra, 2017). Delays in interpreting the type and presence of pests directly affect the accuracy of agronomic handling, such as the timing and dosage of pesticide application (Mohanty et al., 2016).

Along with the development of microcontroller technology, the use of the Internet of Things (IoT) has begun to be applied to assist real-time agricultural monitoring (Farooq et al., 2019; Elijah et al., 2018). Low-cost microcontrollers such as the ESP32 have been widely used as a platform for integrating sensors and wireless connectivity in smart-farming systems (Boursianis et al., 2022). However, most IoT systems developed in previous studies are still partial. The majority of systems rely on a soil moisture sensor merely to predict plant conditions, without the capability of actual visual validation (Khanna & Kaur, 2019). On the other hand, artificial-intelligence technology in the form of digital image processing based on Convolutional Neural Networks (CNN) for detecting pests and plant diseases has been widely developed (Barbedo, 2019; Liu & Wang, 2021; Thenmozhi & Reddy, 2019), but it is generally performed statically (offline) and has not been integrated into a monitoring instrument that operates in real time alongside the plant.

This creates a clear research gap. Existing solutions tend to separate environmental sensing from visual pest detection, and the visual detection itself is rarely deployed as a continuous, on-site instrument. The positioning of this study relative to prior work is summarized in Table 1. Based on this gap, this study designs and implements an IoT-based real-time pest monitoring system that integrates an ESP32-CAM image sensor and a capacitive soil moisture sensor on chili plants, using YOLOv8 as the detection model on the server side. The contribution of this study is threefold. First, it integrates visual pest detection and environmental sensing into a single autonomous instrument that operates in real time. Second, it applies a dedicated eight-stage image pre-processing pipeline to mitigate the domain gap between the high-quality training images and the low-cost ESP32-CAM production images, which is treated as the main methodological novelty of this work. Third, it evaluates the system not only from the accuracy of the detection model but also from the reliability and latency of data transmission for practical deployment.

Table 1. Research gap and novelty positioning against prior IoT / vision-based studies

Study / approach	Environmental sensing	On-site real-time visual detection	Domain-gap handling for low-cost camera
IoT soil/climate monitoring (e.g., Farooq et al., 2019; Boursianis et al., 2022)	Yes	No	Not applicable
Offline CNN/YOLO pest detection (e.g., Liu & Wang, 2021; Li et al., 2022)	No	No (offline / static)	Not addressed
Embedded crop pest detection (Hari & Kumar, 2023)	No	Partial	Limited
This study	**Yes**	**Yes (continuous, on-site)**	**Yes — eight-stage pipeline + lowered threshold**

LITERATURE REVIEW

Object detection based on deep learning has developed rapidly and become the dominant approach for visual recognition tasks in agriculture. The You Only Look Once (YOLO) family of models is widely adopted because it performs localization and classification in a single forward pass, which makes it suitable for near-real-time detection (Jocher et al., 2023). The latest variants improve both accuracy and inference efficiency, making them appropriate for deployment in resource-constrained agricultural scenarios.

* Corresponding author

Journal of Computer Networks, Architecture and High Performance Computing
Volume 08, Number 3, July 2026
https://doi.org/ 10.47709/cnahpc.v8i3.8780

Submitted : 2026-06-25
Accepted : 2026-07-06
Published : 2026-07-06

Several studies have applied convolutional neural networks and YOLO-based models to detect pests and plant diseases from leaf images, reporting that detection performance is strongly influenced by dataset quality, object size, and visual similarity between classes (Khan et al., 2023; Wang et al., 2023). Small pests with low visual distinctiveness are consistently reported as harder to detect, which is consistent with the per-class behavior observed in this study. In parallel, IoT-based monitoring has been increasingly used in precision agriculture to acquire environmental data such as soil moisture, air humidity, and temperature in real time (Nasution et al., 2022; Putra et al., 2023). Prior works demonstrate that low-cost microcontrollers and wireless transmission can deliver reliable monitoring data, but most of these systems focus on environmental parameters alone and do not incorporate on-site visual pest detection (Kashyap et al., 2021).

A synthesis of these two research streams reveals a consistent pattern: environmental-sensing studies achieve robust, low-latency telemetry but lack any biological confirmation of pest presence, whereas vision-based studies achieve strong recognition accuracy but operate offline on curated, high-quality imagery. Neither stream, on its own, delivers a field-deployable instrument that simultaneously senses the micro-environment and visually confirms pests in real time. Table 2 summarizes representative prior studies along these dimensions. This study positions itself precisely at the intersection of the two streams by combining a YOLOv8 visual detector with environmental sensing on a single ESP32-based platform, while explicitly addressing the domain gap introduced by a budget camera sensor.

Table 2. Synthesis of representative prior studies

Reference	Focus	Method / platform	Key limitation
Farooq et al. (2019)	IoT smart farming survey	IoT sensing architectures	No visual pest validation
Liu & Wang (2021)	Pest/disease detection	Deep-learning review (offline)	Not deployed in real time
Li et al. (2022)	Insect pest detection	Advanced CNN models	Curated high-quality images only
Nasution et al. (2022)	Soil-moisture monitoring	IoT environmental node	Environmental data only
Hari & Kumar (2023)	Real-time pest detection	Deep learning on embedded systems	No environmental sensing fusion
This study	**Integrated monitoring**	**YOLOv8 + ESP32-CAM + soil sensor**	**Recall limited by budget camera**

METHOD

This study employed an experimental design in which a hardware prototype of a pest monitoring system was developed by integrating an image sensor and a soil moisture sensor on chili plants. Field testing was conducted at Jl. Ambassador No. 73, Medan Selayang, Medan, North Sumatra, Indonesia. The system requires an internet connection through the internal Wi-Fi module of the ESP32 to connect communication between the microcontroller, a cloud database, and the monitoring interface. The operation of the system is automatic and real-time: the soil moisture sensor detects the water-content level of the growing media, while the ESP32-CAM captures visual images of the chili leaves, which are then processed using the YOLOv8 model to classify the presence of pests.

Rationale for Selecting YOLOv8

YOLOv8 was selected over alternative detectors for three reasons aligned with the constraints of this study. First, as a single-stage detector it performs localization and classification in one forward pass, giving substantially lower inference latency than two-stage detectors such as Faster R-CNN, which is essential for

* Corresponding author

Journal of Computer Networks, Architecture and High Performance Computing
Volume 08, Number 3, July 2026
https://doi.org/ 10.47709/cnahpc.v8i3.8780

Submitted : 2026-06-25
Accepted : 2026-07-06
Published : 2026-07-06

a responsive early-warning system. Second, its anchor-free detection head and decoupled classification/regression branches improve detection of small objects — a critical requirement given that many chili pests (e.g., aphids, spider mites) are very small. Third, the Ultralytics implementation provides a lightweight nano variant (YOLOv8n) and straightforward export paths (e.g., ONNX) suitable for low-cost server-side or edge deployment. These properties make YOLOv8 a better fit than older YOLOv5/YOLOv7 generations or SSD for a resource-constrained, latency-sensitive agricultural setting.

Hardware Configuration

The system is centered on the ESP32 microcontroller as the main controller and communication bridge to the cloud database via Wi-Fi. It integrates the ESP32-CAM module (OV2640 CMOS sensor with 4 MB external PSRAM) to capture leaf images, a WS2812B addressable LED ring to maintain lighting stability during image acquisition, and a capacitive soil moisture sensor to continuously monitor the growing-media condition. The ESP32-CAM communicates with the main ESP32 over a cross-wired UART link (TX2/RX2), while the soil moisture sensor output is read through the ADC pin (GPIO 34) and the LED ring is driven from GPIO 23. To operate autonomously in an open-field environment, the system is powered by a mini 5 V solar panel channeled through a TP4056 charger module to charge two 18650 Li-ion batteries (3.7 V nominal), with an MT3608 step-up module stabilizing the output to 5 V DC. The full electronic wiring is shown in Figure 1, and the deployed prototype is shown in Figure 2. Detailed hardware specifications are listed in Table 3.

Table 3. Hardware components and specifications

Component	Function / key specification
ESP32 DevKit v1	32-bit Tensilica Xtensa SoC (ESP-WROOM-32) with integrated Wi-Fi/Bluetooth; AMS1117 3.3 V LDO regulator; main controller and cloud bridge.
ESP32-CAM (OV2640)	CMOS image sensor with 4 MB PSRAM; captures leaf images at 800×600; communicates via UART (TX2/RX2).
Capacitive soil moisture sensor	Corrosion-resistant capacitive probe; analog output read on ADC pin GPIO 34; 30-sample outlier rejection applied in software.
WS2812B LED ring	Individually addressable RGB LEDs driven from GPIO 23 for stable illumination during capture.
Power subsystem	5 V mini solar panel → TP4056 charger → 2× 18650 Li-ion (3.7 V) → MT3608 step-up to 5 V; CC/CV charging with over-charge/over-discharge protection.

* Corresponding author

Journal of Computer Networks, Architecture and High Performance Computing
Volume 08, Number 3, July 2026
https://doi.org/ 10.47709/cnahpc.v8i3.8780

Submitted : 2026-06-25
Accepted : 2026-07-06
Published : 2026-07-06

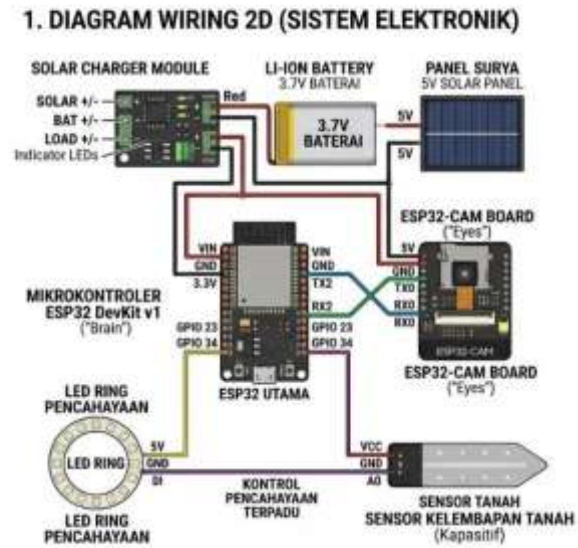

Figure 1. Wiring diagram of the IoT pest monitoring system.

Figure 2. Prototype of the monitoring device installed in the chili field.

Detection Model Training Configuration

The detection model was trained using the Pest Detection dataset obtained from Roboflow Universe, with a total of 38,449 images divided into 33,610 training images, 3,227 validation images, and 1,612 test images, covering 28 pest classes. Training was performed on Google Colab with GPU acceleration for 40 epochs, using an input image size of 480 pixels, a batch size of 64, the AdamW optimizer, an initial learning rate of 0.001, and cosine learning-rate scheduling. Mosaic augmentation with a probability of 0.5 was applied to improve generalization, and an early-stopping mechanism with a patience of 12 epochs was used

* Corresponding author

Journal of Computer Networks, Architecture and High Performance Computing
Volume 08, Number 3, July 2026
https://doi.org/ 10.47709/cnahpc.v8i3.8780

Submitted : 2026-06-25
Accepted : 2026-07-06
Published : 2026-07-06

to stop training automatically when the validation loss no longer improved significantly. These hyperparameters are summarized in Table 4.

Table 4. Training configuration of the YOLOv8 model

Parameter	Value
Base model	YOLOv8n (Ultralytics)
Dataset	Roboflow Pest Detection — 38,449 images, 28 classes
Train / val / test split	33,610 / 3,227 / 1,612
Epochs	40 (early stopping, patience 12)
Input image size	480 px
Batch size	64
Optimizer	AdamW
Initial learning rate	0.001 (cosine scheduling)
Augmentation	Mosaic, p = 0.5

Image Pre-processing Pipeline and Confidence Threshold

Because the training images were high-quality images taken from smartphone and DSLR cameras while the production images were captured with the budget OV2640 sensor of the ESP32-CAM, a domain gap exists that can reduce detection accuracy. To address this, the Flask server applies an eight-stage image pre-processing pipeline before detection. The eight stages are applied sequentially: (1) gray-world white balance, to correct the bluish/purplish cast typical of the OV2640; (2) warm color-temperature correction, to make the image closer to natural lighting; (3) automatic brightness normalization to a consistent target; (4) saturation enhancement, to sharpen color separation between objects; (5) bilateral filtering, to remove noise while preserving edges; (6) unsharp masking, for light edge sharpening; (7) Contrast Limited Adaptive Histogram Equalization (CLAHE), for adaptive local contrast; and (8) 2× upscaling using the Lanczos4 algorithm, to increase the number of analyzable pixels.

In addition, the confidence threshold was set to 0.10, lower than the YOLO default of 0.25. This choice is justified by the characteristics of ESP32-CAM imagery: the limited resolution and dynamic range of the OV2640 systematically depress detection confidence, so retaining the default threshold would discard a large fraction of true detections. Lowering the threshold increases detection sensitivity (recall) at the cost of more false positives that must be verified by the farmer — an acceptable trade-off for an early-warning system where missed detections are more costly than false alarms.

Evaluation Procedure

The system was evaluated from two perspectives. The detection model was assessed on the 1,612 held-out test images using precision, recall, mean Average Precision at a threshold of 0.5 (mAP@0.5), mean Average Precision over the 0.5–0.95 threshold range (mAP@0.5:0.95), and F1-score. The IoT system was evaluated based on the success and latency of sensor-data transmission to the server, as well as the success of on-site pest detection from images captured directly by the ESP32-CAM in the chili field.

* Corresponding author

Journal of Computer Networks, Architecture and High Performance Computing
Volume 08, Number 3, July 2026
https://doi.org/ 10.47709/cnahpc.v8i3.8780

Submitted : 2026-06-25
Accepted : 2026-07-06
Published : 2026-07-06

RESULT AND DISCUSSION

Training and Validation

The training process showed a consistent decrease in loss and an increase in validation accuracy. The loss curves for both training and validation declined steadily without significant divergence, while the mAP curve increased consistently until reaching a convergence point, indicating that the model successfully learned the visual patterns of the various pests. The absence of a widening gap between training and validation loss, combined with the use of mosaic augmentation and early stopping (patience 12), indicates that the model did not exhibit significant overfitting; training effectively halted once the validation loss plateaued. Figure 3 shows the validation precision and recall per epoch.

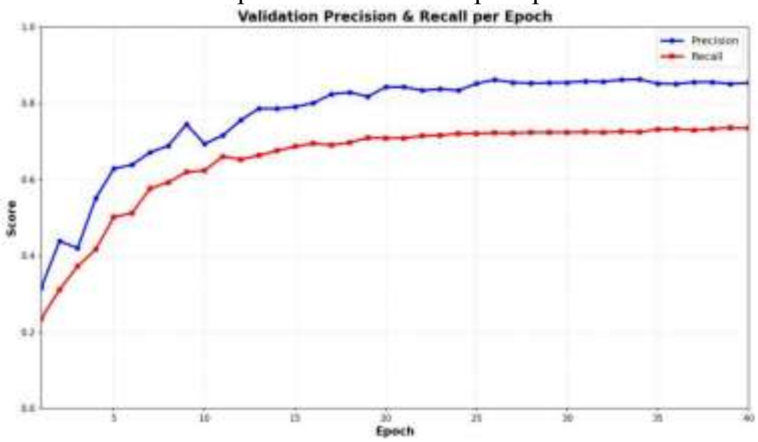

Figure 3. Validation precision and recall per epoch during YOLOv8 training.

Evaluation on the Test Set

After training, the model was evaluated on 1,612 test images that had never been seen by the model. As summarized in Table 5, the model achieved a precision of 0.853, a recall of 0.735, an mAP@0.5 of 0.778 (77.8%), an mAP@0.5:0.95 of 0.606, and an F1-score of approximately 0.79. The mAP@0.5 of 77.8% indicates that the model is able to detect pest objects with reasonably accurate bounding-box positioning. The precision of 0.853 indicates that most of the detections produced by the model were correct, while the recall of 0.735 indicates that the model captured approximately 73.5% of all pests actually present in the test images.

Table 5. Evaluation results of the YOLOv8 detection model

Metric	Value
Precision	0.853
Recall	0.735
mAP@0.5	0.778 (77.8%)
mAP@0.5:0.95	0.606
F1-Score	≈ 0.790

The model performance was not uniform across all 28 pest classes. Several pest classes achieved an mAP@0.5 above 0.95, indicating very good detection capability, while a few others only reached values below 0.55. This difference is influenced by several factors, including the size of the pest object in the image, visual similarity between pest types, and variation in the number of training images per class. Very small pests such as spider mites and aphids were harder to recognize because their visual appearance is not

* Corresponding author

Journal of Computer Networks, Architecture and High Performance Computing
Volume 08, Number 3, July 2026
https://doi.org/ 10.47709/cnahpc.v8i3.8780

Submitted : 2026-06-25
Accepted : 2026-07-06
Published : 2026-07-06

very distinctive, whereas larger pests with specific characteristics such as whiteflies and Helicoverpa were easier to detect. Table 6 contrasts the best- and worst-performing classes.

Table 6. Classes with the highest and lowest detection accuracy

Highest accuracy (mAP@0.5 > 0.95)	Lowest accuracy (mAP@0.5 < 0.55)
Hellula undalis Leaf Webber Helicoverpa Leucinodes Fruit Sucking Moth Root Grubs Blister Beetle Whitefly	Spider Mites Fall Armyworms Aphids Armyworms Corn Borers

The confusion matrix (Figure 4) further showed that most misclassifications occurred between pairs of pest classes with high morphological similarity, such as armyworms and fall armyworms, and aphids and spider mites. This indicates that the dominant error mode is inter-class confusion among visually similar small pests rather than wholesale failure to localize objects, which is consistent with reports in prior YOLO-based pest detection studies (Li et al., 2022; Wang et al., 2023).

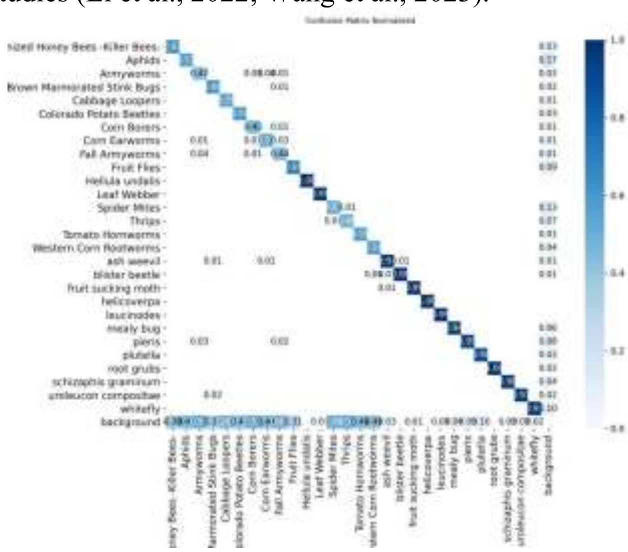

Figure 4. Confusion matrix of the model on the test set.

Pre-processing and Environmental Sensing

The eight-stage pre-processing pipeline was shown to help improve the detection confidence on ESP32-CAM images by reducing the domain gap between the training images and the production images. On the environmental-sensing side, as summarized in Table 7, all sensors on the field node successfully transmitted data to the server with an average latency below two seconds, which is highly adequate for agricultural monitoring because environmental conditions do not change drastically within seconds. The stability of data transmission was further supported by a multi-Wi-Fi strategy that automatically switches to a backup network when the primary connection is lost.

* Corresponding author

Journal of Computer Networks, Architecture and High Performance Computing
Volume 08, Number 3, July 2026
https://doi.org/ 10.47709/cnahpc.v8i3.8780

Submitted : 2026-06-25
Accepted : 2026-07-06
Published : 2026-07-06

Table 7. Sensor data transmission test on the field node

No.	Sensor data type	Reading	Transmission status	Latency
1	Air temperature (DHT22)	29.4 °C	Success	± 1.2 s
2	Air humidity (DHT22)	72 %	Success	± 1.1 s
3	Light intensity (BH1750)	12,480 lux	Success	± 1.3 s
4	Soil moisture	48 %	Success	± 1.4 s
5	Combined (all sensors, JSON)	JSON	Success	± 1.5 s

On-site pest detection using images captured directly from the ESP32-CAM installed in the chili field confirmed that the system was able to detect pests under real production conditions, with detection results stored on the server and accessible through the mobile application via the generated URL. Example detection outputs with bounding boxes are shown in Figure 5. These results indicate that the integration of visual detection and environmental sensing produces a responsive and practical early-warning system.

Figure 5. Example of pest detection results with bounding boxes.

DISCUSSION

The results confirm that the integration of an IoT platform and a deep-learning detection model can produce an effective real-time pest monitoring system for chili plants. The detection model achieved an mAP@0.5 of 77.8% with a high precision of 0.853, which means that the system rarely produces false alarms and is therefore reliable for guiding agronomic decisions such as the timing and dosage of pesticide application. The lower recall of 0.735 is an expected consequence of two interacting factors: the deliberately lowered confidence threshold of 0.10, and the limited image quality of the budget OV2640 sensor. Because the OV2640 uses a fixed-focus plastic lens with constrained resolution and dynamic range, fine-grained features of small pests are partially lost, which depresses recall most strongly for exactly the small, low-distinctiveness classes (spider mites, aphids) identified in Table 6. In other words, the recall gap is driven primarily by sensor hardware rather than by a deficiency of the detector itself.

The non-uniform per-class performance provides an important practical implication. Because detection accuracy depended strongly on object size, inter-class visual similarity, and the amount of training data per class, future improvement should focus on enriching the training data for the hardest classes and on reducing morphological confusion between similar pests, rather than only increasing the overall dataset size. The eight-stage pre-processing pipeline and the lowered confidence threshold were necessary design choices to mitigate the domain gap.

Compared with previous IoT monitoring systems that rely only on environmental sensors (Nasution et al., 2022; Kashyap et al., 2021), the proposed system adds an on-site visual validation capability, so that anomalies on the leaves can be confirmed as pest-related rather than merely the result of water stress — a confirmation that environment-only systems structurally cannot provide. Compared with offline image-

* Corresponding author

Journal of Computer Networks, Architecture and High Performance Computing
Volume 08, Number 3, July 2026
https://doi.org/ 10.47709/cnahpc.v8i3.8780

Submitted : 2026-06-25
Accepted : 2026-07-06
Published : 2026-07-06

classification studies (Liu & Wang, 2021; Li et al., 2022), the proposed system deploys detection as a continuous real-time instrument that operates alongside the plant. The average transmission latency below two seconds confirms the practical suitability of the system for field deployment. From a smart-farming perspective, the practical implication is concrete: a low-cost, solar-powered node that both senses the micro-environment and visually confirms pests enables smallholder farmers to act on objective evidence within seconds, supporting more precise and timely pesticide application and reducing reliance on subjective manual scouting.

Limitations

Several limitations were identified during testing. First, the OV2640 sensor is a budget, fixed-focus camera, so the captured imagery is constrained in sharpness, color accuracy, and dynamic range; enhancement techniques (including the eight-stage pipeline and trials with Real-ESRGAN super-resolution) cannot fully overcome these physical limits. Second, a domain gap remains between the high-resolution training data and the 800×600 ESP32-CAM production imagery, so the effective field detection accuracy is estimated to fall in the 40–55% range, below the 77.8% mAP measured on the Roboflow test set. Third, the VPS used for inference does not support the AVX2/AVX-512 instruction sets required by the oneDNN PyTorch backend, so oneDNN optimization and Test-Time Augmentation were disabled and inference ran in inference-only mode, lengthening per-image inference time (approximately 5–8 seconds per image). Fourth, the model operates under a closed-world assumption and can recognize only the 28 trained classes; visually obvious pests outside these classes are not detected. Fifth, the capacitive soil moisture sensor is sensitive to power-supply noise from the MT3608 booster, requiring software outlier rejection over 30 samples for stable readings. Finally, the evaluation was conducted on a single dataset and a limited field-testing setting.

Future Research

Future work should address the limitations above along several directions. (1) Camera upgrade: replacing the OV2640 with a 5 MP auto-focus OV5640 to eliminate fixed-focus blur and improve recall. (2) Edge AI: porting inference toward edge-capable hardware such as the ESP32-S3, or exporting the model to ONNX Runtime for broader CPU compatibility on non-AVX servers, and alternatively using a GPU-equipped VPS (e.g., NVIDIA T4) to shorten inference time and re-enable Test-Time Augmentation. (3) Multi-season and multi-location evaluation: testing across multiple growing seasons and larger, multi-hectare fields under more diverse weather conditions to assess robustness. (4) Balanced, field-native datasets: building a balanced training set captured directly from ESP32-CAM imagery to close the domain gap and improve per-class consistency. (5) Hybrid environmental sensing and push notifications: integrating additional environmental parameters and adding Firebase Cloud Messaging push alerts so that farmers are notified immediately when a pest is detected.

CONCLUSION

This study designed and implemented an IoT-based real-time pest monitoring system for chili plants that integrates an ESP32-CAM image sensor and a capacitive soil moisture sensor, with YOLOv8 as the detection model on the server side. The detection model achieved a precision of 0.853, a recall of 0.735, an mAP@0.5 of 0.778, an mAP@0.5:0.95 of 0.606, and an F1-score of approximately 0.79 on 1,612 test images, while all sensor data were transmitted with an average latency below two seconds. These results confirm that the integration of visual detection and environmental sensing produces an accurate and responsive early-warning system that preserves low-cost deployment characteristics.

The main contribution of this study lies in combining on-site visual pest detection with environmental sensing into a single autonomous instrument, supported by a dedicated eight-stage pre-processing pipeline that mitigates the domain gap of low-cost camera images. Future work should extend the evaluation to more

* Corresponding author

Submitted : 2026-06-25
Accepted : 2026-07-06
Published : 2026-07-06

diverse field conditions and multiple growing seasons, improve the camera quality and pre-processing pipeline to increase recall, port inference toward edge AI, and balance the training data across pest classes to improve per-class consistency.

REFERENCES

Badan Pusat Statistik. (2023). Statistik hortikultura: Produksi tanaman cabai di Indonesia. Badan Pusat Statistik.

Barbedo, J. G. A. (2016). A review on the main challenges in automatic plant disease identification based on visible range images. Biosystems Engineering, 144, 52–60. https://doi.org/10.1016/j.biosystemseng.2016.01.017

Barbedo, J. G. A. (2019). Plant disease identification from individual lesions and spots using deep learning. Biosystems Engineering, 180, 96–107. https://doi.org/10.1016/j.biosystemseng.2019.02.002

Bock, C. H., Parker, P. E., Cook, A. Z., & Gottwald, T. R. (2010). Plant disease severity estimated visually, by digital photography and image analysis, and by hyperspectral imaging. Critical Reviews in Plant Sciences, 29(2), 59–107. https://doi.org/10.1080/07352681003617285

Boursianis, A. D., Papadopoulou, M. S., Diamantoulakis, P., Liopa-Tsakalidi, A., Barouchas, P., Salahas, G., Karagiannidis, G., Wan, S., & Goudos, S. K. (2022). Internet of Things (IoT) and agricultural unmanned aerial vehicles (UAVs) in smart farming: A comprehensive review. Internet of Things, 18, 100187. https://doi.org/10.1016/j.iot.2020.100187

Clarke, A. R., Armstrong, K. F., Carmichael, A. E., Milne, J. R., Raghu, S., Roderick, G. K., & Yeates, D. K. (2005). Invasive phytophagous pests arising through a recent tropical evolutionary radiation: The Bactrocera dorsalis complex of fruit flies. Annual Review of Entomology, 50, 293–319. https://doi.org/10.1146/annurev.ento.50.071803.130428

Czosnek, H., Hariton-Shalev, A., Sobol, I., Gorovits, R., & Ghanim, M. (2017). The incredible journey of begomoviruses in their whitefly vector. Viruses, 9(10), 273. https://doi.org/10.3390/v9100273

Dewi, T., Risma, P., & Oktarina, Y. (2024). Fruit sorting robot based on color and size for an agricultural product packaging system. Bulletin of Electrical Engineering and Informatics, 13(1), 1–10.

Elijah, O., Rahman, T. A., Orikumhi, I., Leow, C. Y., & Hindia, M. N. (2018). An overview of Internet of Things (IoT) and data analytics in agriculture: Benefits and challenges. IEEE Internet of Things Journal, 5(5), 3758–3773. https://doi.org/10.1109/JIOT.2018.2844296

Farooq, M. S., Riaz, S., Abid, A., Abid, K., & Naeem, M. A. (2019). A survey on the role of IoT in agriculture for the implementation of smart farming. IEEE Access, 7, 156237–156271. https://doi.org/10.1109/ACCESS.2019.2949703

Hari, A., & Kumar, S. (2023). Real-time crop pest detection using deep learning on embedded systems. Journal of Real-Time Image Processing, 20(4), 1–14. https://doi.org/10.1007/s11554-023-01312-9

Jocher, G., Chaurasia, A., & Qiu, J. (2023). Ultralytics YOLOv8 (Version 8.0.0) [Computer software]. https://github.com/ultralytics/ultralytics

Kashyap, P. K., Kumar, S., Jaiswal, A., Prasad, M., & Gandomi, A. H. (2021). Towards precision agriculture: IoT-enabled intelligent irrigation systems using deep learning neural network. IEEE Sensors Journal, 21(16), 17479–17491. https://doi.org/10.1109/JSEN.2021.3069266

Khan, S., Tufail, M., Khan, M. T., Khan, Z. A., & Anwar, S. (2023). Deep learning-based identification system for pest and disease detection in precision agriculture. Computers and Electronics in Agriculture, 205, 107611. https://doi.org/10.1016/j.compag.2023.107611

Khanna, A., & Kaur, S. (2019). Evolution of Internet of Things (IoT) and its significant impact in the field of precision agriculture. Computers and Electronics in Agriculture, 157, 218–231. https://doi.org/10.1016/j.compag.2018.12.039

Li, W., Zhu, T., Li, X., Dong, J., & Liu, J. (2022). Recommending advanced deep learning models for efficient insect pest detection. Agriculture, 12(7), 1065. https://doi.org/10.3390/agriculture12071065

Liu, J., & Wang, X. (2021). Plant diseases and pests detection based on deep learning: A review. Plant Methods, 17(1), 22. https://doi.org/10.1186/s13007-021-00722-9

Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using deep learning for image-based plant disease detection. Frontiers in Plant Science, 7, 1419. https://doi.org/10.3389/fpls.2016.01419

Nasution, T. H., Siregar, I., & Yasir, M. (2022). IoT-based soil moisture monitoring system for precision agriculture. IOP Conference Series: Earth and Environmental Science, 977(1), 012001.

* Corresponding author

Submitted : 2026-06-25
Accepted : 2026-07-06
Published : 2026-07-06

Putra, A. W., Suryani, E., & Wahyuningsih, D. (2023). Smart farming monitoring system based on ESP32 and cloud database. Indonesian Journal of Electrical Engineering and Computer Science, 30(2), 720–730.

Rahman, C. R., Arko, P. S., Ali, M. E., Khan, M. A. I., Apon, S. H., Nowrin, F., & Wasif, A. (2022). Identification and recognition of rice diseases and pests using convolutional neural networks. Biosystems Engineering, 213, 1–12. https://doi.org/10.1016/j.biosystemseng.2021.11.005

Saleem, M. H., Potgieter, J., & Arif, K. M. (2022). Automation in agriculture by machine and deep learning techniques: A review of recent developments. Precision Agriculture, 22(6), 2053–2091. https://doi.org/10.1007/s11119-021-09806-x

Savary, S., Willocquet, L., Pethybridge, S. J., Esker, P., McRoberts, N., & Nelson, A. (2019). The global burden of pathogens and pests on major food crops. Nature Ecology & Evolution, 3(3), 430–439. https://doi.org/10.1038/s41559-018-0793-y

Singh, V., & Misra, A. K. (2017). Detection of plant leaf diseases using image segmentation and soft computing techniques. Information Processing in Agriculture, 4(1), 41–49. https://doi.org/10.1016/j.inpa.2016.10.005

Sudiono, S., & Yasin, B. (2006). Karakterisasi kutu kebul (Bemisia tabaci) sebagai vektor virus gemini dengan teknik PCR-RAPD. Jurnal Hama dan Penyakit Tumbuhan Tropika, 6(2), 113–121.

Thenmozhi, K., & Reddy, U. S. (2019). Crop pest classification based on deep convolutional neural network and transfer learning. Computers and Electronics in Agriculture, 164, 104906. https://doi.org/10.1016/j.compag.2019.104906

Tripodi, P., et al. (2021). Global range expansion history of pepper (Capsicum spp.) revealed by over 10,000 genebank accessions. Proceedings of the National Academy of Sciences, 118(34), e2104315118. https://doi.org/10.1073/pnas.2104315118

Wang, X., Zhang, S., Wang, X., Xu, C., & Nagwanshi, K. K. (2023). Crop pest detection by three-scale convolutional neural network with attention. PLOS ONE, 18(6), e0276456. https://doi.org/10.1371/journal.pone.0276456

* Corresponding author