Real-Time Chili Harvest Estimation with Object Detection Technology in Enhancing Agricultural Efficiency

Arjon Turnip, Lidia Marselina, Endra Joelianto, Poltak Sihombing, and Putra Sumari

***Abstract*— Agriculture plays a critical role in ensuring food security, yet technological advancements in this sector remain limited compared to its potential. Modern innovations, such as Object Detection, offer promising solutions to enhance agricultural efficiency and productivity. This study explores the application of the YOLOv8 algorithm, the latest evolution of the YOLO object detection framework, to detect chili fruits rapidly and accurately. By employing this method, farmers can estimate potential chili harvests, streamlining yield prediction and improving decision-making processes in real time. Experimental results demonstrate that the model achieved a mean Average Precision (mAP) of 75.4%, an F1 score of 77.21%, a Precision of 74.7%, a Recall of 79.9%, and a processing speed of 6.9 milliseconds per image. These results highlight the model's effectiveness in practical applications but also indicate room for improvement, as performance is influenced by the limited number of training iterations. Future work could focus on increasing training iterations and expanding the dataset to enhance detection accuracy and robustness, ultimately supporting precision agriculture advancements.**

***Index Terms*—Object detection, YOLO algorithm, Real-time, Precision agriculture, Chili Pepper.**

I. INTRODUCTION

THE ever-evolving modern society has led to the growth of global consumption levels [1]. According to the Central Bureau of Statistics, Indonesia's market expenditure share in 2022 amounted to 50.14 percent with an increase of 0.89 percent from the previous year. The share of food expenditure itself is inversely proportional to food security. The higher the food expenditure figure, the worse the food security value [2]. Indonesia itself has committed to maintaining national food security by increasing production in agriculture, with a contribution of 12.98% to the national economy [3]. Agriculture also has the highest percentage for the main employment structure in Indonesia in 2022 at 28.61% [4]. However, in Indonesia agriculture is still done conventionally which is not effective and efficient. Not much technology has been applied to this field.

There are several main problems that become obstacles and need to be solved, namely the accuracy of detection needs to be continuously improved, the speed of inference in the model used, and the use of appropriate and lightweight models for data processing [5]. Inefficient agricultural methods can cause agricultural pollution, and in the event of crop disorders, such as diseases and defects, manual diagnosis will be very slow, laborious, and require large capital [5-6].

The development of technology plays a major role in agriculture [1-3][5][6-10] which allows farmers to get information quickly, accurately, and take immediate action if there are obstacles, and can improve the quality of crops [5][11]. One technology that is very useful in agriculture is Image Processing technology, used as an object detector (Object Detection) [12]. Object Detection technology has been widely developed and applied in various fields, such as disease detection in the health sector, facial recognition in the security sector, automatic drivers, smart robots, and so on. In agriculture, using Object Detection technology to monitor the state of plants is much more efficient than manual detection [13-14]. This is because Object Detection technology is able to monitor the plants at all times while the system is active, so that if something happens that is not as it should be, it can be detected immediately. This process is very beneficial for plant growth.

Basically, Object Detection algorithms have been developed for plant detection by extracting target features into complex scenes. Several techniques, including LiDAR, sonar, RGB-Depth Map (RGB-D) Imaging Vision, Region-based Convolutional Neural Networks (R-CNN), and Feedforward Neural Networks (FNN), utilize convolution operations and feature intricate architectures. Numerous CNN-based algorithms, such as YOLO and Faster R-CNN, have been extensively employed for fruit object detection [5][11].

One method that is now being developed for Object Detection is You Only Look Once (YOLO). YOLO has far superior detection accuracy and speed compared to other methods [1][5][12-13]. This method falls into the category of

A. Turnip and L. Marselina are with Department of Electrical Engineering, Universitas Padjadjaran, Indonesia (e-mail: turnip@unpad.ac.id*, lidiamarserlina.16@gmail.com).

E. Joelianto is with Instrumentation, Control, and Automation Research Group, Institut Teknologi Bandung, Indonesia (e-mail: ejoel@itb.ac.id).

P. Sihombing is with Department of Computer Science, Universitas Sumatera Utara, Medan, Indonesia (e-mail: poltak@usu.ac.id).

P. Sumari is with School of Computer Sciences, Universiti Sains Malaysia, Malaysia (e-mail: putras@usm.my).

one-stage algorithms, which obtain location and category information of objects directly [15-16]. The latest research of YOLO is YOLOv7 which is designed using bag-of-freebies for real time detection with high accuracy and low inference cost [1][17].

Several object detection methods have been developed, such as the one developed by H. S. Gill et al. [18] that improved fruit recognition from images using deep learning. The research combines CNN, RNN, and LSTM methods for fruit classification based on optimal features and selected derivatives, with the resulting accuracy being very effective. However, because it combines 3 methods at once, the computational process takes a long time and is a relatively heavy process [18].

Object Detection methods using two-stage algorithms such as R-CNN, LetNet, ResNet, SPP-Net, and the like have also been widely researched. As conducted by S. Wan and S. Goudos [14] in their research discussing the Faster R-CNN method for detecting multi-class fruits using 4000 images resulting in an increase in convolutional and pooling layers achieved has a high and fast accuracy of 91% mAP. W. Zhang et al. [19] have also conducted research using the ResNet-50 method to evaluate wheat quality. The research resulted in an average accuracy of 97.5%. One of the causes of the lack of accuracy is due to the image quality that is often lacking. The integration of LeNet-5, ResNet-34, and VGG-16 models with an image enhancement approach to refine particle features in images has been explored, resulting in a 1% accuracy improvement over models without enhancement [20].

Recent research has favored one-stage methods such as SSD, YOLO, and RPN [15] over two-stage methods. Some studies that use this one-stage method include research conducted by K. Cai et al. [21] discussed fish detection using YOLOv3 and MobileNetv1. The results obtained are data of 2000 smaller images, thereby speeding up the computing process. YOLO is one method that continues to be developed and widely used. Until now, the latest version is YOLOv7. C.-Y. Wang et al. [17] conducted research using the YOLOv8 method by training bag-of-freebies which resulted in effectively reducing about 40% parameters and 50% object detector calculations, thus having a high inference speed. D. Wu et al. [25] also conducted research using YOLOv7 and Data Augmentation for Camellia Oleifera fruit detection which resulted in an average value of 96.03% mAP, 94.76% precision, 95.54% recall, F1 score of 95.15%, and detection time of 0.025 seconds per image. From some of these sources, it can be seen that there has been no research to detect chili peppers using YOLOv7.

This paper considers chili peppers growth detection using the YOLOv8 method [22-25]. In this investigation, YOLOv8 will be applied to the detection of chili fruit on chili plants. The results of this detection can be developed into several applications, such as estimating the number of chilies to be harvested, calculation of health and fertility in chili plants, detection of damage to chili fruit before harvesting, detecting the type or size of chili fruit, and so on.

II. Methodology

A. Data Collection

This research was conducted in an outdoor chili farm located in the D3 area of the Faculty of Agriculture, Universitas Padjadjaran, Sumedang, Indonesia. There are 18 rows of beds with a length and width of 6 meters and 1.2 meters respectively. The tools and materials used in this Object Detection research are IP Camera, Jetson Nano/Laptop, website, and chili plants. The camera is installed in the field to get image data of chili peppers to be studied. The image is then forwarded to the Jetson Nano/Laptop as a processor to be processed using the YOLOv8 algorithm. In this case, the data used as a program dataset is obtained from various sources on the internet media, as a temporary substitute, during the trial process. The results obtained from the process are then displayed on the website in the form of information that can be read easily by users. This research is one of a series of studies that discuss the irrigation system of chili plants, which consists of Soil Moisture, Air Temperature and Humidity, and Soil pH systems as shown in Fig. 1.

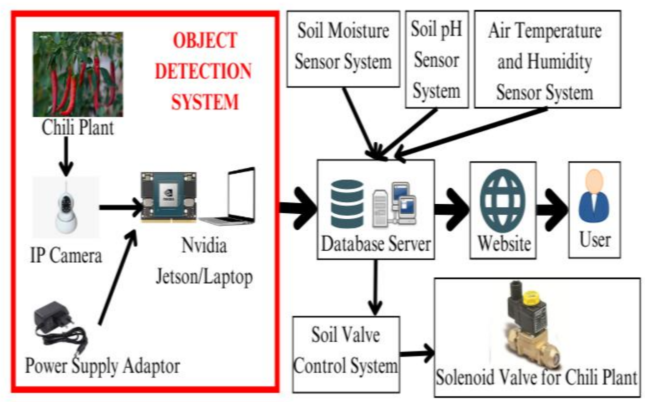

Fig 1. Block Diagram of the Whole System

B. YOLOv8 Architecture

You Only Look Once (YOLO) is a single-stage Object Detection algorithm that has high detection accuracy and speed. The current development version, YOLOv8, is estimated to be 120% faster than YOLOv5, 180% faster in terms of FPS compared to YOLO X, 1200% faster than Dual-Swin-T, 550% faster than ConvNext, and 500% faster than SWIN-L.

The YOLO algorithm usually performs classification and object detection simultaneously by only looking once at the input image or video, because YOLO is a single-stage object detection method. This method is able to speed up the detection process in YOLO, compared to other methods that belong to the two-stage object detection method.

YOLOv8 consists of four networks, namely input, backbone, neck, and head or dense prediction as illustrated in Fig. 2. The input network is the outermost part where the image or video to be processed is input in the form of a 2 or 3-dimensional RGB array. In this section, the image or video is resized to 640 x 640 pixels or 640 x 640 x 3, Mosaic data enhancement, and image scaling before being fed into the backbone network. In the case of chili peppers, the resulting dataset has non-uniform samples, with small and large samples. The input image will be adaptively scaled, if the sample is too small it will be spliced, and if the input image is too large it will be cropped, so as to achieve an even size.

The second YOLOv8 network from the outside, the backbone, is the Convolutional Layer or where the Pre-Trained Neural Network occurs. This network contains 50 integrated modules, including the CBS composite module, Efficient Layer Aggregation Networks (ELAN), and MP. The CBS composite module comprises standard convolution, a BN layer, and SiLU, which together facilitate the activation function through these three components. The architecture of the ELAN module in YOLOv8 is modified into Extended Efficient Layer Aggregation Networks (E-ELAN). E-ELAN uses group shift convolution to extend the channel and cardinality of computation blocks. The feature map of each compute block will be randomized to a group size g and merged together. This aims to perform cardinality merging. The MP module consists of Maxpool and CBS, which are divided into MP1 and MP2.

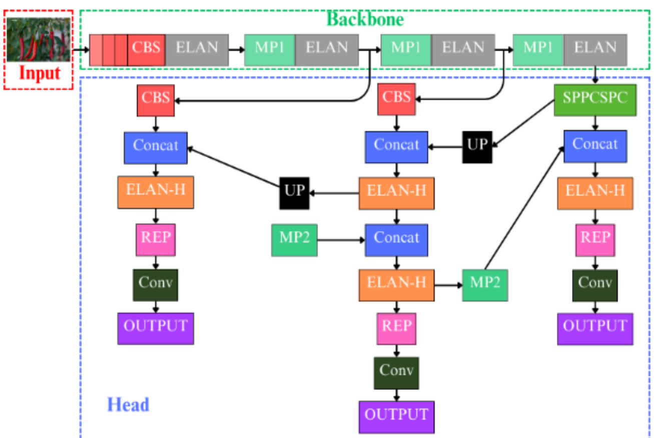

Fig. 2. YOLOv8 Architecture

The third network on YOLOv8 from the outside is the neck. The neck network is the connecting network between the backbone and the head. This network functions as an Object Detector Model that collects feature maps for different stages. This is because in the head network the classification and localization of objects are performed. In the neck network, there is no special process like in the input and backbone, so this network is often not mentioned in the YOLOv8 structure.

The head network or dense prediction in YOLOv8 combines the advantages of Feature Pyramid Network (FPN) and Path Aggregation Network (PAN), which then becomes PA-FPN. This network architecture incorporates the SPPCSPC module along with a series of CBS, MP, Catconv, and repconv modules. The SPPCSPC module shares similarities with the SPPF module in YOLOv5, designed to enhance the receptive field and improve feature extraction for object detection tasks. Initially, an input feature map of size $512 \times 20 \times 20$ is processed through three successive convolution operations to refine feature representation. To capture multi-scale spatial information, max-pooling operations are applied three times with kernel sizes of 5, 9, and 13, where adaptive padding ensures proper alignment. Finally, the output feature map, maintaining its original size of $512 \times 20 \times 20$, is obtained by integrating the results, utilizing only the 1×1 convolution operation without additional pooling layers. This design enables the SPPCSPC module to effectively extract multi-scale object features while preserving spatial resolution, contributing to more robust and accurate object detection.

C. Images and Dataset Preprocessing

First, a total of 1099 chili image data was collected to be used as a dataset. There are no repeated images in the dataset, aiming to prevent model over fitting. The dataset is divided into 94% train data, 4% validation data, and 2% test data. The data will then enter the labeling process using the Roboflow platform. The image dataset will be given a bounding box outside the target chili fruit to be labeled based on the smallest bounding box around the chili fruit, to ensure that the background area of the image is carried in the bounding box as little as possible. Before the data is exported, the image is automatically augmented. This serves to multiply the image data, without repetition. The image can be augmented into several models such as mirrored horizontally or vertically, rotated in various directions and angles, given cover or noise, color changes, and so on.

The dataset was exported by multiplying the data by twice the previous data, by doing vertical and horizontal reversals, and rotating the image copy 90 degrees to the right, left, top, and bottom. This resulted in a dataset of 2077 different images. TXT annotations and YAML configurations will be saved in YOLOv8 format in which there are bounding box coordinates for reference in the training process later. The use of YOLOv8 format also intends to simplify the data processing process. Where the data does not need to be converted back into a format supported by the YOLOv8 program.

D. Training Images and Datasets

This research was conducted on a computer with Windows 11 operating system, configured with NVDIA-SMI 525.85.12, GPU runtime, and PyTorch 2.1.0 deep learning framework. The programming language used was Python 3.10 with Google Collabority platform, CUDA 12.0, and Ultralystic YOLOv8.0.20.

The dataset images to be input were set at a resolution of 640 x 640 pixels, with a speed of 4.0ms pre-process, 7.1ms inference, 0.0ms loss, 3.1ms post-process per image. To record data, observe loss, and save model weights for each epoch during the training period, the Tensorboard visualization tool is used, which is built-in to Ultralystic YOLOv8. The model will continue to be trained until it reaches a certain accuracy value. The weight of the model file is then exported to be included in the fruit calculation algorithm if the model has reached the intended accuracy during this training process. However, if the accuracy value has not been reached then the training will continue to be repeated.

E. Indicator Evaluation

To evaluate the accuracy and performance of the model, Precision calculations are used to measure the level of True Positive values obtained from all positive predictions, Recall to measure the True Positive level of all predictions, Mean Average Precision (mAP) to measure the performance of the object detection model, and F1 score, with the following equation form:

Precision:

$$P = \frac{TP}{TP+FP} \times 100\% \quad (1)$$

Recall

$$R = \frac{Tp}{TP+FN} \times 100\% \quad (2)$$

Average Precision:

$$AP = \int_{p}^{a} \quad P(r)dr \quad (3)$$

Mean average Precision:

$$\frac{1}{n}\sum_{i=1}^{n} \quad AP_I \quad (4)$$

F1 Score:

$$F1 = 2 \times \frac{P \times R}{P+R} \quad (5)$$

where True Positive (TP) represents the number of correctly detected chili fruit objects; False Positive (FP) represents the number of other objects detected as chili fruit, and False Negative (FN) represents the number of undetected or missed chili fruit. Overall, this study can be summarized and represented in the flowchart shown in Fig. 3.

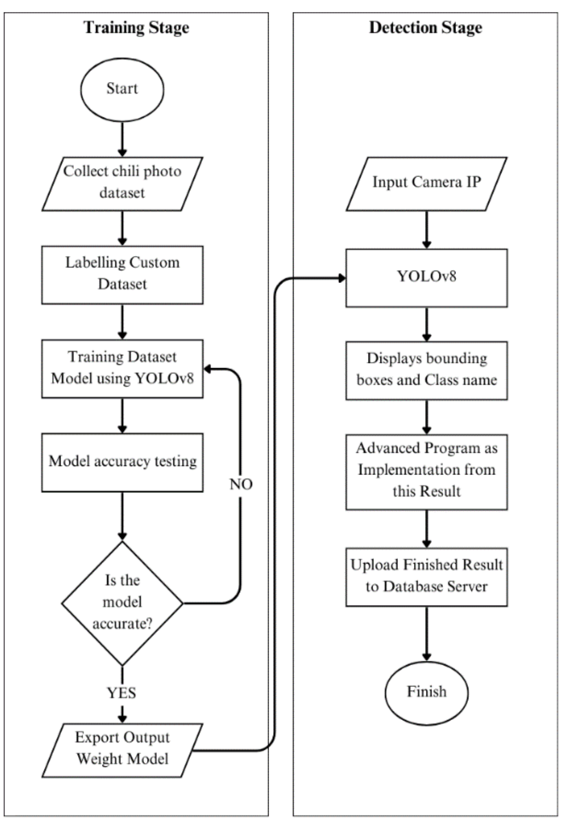

Fig. 3. Flowchart of Object Detection System

III. Result and Discussion

A. Indicator Achievement

The YOLOv8 model was employed for chili fruit detection, using diverse datasets from various media. Fig. 4 illustrates the curves for Box Loss, Class Loss, and Distribution Focal Loss, showing a consistent decrease, which indicates the model's improved localization and classification performance. The precision curve reveals a positive correlation with confidence thresholds; higher confidence values result in greater precision, demonstrating that true positive detections significantly outnumber false negatives. This highlights the model's reliability in accurately detecting chili fruits while minimizing missed instances, a critical factor for agricultural yield estimation. While the model shows strong performance, further improvements could be achieved by expanding the dataset with more diverse scenarios and fine-tuning hyperparameters. These steps would enhance robustness and ensure reliable detection across real-world conditions.

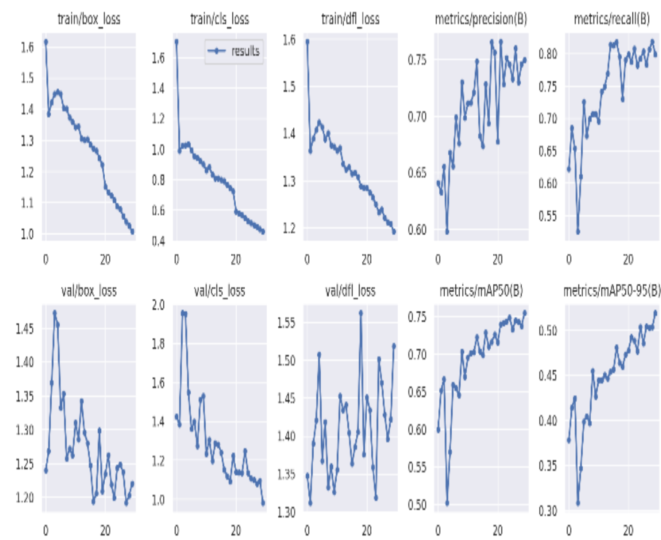

Fig. 4. (a) Precision Curve (b) Recall Curve

The results of the experiments are presented in Table 1. The data shows the accuracy values derived from the chili object detection images, based on calculations of precision, recall, average precision, mAP, and F1 score. The performance data for the chili fruit detection model across 15 trials reveals a consistent improvement in key metrics, including Precision, Recall, Average Precision (AP), Mean Average Precision (mAP), and F1 Score. Precision improves from 75.4% in trial 1 to 85.1% in trial 15, indicating that the model becomes increasingly adept at correctly identifying chili fruits while reducing false positives. Similarly, recall increases from 78.9% to 85.5%, reflecting the model's enhanced ability to detect relevant instances of chili fruits and minimize missed detections.

The Average Precision (AP) and Mean Average Precision (mAP) also show steady progress, rising from 74.7 and 75.4 in the first trial to 79.9 and 82.0 in the final trial, respectively. These metrics confirm the model's growing reliability in assigning accurate confidence scores and achieving consistent performance across diverse test cases. The F1 score, a balanced measure of precision and recall, climbs from 77.0 in trial 1 to 84.9 in trial 15, demonstrating that the model strikes an effective balance between identifying all chili fruits and accurately classifying them.

TABLE I
PREDICTED INDICATOR ACHIEVEMENT VALUE

Trial Number	Precision (%)	Recall (%)	AP	MAP	F1
1	75.4	78.9	74.7	75.4	77
2	76.1	79.3	75.4	75.9	77.5
3	76.9	79.8	75.6	76.5	78
4	77.2	80.1	75.9	77.2	78.4
5	77.8	80.5	76.3	77.8	78.9
6	80.6	80.9	76.7	78.1	79.5
7	81.1	81.4	77	78.4	80.3
8	81.8	81.8	77.3	78.9	80.7
9	82.3	82.5	77.7	79.4	81.4
10	83.6	83.1	77.9	79.8	81.9
11	83.9	83.8	78.2	80.1	82.5
12	84.2	84.1	78.5	80.5	83.1
13	84.4	84.8	79	80.7	83.7
14	84.7	85.3	79.3	81.4	84.3
15	85.1	85.5	79.9	82	84.9

This steady improvement across all metrics highlights the model's robustness and suitability for practical applications in agricultural systems. The rising mAP, in particular, underscores its capability to handle various challenging scenarios, such as detecting smaller or partially occluded chili fruits. By the final trial, the model achieves a high level of accuracy, precision, and recall, making it reliable for deployment in automated chili fruit detection tasks. Further enhancements could involve expanding the training dataset or refining the model parameters to further boost its performance.

The confusion matrix provides a comprehensive overview of the performance of the YOLOv8 model in detecting chili fruits across multiple trials. By breaking down the detection results into True Positives (TP), False Positives (FP), False Negatives (FN), and True Negatives (TN), it offers insights into the model's precision, recall, and overall classification accuracy. This analysis helps evaluate the effectiveness of the detection process and highlights areas for potential improvement. The confusion matrix derived from the data is presented in Fig. 5.

Fig. 6 demonstrates the relationship between confidence thresholds and key performance metrics—Precision, Recall, and F1-Score—for the YOLOv8 model in chili fruit detection. The Precision curve reflects the proportion of true positive detections relative to all positive predictions, generally improving as confidence increases due to a reduction in false positives. In contrast, the Recall curve shows the proportion of true positive detections among all actual positives, which tends to decline with higher confidence thresholds as stricter criteria exclude some true positives. The F1-Score curve balances precision and recall, highlighting the confidence level at which the model achieves optimal performance. This analysis is essential for determining the most effective confidence threshold for practical applications, ensuring the model delivers the desired balance between precision and recall.

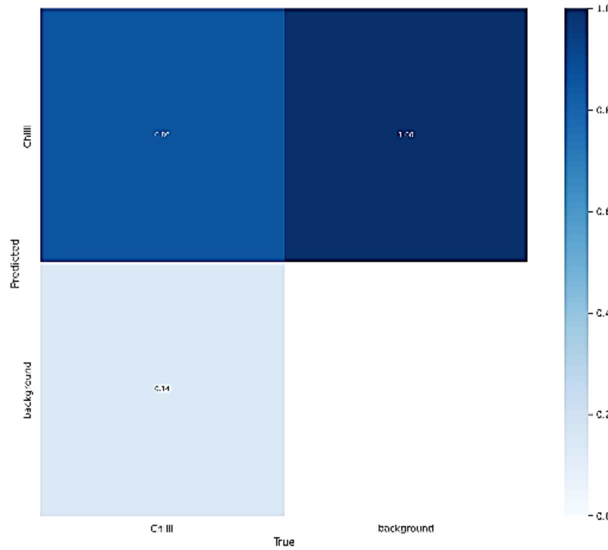

Fig. 5. Confusion Matrix

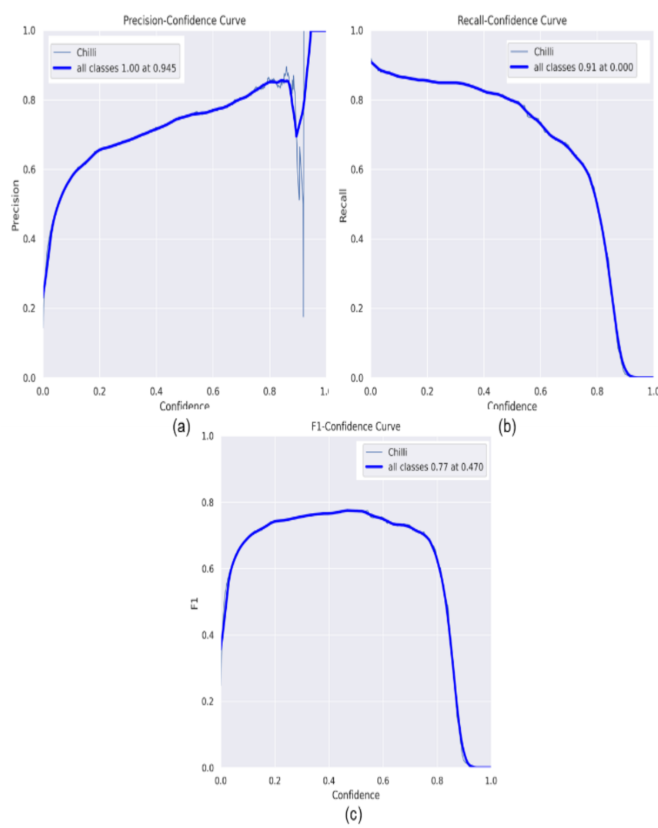

Fig. 6 .The relationship between confidence thresholds and key performance metrics—Precision, Recall, and F1-Score

B. *Object Detection Result*

The detection process was performed on both a test dataset and additional video footage of chili plants bearing fruit. The use of video highlights the algorithm's capability for real-time

ISSN: 1942-9703 / CC BY-NC-ND

object detection, demonstrating its practical applicability in dynamic environments. The results of object detection on still images from the test dataset are illustrated in Fig. 7(a), showcasing the model's accuracy in identifying and localizing chili fruits under controlled conditions. Meanwhile, Fig. 7(b) presents the real-time object detection results, demonstrating the algorithm's ability to maintain performance in live scenarios. This dual evaluation approach underscores the robustness and adaptability of the YOLOv8 model for agricultural applications. The real-time detection capability is particularly valuable for tasks such as automated harvesting, monitoring crop health, or yield estimation. Additionally, it reflects the model's capacity to handle variable lighting, motion, and occlusion commonly encountered in field environments. By integrating video-based detection, the system demonstrates potential for deployment in smart farming solutions, offering both scalability and efficiency in managing large agricultural areas.

Fig. 7 illustrates the results of chili fruit detection using the YOLOv8 model, with each detected chili enclosed in a red bounding box labeled "Chilli" along with a confidence score. The confidence scores, such as 0.88, 0.57, and 0.49, represent the model's certainty about each detection, with higher values indicating greater confidence in the accuracy of the prediction. The model effectively identifies and localizes chili fruits of various sizes and orientations, even amidst the plant's dense foliage. This demonstrates its robustness in detecting objects under real-world conditions. However, some detections with lower confidence scores (e.g., 0.29) suggest areas where the model may benefit from refinement, such as threshold adjustment or additional training data. The figure also highlights the model's ability to handle variability in detection, including overlapping and occluded objects, although these conditions could impact overall performance. This outcome underscores the potential of the YOLOv8 model for agricultural applications, particularly in tasks like chili fruit monitoring and yield estimation.

(a)

(b)

Fig. 7. Prediction results of Chili image (a) Chili video (b)

IV. Conclusion

To help the development in agriculture, specifically to detect chili fruit, a real-time and accurate object detection method using the YOLOv8 target detection network is proposed. This chili fruit detection can be developed and applied in various matters related to the growth and development of chili plants. There are 2077 data used as datasets and divided into 94% train data, 4% validation data, and 2% test data. The process of labeling the training data uses the Roboflow platform. The results showed the detection performance value of YOLOv8 with mAP 75.4%, F1 score 77.21%, Precision 74.7%, Recall 79.9%, and data processing speed 6.9ms per image. The value of indicators with an average value in the 70s is caused by very few iterations in the training process, accordingly, reducing the performance in detecting chili fruit objects.

Acknowledgment

This research was supported by the Department of Electrical Engineering, Faculty of Mathematics and Sciences (Padjadjaran University), Institut Teknologi Bandung, and Universitas Sumatera Utara, Indonesia and funded by Indonesian Collaborative Research (*Riset Kolaborasi Indonesia (RKI)*) program.

References

[1] K. Jiang, et al., "An Attention Mechanism-Improved YOLOv7 Object Detection Algorithm for Hemp Duck Count Estimation," *Agriculture* (Switzerland), vol. 12, no. 10, Oct. 2022. doi: 10.3390/agriculture12101659.

[2] "Consumption Expenditure of the Indonesian Population," March 2022. Coordinating Ministry of Economic Affairs, "Developing Agriculture Sector Resilience, Indonesia Receives Award from International Rice Research Institute," Republic of Indonesia. [Online] https://www.ekon.go.id/

[3] "Indonesia Labor Market Indicators," August 2022.

[4] A. I. B. Parico and T. Ahamed, "Real time pear fruit detection and counting using yolov4 models and deep sort," *Sensors*, vol. 21, no. 14, Jul. 2021. doi: 10.3390/s21144803.

[5] G. Sambasivam and G. D. Opiyo, “A predictive machine learning application in agriculture: Cassava disease detection and classification with imbalanced dataset using convolutional neural networks,” *Egyptian Informatics Journal*, vol. 22, no. 1, pp. 27-34, Mar. 2021. doi: 10.1016/j.eij.2020.02.007.

[6] A. Turnip et al, “Autonomous drone technology based on random forest classifier for revolutionizing agriculture,” in *Proc. 2024 IEEE Int. Conf. Artif. Intell. Mechatron. Syst. (AIMS)*, Bandung, Indonesia, 2024, pp. 1–5, doi: 10.1109/AIMS61812.2024.10512467.

[7] A. Turnip et al, “Sprayer system on autonomous farming drone based on decision tree,” in *Proc. 2024 IEEE Int. Conf. Artif. Intell. Mechatron. Syst. (AIMS)*, Bandung, Indonesia, 2024, pp. 1–6, doi: 10.1109/AIMS61812.2024.10512852.

[8] A. Turnip et al., “Optimization of smart farming irrigation using IoT-based artificial neural network,” in *Proc. 2024 IEEE Int. Conf. Artif. Intell. Mechatron. Syst. (AIMS)*, Bandung, Indonesia, 2024, pp. 1–5, doi: 10.1109/AIMS61812.2024.10513321.

[9] A. Turnip et al, “Control valve discharge prediction using multiple linear regression for smart precision farming,” in *Proc. 2024 IEEE Int. Conf. Artif. Intell. Mechatron. Syst. (AIMS)*, Bandung, Indonesia, 2024, pp. 1–5, doi: 10.1109/AIMS61812.2024.10512998.

[10] B. Xu, et al., “Automated cattle counting using Mask R-CNN in quadcopter vision system,” *Comput. Electron. Agric*., vol. 171, Apr. 2020. doi: 10.1016/j.compag.2020.105300.

[11] N. Xu, “Image processing technology in agriculture,” in *Journal of Physics: Conference Series* (Vol. 1881, No. 3, p. 032097), April, 2021, IOP Publishing.

[12] J. Liu and X. Wang, “Early recognition of tomato gray leaf spot disease based on MobileNetv2-YOLOv3 model,” *Plant Methods*, vol. 16, no. 1, Jun. 2020. doi: 10.1186/s13007-020-00624-2.

[13] S. Wan and S. Goudos, “Faster R-CNN for multi-class fruit detection using a robotic vision system,” *Comput. Netw.*, vol. 168, Feb. 2020. doi: 10.1016/j.comnet.2019.107036.

[14] J. Zhao, et al., “A wheat spike detection method in UAV images based on improved YOLOv5,” *Remote Sens.* (Basel), vol. 13, no. 16, Aug. 2021. doi: 10.3390/rs13163095.

[15] G. Li, et al., “Real-time detection of kiwifruit flower and bud simultaneously in orchard using YOLOv4 for robotic pollination,” *Comput. Electron. Agric.*, vol. 193, no. 106641, 2022.

[16] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, “YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors,” Jul. 2022. [Online]. Available: http://arxiv.org/abs/2207.02696

[17] H. S. Gill, et al., “Fruit recognition from images using deep learning applications,” *Multimed. Tools Appl.*, vol. 81, no. 23, pp. 33269-33290, 2022. doi: 10.1007/s11042-022-12868-2.

[18] W. Zhang, et al., “Imperfect wheat grain recognition combined with an attention mechanism and residual network,” *Appl. Sci.* (Switzerland), vol. 11, no. 11, Jun. 2021. doi: 10.3390/app11115139.

[19] J. He, W. Xiaohong, X. He, J. Hu, and L. Qin, “Imperfect wheat kernel recognition combined with image enhancement and conventional neural network,” *J. Comput. Appl.*, vol. 2021, no. 3, pp. 911-916.

[20] K. Cai, et al., “A modified YOLOv3 model for fish detection based on MobileNetv1 as backbone,” *Aquac. Eng.*, vol. 91, Nov. 2020. doi: 10.1016/j.aquaeng.2020.102117.

[21] M. Hussain, et al., “Domain Feature Mapping with YOLOv7 for Automated Edge-Based Pallet Racking Inspections,” *Sensors*, vol. 22, no. 18, Sep. 2022. doi: 10.3390/s22186927.

[22] L. Cao, et al., “The Semantic Segmentation of Standing Tree Images Based on the Yolo V7 Deep Learning Algorithm,” *Electronics* (Switzerland), vol. 12, no. 4, Feb. 2023. doi: 10.3390/electronics12040929.

[23] J. Chen, et al., “A Multiscale Lightweight and Efficient Model Based on YOLOv7: Applied to Citrus Orchard,” *Plants*, vol. 11, no. 23, Dec. 2022. doi: 10.3390/plants11233260.

[24] D. Wu, et al., “Detection of Camellia oleifera Fruit in Complex Scenes by Using YOLOv7 and Data Augmentation,” *Appl. Sci.* (Switzerland), vol. 12, no. 22, Nov. 2022. doi: 10.3390/app122211318.