Comparative Study of YOLOv5, YOLOv7 and YOLOv8 for Robust Outdoor Detection

Ryan Satria Wijaya[1*], Santonius[1], Anugerah Wibisana[1], Eko Rudiawan Jamzuri[1], dan Mochamad Ari Bagus Nugroho[2]

[1]Jurusan Teknik Elektro, Politeknik Negeri Batam, Batam, Indonesia
[2]Departemen Teknik Elektro, Politeknik Elektronika Negeri Surabaya, Surabaya, Indonesia

[*]Email: ryan@polibatam.ac.id

Abstract—**Object detection is one of the most popular applications among young people, especially among millennials and generation Z. The use of object detection has become widespread in various aspects of daily life, such as face recognition, traffic management, and autonomous vehicles. The use of object detection has expanded in various aspects of daily life, such as face recognition, traffic management, and autonomous vehicles. To perform object detection, large and complex datasets are required. Therefore, this research addresses what object detection algorithms are suitable for object detection. In this research, i will compare the performance of several algorithms that are popular among young people, such as YOLOv5, YOLOv7, and YOLOv8 models. By conducting several Experiment Results such as Detection Results, Distance Traveled Experiment Results, Confusion Matrix, and Experiment Results on Validation Dataset, I aim to provide insight into the advantages and disadvantages of these algorithms. This comparison will help young researchers choose the most suitable algorithm for their object detection task.**

Keywords: Deep Learning algorithms, Object detection, YOLOv5, YOLOv7, YOLOv8

I. INTRODUCTION

YOLO model has demonstrated great versatility, being used in a variety of scenarios such as aerial imaging, autonomous driving, medical imaging, and agricultural monitoring. For example, research by compared YOLOv5 and YOLOv8 for human detection in aerial images, highlighting the improved accuracy and efficiency of YOLOv8 in challenging environments [1]. Likewise, demonstrated superiority of YOLOv8 over YOLOv5 in American Sign Language detection, indicating the model's potential in facilitating communication for the hearing impaired [2]. The applications of YOLO models also extend to intelligent transport systems. evaluated YOLOv5 and YOLOv8 for vehicle and license plate detection, finding that YOLOv8 was more precise and faster. Additionally, [3], explored the use of YOLOv8 for accurate face mask classification, enhancing COVID-19 safety measures [4].

Significant improvements to YOLO models have been the focus of various studies. introduced TPH-YOLOv5, which incorporates a Transformer Prediction Head for better object detection in drone-captured scenarios [5]. Developed an improved YOLOv5 model for plant disease recognition, achieving high accuracy in agricultural applications [6]. In the realm of fruit detection, used an enhanced YOLOv5 model for litchi fruit detection, facilitating precise yield estimation [7]. Beyond agriculture, devised a real-time algorithm for detecting kiwifruit defects using YOLOv5, demonstrating the model's effectiveness in quality control [8]. improved YOLOv5 for real-time object detection in vehicle-mounted camera scenarios, which is crucial for autonomous driving technologies [9].

The adaptability of YOLO models is further evidenced by their application in diverse regions and contexts. analyzed YOLO models for vehicle recognition in South Asia, demonstrating their effectiveness in regional vehicle detection [10]. Similarly, focused on improving small object detection in autonomous vehicles with YOLO-Z, a variation of YOLOv5 [11]. In the domain of UAV applications, introduced UAV-YOLOv8 for small object detection in aerial photography, achieving significant improvements [7]. applied YOLOv8 for brake light status detection, enhancing road safety systems [12]. used YOLOv8 for fruit ripeness identification, aiding agricultural productivity [13], while developed a lightweight YOLOv8 algorithm for tomato detection, combining feature enhancement and attention mechanisms to improve performance [14].

Innovations in retail and road safety have also benefited from YOLOv8's capabilities [15]. enhanced retail checkout processes using YOLOv8 and DeepSort tracking, and [7] developed BL-YOLOv8 for improved road defect detection. The evolution of YOLO models has been comprehensively reviewed by detailing their impact on digital manufacturing and industrial defect detection. In the medical field [16], leveraged YOLOv8 for fracture detection in pediatric X-ray images, demonstrating the model's clinical applicability [17]. Combined YOLOv8 with advanced segmentation models for multimodal medical imaging, further highlighting its versatility and precision [18].

These diverse applications and continuous improvements of YOLO models, particularly YOLOv5, YOLOv7, and YOLOv8, underscore their pivotal role in advancing object detection technologies across various industries. The research compiled here provides a comprehensive overview of how YOLO models are being tailored and optimized to meet specific needs, enhancing accuracy, efficiency, and applicability in real-world scenarios.

II. METHOD

If in this study, we adopted two object detection methods that have proven to be effective, namely YOLOv5, YOLOv7 and YOLOv8. This method is commonly utilized in a wide array of fields and has its own advantages in object detection

A. Network Architecture YOLO

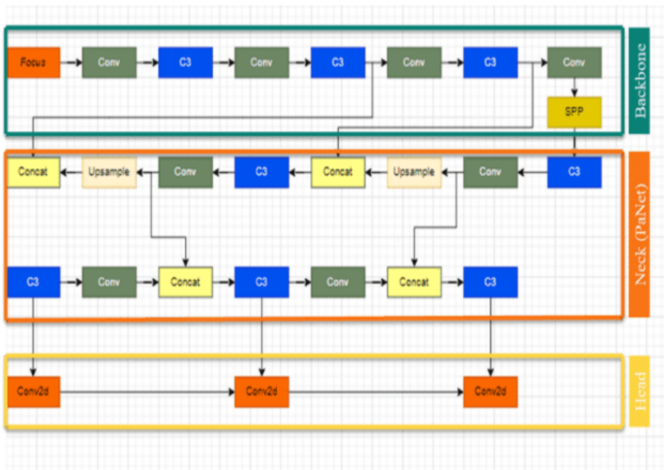

Fig. 1. Network Architecture YOLO

Figure 1 shows the YOLO Network Architecture in three main parts, namely:

1). Backbone

Backbone is the part of the network that is responsible for extracting features from images. In YOLOv5, *Efficient Net* is used as the backbone for feature extraction. *Efficient Net* is combined with several convolution layers and pooling layers to produce better features.

2). Necks

Necks is part of the network that connects the backbone with the head. In YOLO, the neck uses Spatial Pyramid Pooling which functions to extract more detailed features at each image scale.

3). Head

Head is the part of the network that is responsible for predicting objects in images. In YOLO, the head uses multiple convolution layers and nonlinear layers to produce an output in tensor form with the same size for each grid in the image. Each output tensor will be processed to determine the location, size, and class of objects in the image.

B. Alternative Research Methodologies

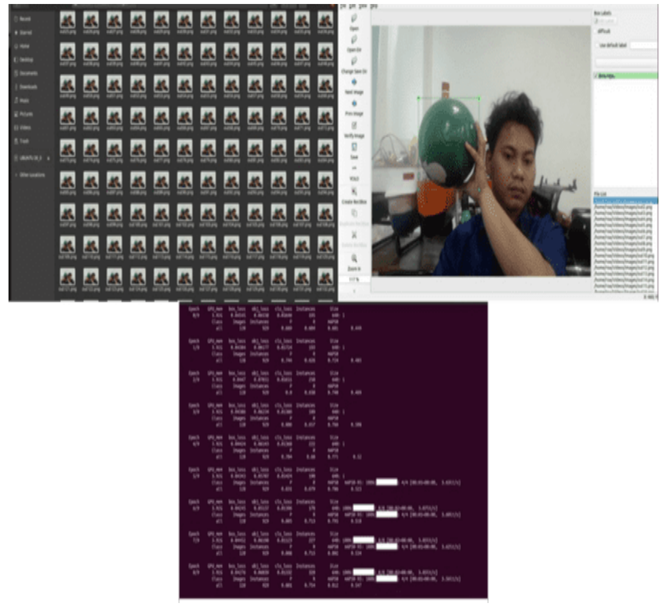

Fig. 2. Dataset, Label, Train

Figure 2 shows the Alternative Research Methodologies in three main parts, namely:

1). Dataset

The dataset used in this study compares the performance of YOLOv5, YOLOv7, and YOLOv8 algorithms. This dataset consists of 2671 Red Ball and Green Ball images, specially selected for YOLOv5, YOLOv7 and YOLOv8, along with their corresponding object labels, other journals used only 757 images [19].These images serve as input for each object detection algorithm. A performance assessment measure is used to compare the efficacy of the two algorithms. The dataset plays an important role in ensuring the accuracy and reliability of the performance comparison results between YOLOv5, YOLOv7, and YOLOv8.

2). Labeling the Dataset

After collecting the dataset, the next phase requires data annotations to be prepared. Annotation refers to the process of assigning labels or categories to data in a data set, facilitating the training and testing of deep learning models.

3). Model Training

Following the dataset preparation, the subsequent step entails creating deep-learning models for Yolov5, YOLOv7 and YOLOv8. These models are then trained using the prepared data. The model training process involves multiple iterations, including inputting data into the model, examining the resulting output, evaluating model performance, and optimizing parameters to enhance the model's effectiveness.

C. Object Area Calculation

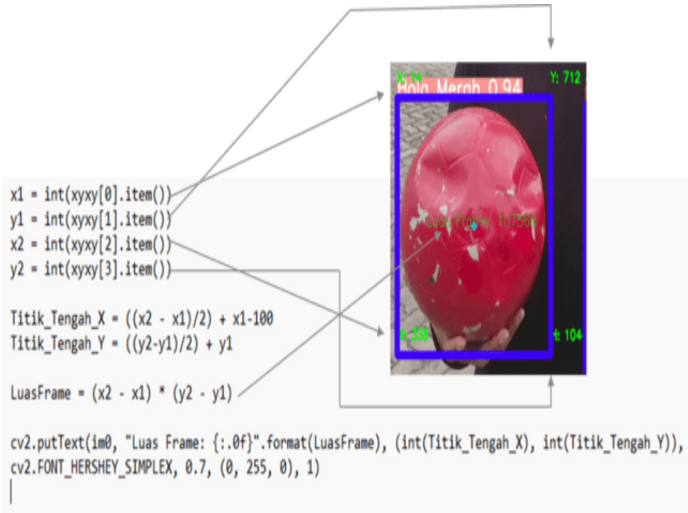

Fig. 3. Object Area Calculation

In Figure 3, we can see the Object Area Calculation with the following explanation:

$$Square\ Area = (x2 - x1) * (y2 - y1) \quad (1)$$

- (x21 - x1) calculates the length of the object on the horizontal (x) axis. It is the difference between the x coordinate of the lower right corner point and the x coordinate of the upper left corner point.
- (y21 - y1) calculates the height of the object on the vertical (y) axis. This is the difference between the y coordinate of the lower right corner point and the y coordinate of the upper left corner point.
- The two length and height calculations are multiplied together. The result of this multiplication is the area of the object.
- Titik_Tengah_X: Function to make the X axis stay in the center position
- Titik_Tengah_Y: Function to make the Y axis remain in the center position
- cv2.putText serves to display the result of "LuasFrame"

III. RESULTS AND DISCUSSION

In this chapter, we present the results of the analysis. The results can be presented in the form of figures and tables that the reader can easily understand. In the discussion section, there are some important points that need to be identified for more accurate object detection. These can be critical variables such as object parameters, brightness level and observation distance. This allows the reader to better understand the results of the analysis and the influence of important factors in object detection.

A. Detection Results

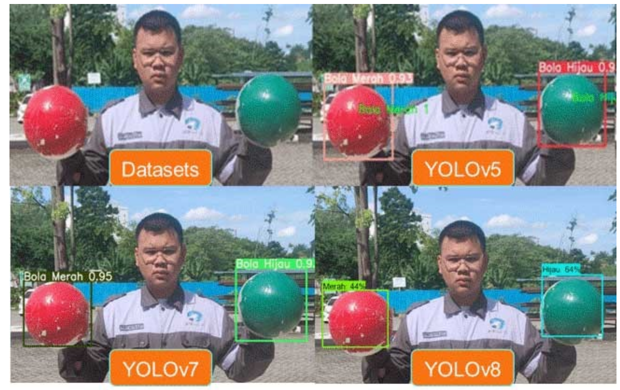

Fig. 4. Dataset, YOLOv5, YOLOv7, YOLOv8

In Figure 4, we can see the object detection results of the YOLOv5, YOLOv7, and YOLOv8 algorithms, as shown. The image can be detected because of the previous steps, namely, preparing the dataset, labeling the dataset, then training.

B. Experiment results Mileage

TABLE I
MILEAGE COMPARISON YOLOV5

Distance	Perfectly Detected	Imperfectly Detected	Not detected
1 (M)	✓		
2 (M)	✓		
3 (M)	✓		
4 (M)	✓		
5 (M)	✓		
6 (M)	✓		
7 (M)	✓		
8 (M)			✓
>8 (M)			✓

Based on the experimental results obtained from the YOLOv5 algorithm in Table 1, it can be concluded that: In the range of 1 meter to 7 meters, the YOLOv5 algorithm successfully detects objects perfectly. However, at a distance of 8 meters and greater than 8 meters, the YOLOv5 algorithm is unable to detect objects.

TABLE II
MILEAGE COMPARISON YOLOV7

Distance	Perfectly Detected	Imperfectly Detected	Not detected
1 (M)	✓		
2 (M)	✓		
3 (M)	✓		
4 (M)	✓		
5 (M)	✓		
6 (M)	✓		
7 (M)	✓		
8 (M)		✓	
>8 (M)			✓

Based on the results shown in Table 2, we conclude that the YOLOv7 algorithm detects objects perfectly within the range of 1 to 7 meters. At a distance of 8 meters, object detection by the YOLOv7 algorithm becomes less accurate, and at distances of 8 meters and above, it fails to detect objects. Further analysis indicates that the decrease in detection accuracy at longer distances may be attributed to several factors, including reduced image resolution, environmental interference, and limitations of the camera sensor used. Therefore, for applications requiring object detection at greater distances, it is necessary to consider using additional algorithms or more advanced hardware to ensure consistent accuracy.

TABLE III
MILEAGE COMPARISON YOLOV8

Distance	Perfectly Detected	Imperfectly Detected	Not detected
1 (M)	✓		
2 (M)	✓		
3 (M)	✓		
4 (M)	✓		
5 (M)	✓		
6 (M)	✓		
7 (M)	✓		
8 (M)		✓	
>8 (M)			✓

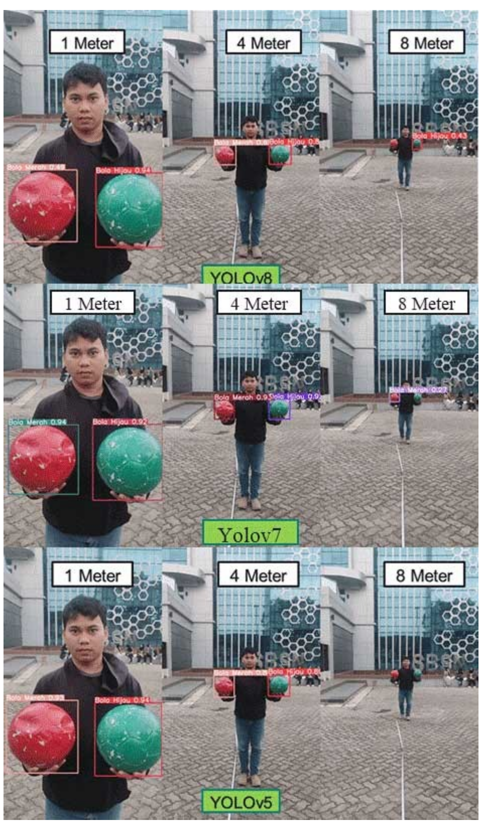

Fig. 5. Object detection distance of YOLOv8, YOLOv7, and YOLOv5

Based on the findings presented in Table 3, the YOLOv7 algorithm demonstrates precise object detection within the 1 to 7 meters range. However, its accuracy diminishes beyond 8 meters, leading to a failure in object detection. From Figure 5, it can be seen that YOLOv5 can detect objects perfectly in the distance range of 1 to 7 meters. However, when the distance reaches 8 meters and more than 8 meters, YOLOv5 cannot detect objects. On the other hand, YOLOv7 and YOLOv8 can detect objects at a distance range of 1 to 7 meters perfectly, at a distance of 8 meters YOLOv7 and YOLOv8 can detect objects although the detection is not perfect and at a distance of 8 meters YOLOv7 and YOLOv8 cannot detect objects.

C. *Confusion Matrix*

Confusion Matrix is a method used to evaluate the performance of algorithms or classifiers, including deep learning algorithms such as YOLOv5, YOLOv7 and YOLOv8, which uses a matrix to assess the model's performance in object detection tasks. To measure object detection performance, there are three outcomes, namely.

1) *Precision - Confidence Curve*

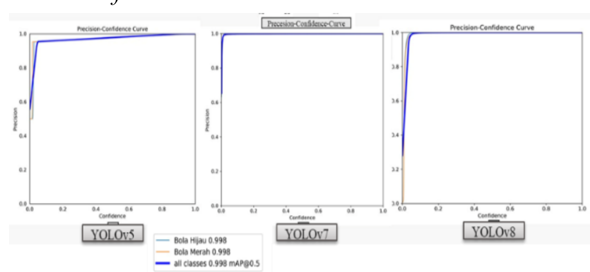

Fig. 6. Precision - Confidence Curve

Figure 6 shows that the Precision-Confidence Curve for YOLOv8 is better than those for YOLOv7 and YOLOv5. This can be seen from the figure where YOLOv8 starts from a confidence of 0.1 and maintains a high precision close to 1, while YOLOv7 starts at confidence 0.6 and YOLOv5 starts from confidence 0.5, both also maintaining high precision close to 1. This demonstrates that YOLOv8 is able to maintain high precision even at lower confidence levels, indicating that this model is more robust and reliable across various confidence levels compared to YOLOv7 and YOLOv5.

2) *Precision-Recall Curve*

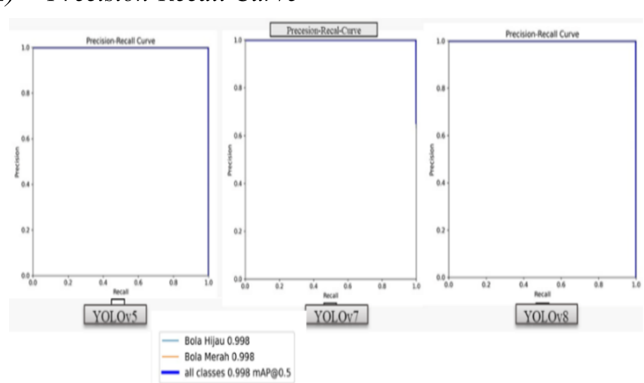

Fig. 7. Precision - Recall Curve

Figure 7 shows that the Precision-Recall Curves for YOLOv8 and YOLOv5 are better than that for YOLOv7. This is evident from the figure where YOLOv8 and YOLOv5 start from high recall values and maintain high precision close to 1, while YOLOv7 also starts from a high recall value and maintains precision, but its performance is not as good as YOLOv8 and YOLOv5.

3) *Recall-Confidence Curve*

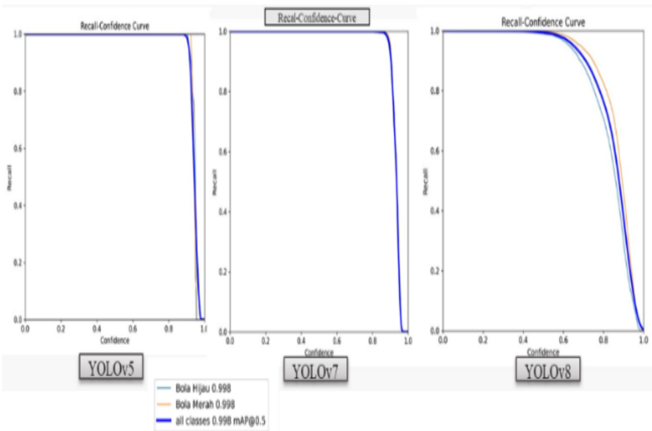

Fig. 8. Recall - Confidance Curve

Figure 8 shows that the Confidence Curve for YOLOv8 is better than YOLOv7 and YOLOv5, as seen from the curve where YOLOv8 starts from a high recall value of 1 and maintains a high confidence close to 1, with the graph showing the curve immediately curving to confidence 1 at confidence 0.5. In contrast, YOLOv7 and YOLOv5 also start from a high recall value of 1 and maintain a high confidence close to 1, but their curves only curve towards confidence 1 at confidence 0.9. This shows that YOLOv8 performs better as it is able to maintain high confidence at lower confidence levels than YOLOv7 and YOLOv5.

D. *Object Detection Results in various conditions*

TABLE IV
COMPARISON OF CONDITIONS YOLOV5

Condition	Perfectly Detected	Imperfectly Detected	Not detected
Bright	✔		
Dim	✔		
Dark		✔	

Table 4 shows the comparison of bright, dim and dark conditions using the YOLOv5 algorithm. The comparison above shows that in bright and dim conditions the YOLOv5 algorithm successfully detects objects perfectly. while in dark conditions the YOLOv5 algorithm successfully detects objects but the detection is less perfect.

TABLE V
COMPARISON OF CONDITIONS YOLOV7

Condition	Perfectly Detected	Imperfectly Detected	Not detected
Bright	✔		
Dim	✔		
Dark	✔		

Table 5 shows a comparison of bright, dim and dark conditions using the YOLOv7 algorithm. The comparison above shows that in bright, dim, and dark conditions the YOLOv7 algorithm successfully detects objects perfectly.

TABLE VI
COMPARISON OF CONDITIONS YOLOV8

Condition	Perfectly Detected	Imperfectly Detected	Not detected
Bright	✔		
Dim		✔	
Dark			✔

Table 6 shows the comparison of bright, dim and dark conditions using the YOLOv8 algorithm. The comparison above shows that in bright and dim conditions the YOLOv8 algorithm successfully detects objects perfectly. while in dark conditions the YOLOv8 algorithm successfully detects objects but the detection is less perfect.

A summary of the three tables in Figure 9 shows that YOLOv5 perfectly detects objects in bright and dim conditions, but the detection is less perfect in dark conditions. YOLOv7 perfectly detects objects in all conditions, whether bright, dim, or dark. While YOLOv8 perfectly detects objects in bright and dim conditions, but the detection is less perfect in dark conditions. From these results, it can be concluded that YOLOv7 has the best performance in detecting objects in various conditions, followed by YOLOv8, and YOLOv5 has the lowest performance.

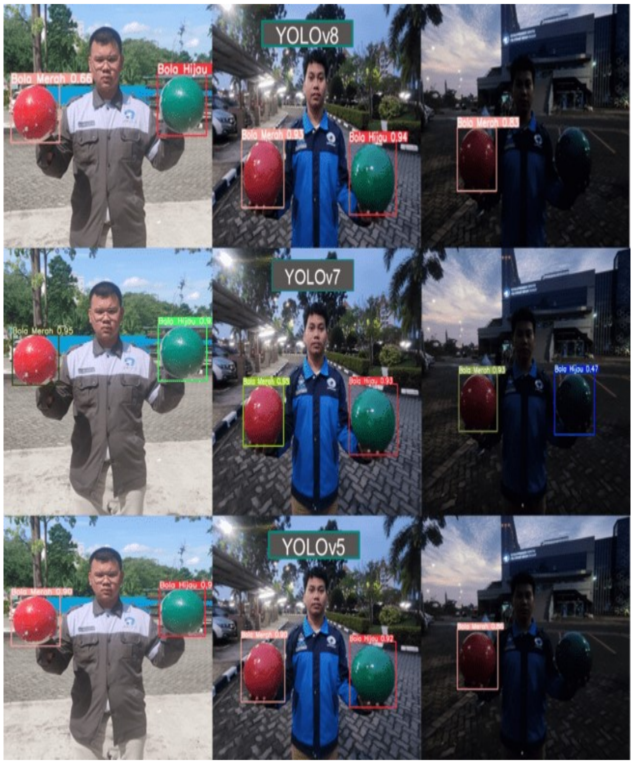

Fig. 9. Condition comparison between YOLOv5, YOLOv7 and YOLOv8 algorithms

E. Experimental Results on Validation Dataset

TABLE VII
EXPERIMENTAL RESULT ON VALIDATION DATASET

Model	Images	Precision	Recal	mAP50	mAP50-95
YOLOv8	3152	1	0,694	0,688	0,558
YOLOv7	3152	1	1	0,693	00.09
YOLOv5	3152	0,672	0,674	0,691	0,517

Based on the experimental results on the validation datasets obtained from the three algorithms in Table 7, it can be concluded that: In different models using the same dataset images, the values of Precision, Recall, mAP50, and mAP50-95 for YOLOv8 are higher than those of YOLOv5 and YOLOv7. Hence, it can be concluded that the YOLOv8 algorithm is superior in the experimental results on the validation dataset.

IV. CONCLUSION

From the results of this study, it can be concluded that YOLOv5, YOLOv7, and YOLOv8 are very useful deep learning algorithms for implementing real-time computer vision applications. Experimental results on travel distance, object recognition results under different conditions, and experimental results on validation datasets show that YOLOv7 and YOLOv8 are superior compared to YOLOv5 in terms of object detection. However, the computation of the YOLOv5 algorithm is lighter compared to the YOLOv7 and YOLOv8 algorithms. The limitations of this study include the limited dataset used and the variety of test conditions that may not cover all possible scenarios in the real world. In addition, external factors such as light environment and weather conditions may affect the performance of object detection algorithms. The results of this study provide valuable insights into the performance comparison between YOLOv5, YOLOv7, and YOLOv8 in object detection. The main contribution of this research is to present a better understanding of the advantages and disadvantages of each algorithm, so that researchers and practitioners can choose the most suitable algorithm for their applications. For future research, it is recommended to expand the testing dataset to cover a wider variety of conditions and objects. In addition, further research can be conducted to improve the accuracy and speed of object detection by integrating new technologies or adjusting the parameters of existing algorithms.

REFERENCES

[1] I. Purwita Sary, E. Ucok Armin, S. Andromeda, E. Engineering, and U. Singaperbangsa Karawang, "Performance Comparison of YOLOv5 and YOLOv8 Architectures in Human Detection Using Aerial Images," *Ultima Computing : Jurnal Sistem Komputer*, vol. 15, no. 1, 2023.

[2] S. Tyagi, P. Upadhyay, H. Fatima, S. Jain, I. Avinash, and K. Sharma, "American Sign Language Detection using YOLOv5 and YOLOv8," 2023, doi: 10.21203/rs.3.rs-3126918/v1.

[3] M. Henrique Fonseca Afonso *et al.*, "Vehicle and Plate Detection for Intelligent Transport Systems: Performance Evaluation of Models YOLOv5 and YOLOv8", doi: 10.13140/RG.2.2.11022.95042.

[4] S. Tamang, B. Sen, A. Pradhan, K. Sharma, and V. K. Singh, "International Journal of Intelligent Systems And Applications In Engineering Enhancing COVID-19 Safety: Exploring YOLOv8 Object Detection for Accurate Face Mask Classification." [Online]. Available: www.ijisae.org

[5] X. Zhu, S. Lyu, X. Wang, and Q. Zhao, "TPH-YOLOv5: Improved YOLOv5 Based on Transformer Prediction Head for Object Detection on Drone-captured Scenarios," Aug. 2021, [Online]. Available: http://arxiv.org/abs/2108.11539

[6] Z. Chen *et al.*, "Plant Disease Recognition Model Based on Improved YOLOv5," *Agronomy*, vol. 12, no. 2, Feb. 2022, doi: 10.3390/agronomy12020365.

[7] X. Wang, H. Gao, Z. Jia, and Z. Li, "BL-YOLOv8: An Improved Road Defect Detection Model Based on YOLOv8," *Sensors (Basel)*, vol. 23, no. 20, Oct. 2023, doi: 10.3390/s23208361.

[8] J. Yao, J. Qi, J. Zhang, H. Shao, J. Yang, and X. Li, "A real-time detection algorithm for kiwifruit defects based on yolov5," *Electronics (Switzerland)*, vol. 10, no. 14, Jul. 2021, doi: 10.3390/electronics10141711.

[9] Z. Ren, H. Zhang, and Z. Li, "Improved YOLOv5 Network for Real-Time Object Detection in Vehicle-Mounted Camera Capture Scenarios," *Sensors*, vol. 23, no. 10, May 2023, doi: 10.3390/s23104589.

[10] M. Mahmud Rafi *et al.*, "Performance Analysis of Deep Learning YOLO Models for South Asian Regional Vehicle Recognition." [Online]. Available: www.ijacsa.thesai.org

[11] A. Benjumea, I. Teeti, F. Cuzzolin, and A. Bradley, "YOLO-Z: Improving small object detection in YOLOv5 for autonomous vehicles," Dec. 2021, [Online]. Available: http://arxiv.org/abs/2112.11798

[12] G. Oh and S. Lim, "One-Stage Brake Light Status Detection Based on YOLOv8," *Sensors*, vol. 23, no. 17, Sep. 2023, doi: 10.3390/s23177436.

[13] B. Xiao, M. Nguyen, and W. Q. Yan, "Fruit ripeness identification using YOLOv8 model," *Multimed Tools Appl*, vol. 83, no. 9, pp. 28039–28056, Mar. 2024, doi: 10.1007/s11042-023-16570-9.

[14] G. Yang, J. Wang, Z. Nie, H. Yang, and S. Yu, "A Lightweight YOLOv8 Tomato Detection Algorithm Combining Feature Enhancement and Attention," *Agronomy*, vol. 13, no. 7, Jul. 2023, doi: 10.3390/agronomy13071824.

[15] A. Vats and D. C. Anastasiu, "Enhancing Retail Checkout through Video Inpainting, YOLOv8 Detection, and DeepSort Tracking," 2023. [Online]. Available: https://github.com/davidanastasiu/RetailCounter

[16] M. Hussain, "YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial

Defect Detection," *Machines*, vol. 11, no. 7. Multidisciplinary Digital Publishing Institute (MDPI), Jul. 01, 2023. doi: 10.3390/machines11070677.

[17] R.-Y. Ju and W. Cai, "Fracture Detection in Pediatric Wrist Trauma X-ray Images Using YOLOv8 Algorithm," Apr. 2023, [Online]. Available: http://arxiv.org/abs/2304.05071

[18] S. Pandey, K.-F. Chen, and E. B. Dam, "Comprehensive Multimodal Segmentation in Medical Imaging: Combining YOLOv8 with SAM and HQ-SAM Models," 2022.

[19] T. Li, Y. Gao, K. Wang, S. Guo, H. Liu, and H. Kang, "Diagnostic assessment of deep learning algorithms for diabetic retinopathy screening," *Inf Sci (N Y)*, vol. 501, pp. 511–522, Oct. 2019, doi: 10.1016/j.ins.2019.06.011.