Jurnal Polimesin
Department of Mechanical Engineering
State Polytechnic of Lhokseumawe
http://e-jurnal.pnl.ac.id/polimesin

e-ISSN	: 2549-1999	No.	: 3	Month	: June
p-ISSN	: 1693-5462	Volume	: 23	Year	: 2025

Processing dates: received on 2025-02-06, reviewed on 2025-04-23, accepted on 2025-05-10 and online availability on 2025-06-30

Development of a real-time excavator cycle time detection system using YOLO for mining operations: a company case study

Setia Abikusna*, Amadeus Renjiro Wang, Leo Setiawan, Nur Rofiq Syuhada, Randy Putra Afani
Department of Mechanical and Industrial Engineering, Polytechnic of Astra, Cikarang 17530, Indonesia
*Corresponding Author: setia.abikusna@polytechnic.astra.ac.id

Abstract

Productivity in mining operations is highly dependent on the efficiency of heavy equipment such as excavators. Conventional methods for measuring excavator cycle time are often manual and inefficient. This study aims to develop a real-time cycle time detection system using the You Only Look Once (YOLO) object detection algorithm. Video data from excavator operations were annotated to train the YOLO model, which was then integrated into a user-friendly application using OpenCV. The system achieved a mean Average Precision (mAP) of 94.2% and operated at 30 Frames Per Second (FPS), enabling accurate and real-time detection of excavator activities. The system enhanced monitoring efficiency and operational productivity. Its implementation in mining environments demonstrates the potential for automated cycle time analysis to support equipment management, improve safety, and reduce operational delays.

Keywords:
Object detection, cycle time, excavator, application, mining

1 Introduction

A company engaged in various industrial sectors, including mining. It sells heavy equipment, including excavators. In mining operations, excavators play a crucial role in excavation and material transfer activities. The efficiency of excavator operations depends on cycle time, which is the time required to complete a full work cycle. Optimal cycle time can increase productivity and reduce operational costs. Therefore, monitoring and analyzing cycle time in real time is very important. Under certain conditions, conventional excavator cycle time measurement often takes a long time. This is due to the long and very complex procedure.

One way to achieve this is through digital monitoring using object detection and image processing technology that can monitor the movement of excavator activities accurately and in real time. Vision-based methods allow visualization of equipment conditions directly from images and videos. This facilitates the identification of false recognition and analysis of the causes of low productivity. In contrast, non-vision-based methods have difficulty categorizing activities in detail and require the installation of sensors to monitor equipment [1], [2].

Currently, monitoring cycle time on excavators is often done manually, and the procedures are very time-consuming. With the advancement of technology, especially in the field of computer vision and machine learning, there is an opportunity to develop a more accurate object detection system that can be done in real time [3]. Object detection is a field of computer vision that aims to identify and locate objects of a certain class (such as humans, vehicles, etc.) in images or videos. Object detection algorithms include several stages, such as segmentation, feature extraction, and object classification. One of the algorithms that is often used in the field of object detection is You Only Look Once (YOLO), which is included in the category of Convolutional Neural Network (CNN) algorithms, and the way it works imitates the way the human brain works [4].

YOLO is one of the most popular and efficient object detection algorithms [5]–[7]. YOLO is known for its unique approach, where the entire image is analyzed only once, unlike other methods that perform detection by processing several parts of the image separately [8], [9]. YOLO combines detection and classification in one neural network that allows for a very fast and accurate detection process. In its development, YOLO has undergone several significant improvements, with the latest version being YOLOv8 [10]–[12]. YOLOv8 introduces various improvements in network architecture and optimization techniques, thus providing better performance in terms of speed and accuracy compared to previous versions [13]–[15]. YOLOv8 is the latest model in the YOLO algorithm series, the most well-known family of object detection and classification models in the field of Computer Vision [16]–[18].

The following are some of the advantages of the YOLOv8 method [19]–[21]:

a. High performance. YOLOv8 has better performance than previous versions, with faster and more accurate object detection.
b. Smaller model. The YOLOv8 model is smaller than previous versions, making it easier to use and implement.
c. Better feature extraction. YOLOv8 uses a better feature extraction method than previous versions, resulting in more accurate object detection.

Based on the above advantages, YOLOv8 offers better speed and accuracy. Compared to previous models, YOLOv5 stands out for its user-friendly nature, making it easier to use [22]. By developing an object detection system on the excavator cycle time in mining operations using the YOLO method, this system is expected to be able to monitor excavator activities accurately and in a shorter time, thereby helping to make more effective operational decisions. This study aims to develop a cycle time detection system application for real-time measurements. The method used is based on YOLO.

2 Research methodology

The research [23] shows that with YOLO implemented in a Small Medium Enterprise (SME), they can monitor visitor tracking systems. This system is to capture images and can classify them as human or non-human. The research [24] shows that the implementation of YOLO can improve work safety through the early detection of signs of fatigue in workers. With this system, it can recognize the potential for work accidents. This research methodology has several stages used to achieve the expected results, which can be seen in Fig. 1. The following are the stages [25]:

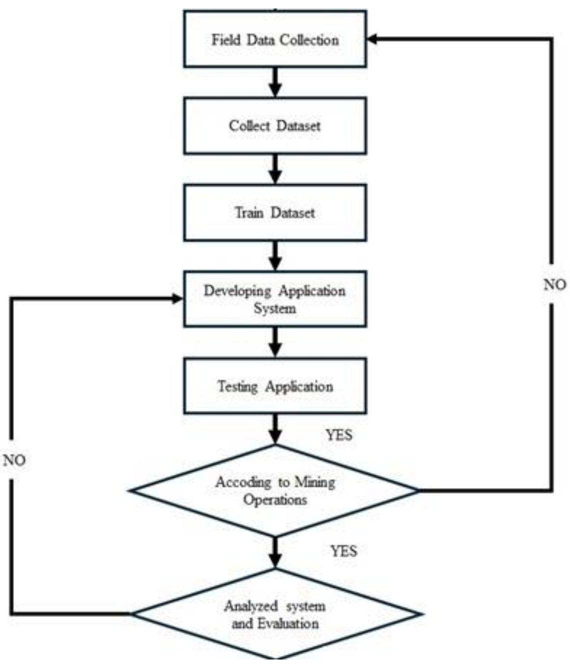

Fig. 1. Research methodology flowchart

2.1 Field data collection

Field data collection is an important step in this research to obtain the data needed for the development and implementation of the YOLO-based object detection system. The following are the steps taken in field data collection.

2.1.1 Research location

The research was conducted at the mining site of the company is located in a Regency. This location was chosen because it has high excavator operating activity and is experiencing a decline in productivity**,** so the data obtained can describe field conditions realistically.

2.1.2 Used

The equipment used includes high-resolution drone cameras flown at several strategic points around the excavator operation to record video of the excavator's productivity activities during the operation [26]. The collected data is stored in a computer and later processed using special software.

2.1.3 Data collection techniques

Data collection techniques involve direct observation to understand the excavator operation process and ensure that the drone camera is flown in an optimal position. During the observation, field notes were made to record important events or anomalies that might affect the data. Videos of excavator activity were recorded for several minutes and in different positions to obtain a representative data sample. Cycle time measurements were carried out using video and sensor data, where the start and end times of each operation cycle were recorded manually. In addition, interviews were conducted with the production team to gain insight into the factors that affect cycle time and to validate the data collected.

Each piece of data collected contains more than 100 pictures, but for this study, around 70% of all picture collections were used. This aims to ensure that the collected data represents all pictures. With this comprehensive field data collection method, the researchers are expected to produce accurate and reliable data for developing a YOLO-based cycle time detection system in mining operations.

2.2 Dataset collection

Dataset collection by identifying the specific needs of the object to be detected, in this case, an excavator [27]. The dataset covers various types of excavators, various types of positions, and various operating conditions. To get accurate results, a large and varied dataset is needed. The process of taking pictures and videos is carried out by taking data recorded in various lighting conditions and shooting angles to ensure that the model can detect excavators in various situations. Labeling is done using the Roboflow platform to annotate images. Each image is annotated with a bounding box that covers the excavator. Specific object classes are determined: “excavator”, “structure”, and various bucket classifications: “bucket-digging”, “bucket-dumping”, “bucket-empty”, and “bucket load” [28]. The following dataset annotation through Roboflow can be seen in Fig. 2.

Fig. 2. Dataset annotation through roboflow

Based on Fig.2, labeling quality is maintained to ensure all annotations are done accurately and consistently. From the annotation results, bad data, such as blurry or irrelevant images, is eliminated. The dataset is then divided into three parts: training set (70%), validation set (20%), and test set (10%) for proper model performance evaluation.

Labeling quality is maintained to ensure all annotations are done accurately and consistently. From the annotation results, bad data, such as blurry or irrelevant images, is eliminated. The dataset is then divided into three parts: training set (70%), validation set (20%), and test set (10%) for proper model performance evaluation.

2.3 YOLO model training

The data is formatted according to YOLO’s requirements, typically by creating a text file for each image that lists the bounding box coordinates and object class. The data is then augmented with rotation, cropping, and lighting changes to improve the variety and quality of the training data [29]. The training process is carried out using the YOLOv8s model with specific configurations, including 100 epochs and an image size of 640x640 pixels. Google Colab is used to utilize GPUs and accelerate the training process. During training, metrics such as loss and Mean Average Precision (mAP) are continuously monitored to ensure that the model experiences significant performance improvements [30], [31]. Final evaluations are performed on the validation set and test set to validate the model’s performance. If the evaluation results do not meet the expected standards, additional iterations are performed**,** focusing on improving annotations, adding data, or adjusting training parameters to achieve optimal results. The following metric variations with 100 epochs can be seen in Fig. 3.

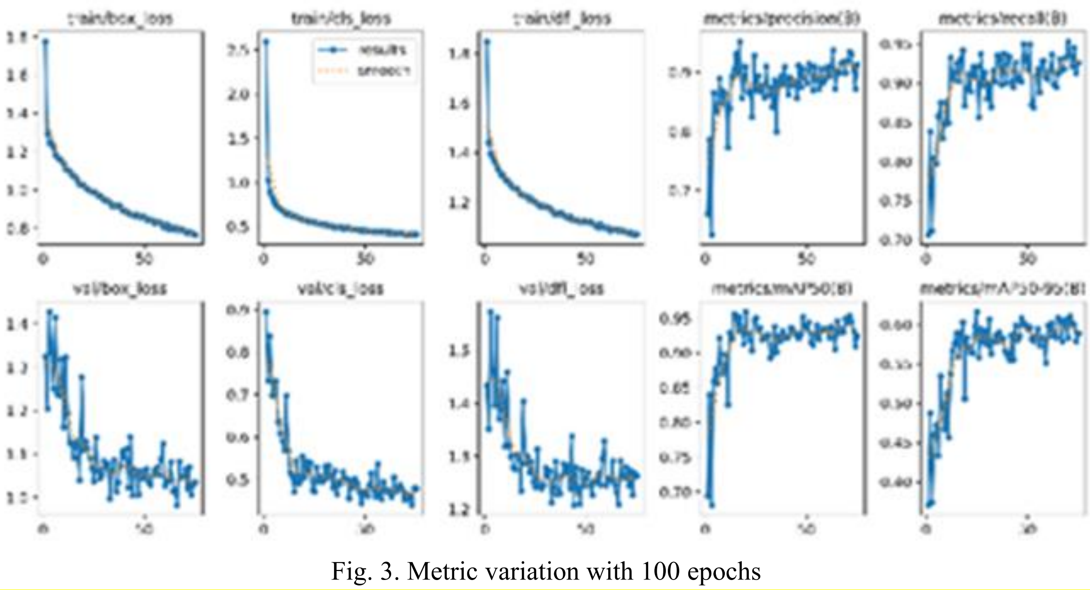

Fig. 3. Metric variation with 100 epochs

3 Results and discussion

3.1 Results

3.1.1 Development of the cycle time detection system application

Application development by implementing the YOLO algorithm into an excavator cycle time object detection system, built using the Python programming language and utilizing open-source libraries such as OpenCV and Pytorch to process images and videos in real time. The user interface is designed to be easy to use by users in the field, displaying object detection visually and providing information if the excavator is detected under certain conditions. The user interface of the excavator cycle time detection system application can be seen in Fig. 4.

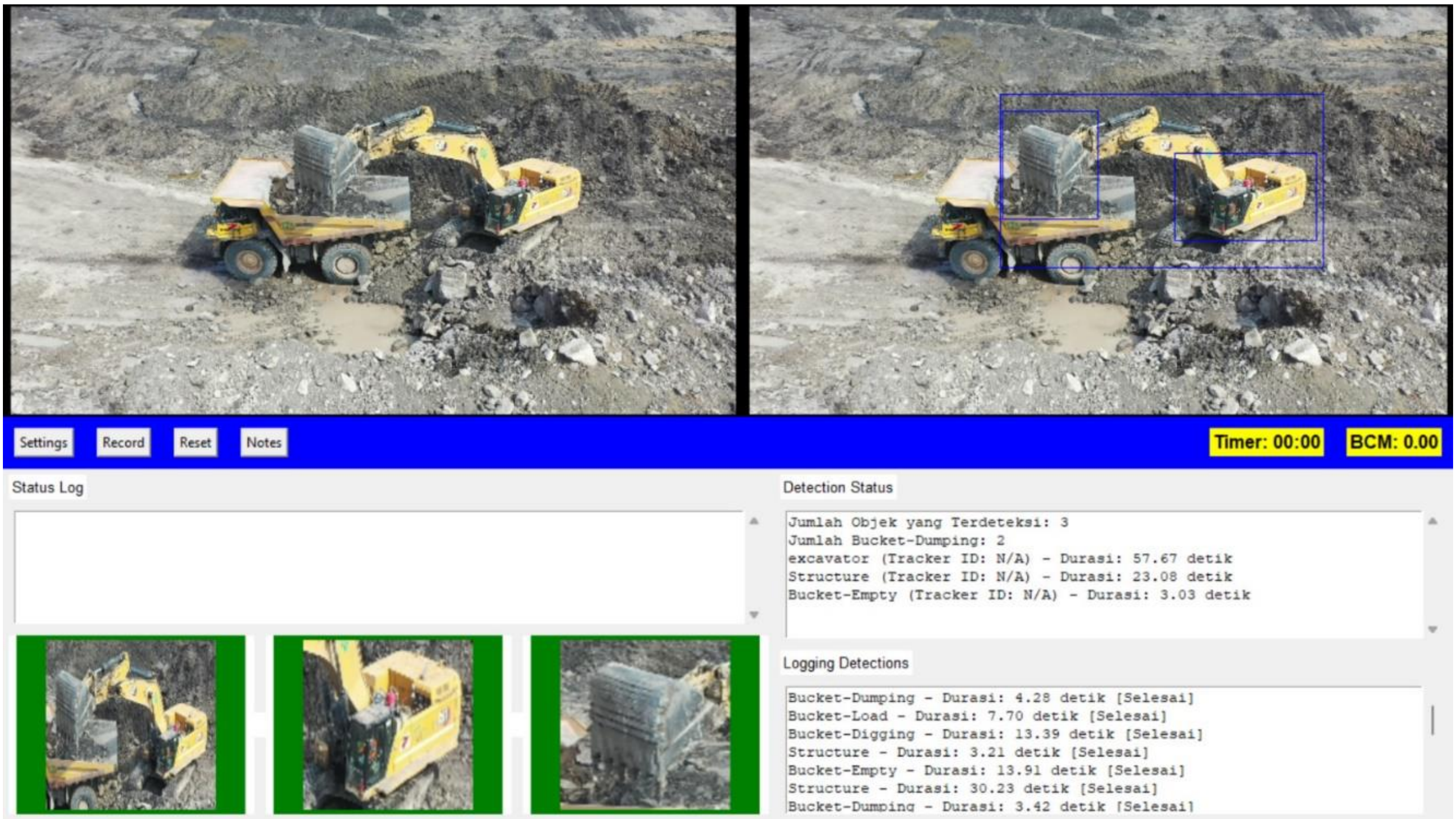

Fig. 4. User interface of the excavator cycle time detection system application

Based on the application displayed in Fig. 4, there are several features, including the following:

a. Real-time detection of excavator and bucket objects with various classifications (bucket-digging, bucket-dumping, bucket-empty, bucket-load);
b. Visualization of bounding boxes on the screen for each detected object;
c. Logging detection data for further analysis;
d. Setting the confidence of object detection and movement of detected objects; Displaying the Bank Cubic Meter (BCM) in real-time to monitor the volume of material moved;
e. Timer to measure and record the duration of excavator operations; and
f. Saving results in Excel format**,** which facilitates further analysis and reporting.

3.1.2 Field testing

Field-testing was conducted at an active mining site to test the app's performance under real-world conditions [32]. This involved using drone cameras connected directly to the app, as well as cameras mounted at strategic angles to capture excavator activity from various positions and lighting conditions. The DJI GO 4.0 app interface can be seen in Fig. 5.

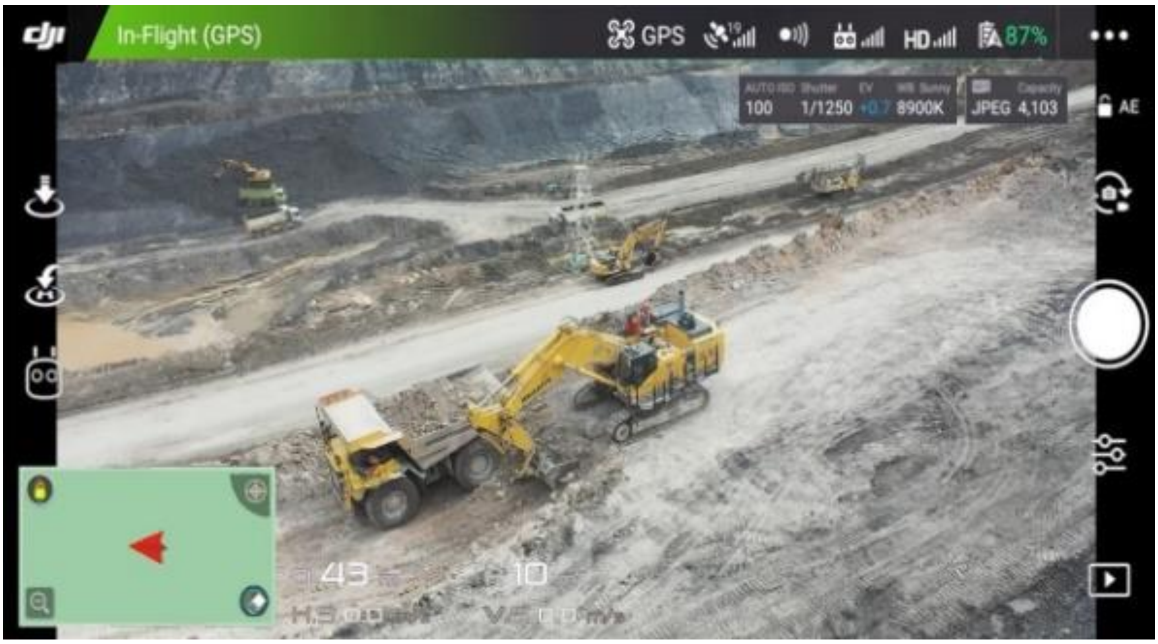

Fig. 5. DJI GO 4.0 App Interface

Based on Fig. 5, the data from the drone camera is used to provide a wide view and strategic angles of the excavator operation. The drone directly transmits image and video data to the application, which then performs real-time object detection and displays the results directly on the user interface.

During the field testing, various parameters were measured to evaluate the performance of the excavator cycle time detection system application. Detection accuracy was one of the key parameters, where the number and type of objects detected by the application were compared with manual records. This involved calculating the percentage of accuracy as well as identifying false positives and false negatives to evaluate how well the application could recognize objects under various conditions. Processing speed was also a major focus, measured by the time it took to detect an object from the time the data was received to the time the results were displayed. This measurement was important to ensure the application’s ability to work in real time and provide a quick response. In addition, the application’s consistency and stability were monitored by observing its performance under changing lighting conditions and its stability during continuous operation. This helped evaluate the application’s reliability in the face of varying operational environments. Lastly, ease of use was assessed based on feedback from operators using the application during the testing. These aspects include the level of user satisfaction, ease of interface navigation, and usability of the features available in the application. This assessment provided insight into the user experience and areas for further optimization.

The test was conducted using an excavator cycle time detection system application for 10 minutes, while a stopwatch was used to manually record the duration of excavator activity as comparative data.

3.2 Discussion

Performance evaluation is carried out by comparing the results obtained from the excavator cycle time detection system application with manual data recorded using a stopwatch.

3.2.1 Object detection accuracy

The evaluation of the accuracy of the object detection model is carried out to measure how well the model detects and classifies the desired objects, such as excavators, and various bucket classifications (bucket-digging, bucket-dumping, bucket-empty, bucket-load). The metrics used in this evaluation are Precision, Recall, and mAP [33].

mAP is the average value of Average Precision (AP) in a model. To measure the performance of object detection in the model, mAP can be calculated by limiting the IoU of the detection results. AP is obtained by calculating the AP value compared to the recall value in the range of 0 to 1 [34].

The developed object detection model is evaluated using an annotated dataset with bounding boxes and labels for each object class. Precision and Recall are calculated at various confidence score thresholds to generate Precision-Recall curves. The area under these curves is integrated to obtain the AP for each object class. The mAP is then calculated as the average of all APs. The results of the model's performance accuracy after improvements can be seen in Fig. 6

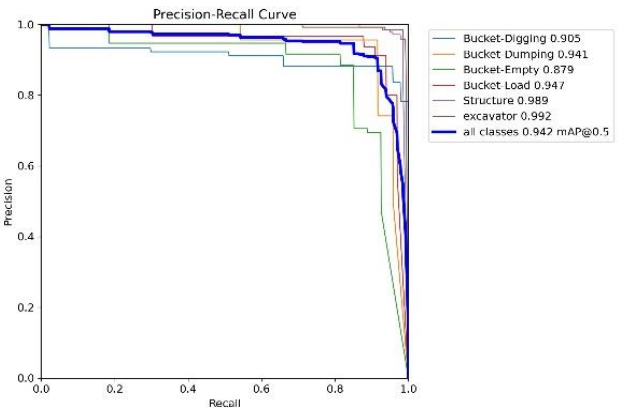

Fig. 6. Precision-recall curve for each object class

Based on the evaluation results from Fig. 6, the object detection model showed very good performance with a mAP value of 0.942 (94.2%). This means that the model can detect and classify objects with high accuracy. The AP value for each object class also shows that the model has consistent and good performance for various types of objects. High Precision and Recall values at various confidence score thresholds indicate that the model has good accuracy and sensitivity in detecting objects. Overall, the evaluation results show that the developed object detection model has high accuracy and can be relied on for use in excavator cycle time monitoring applications in mining operations.

3.2.2 Processing speed

Processing speed is measured in Frames Per Second (FPS) or inference time per frame. In addition, preprocessing and postprocessing time also affect the overall performance of the system. Processing speed testing is done by running an object detection model on a GTX 1060 Ti GPU, as shown in Table 1. Measurements include:

1. Preprocessing time: The time required to prepare the data before it is fed to the model.
2. Inference time: The time required for the model to process one frame and produce a prediction.
3. Postprocessing time: The time required to process the model's prediction results.

The detection model demonstrates efficient processing capabilities when running on GPU NVIDIA GTX 1060 Ti hardware. The system achieves a total processing time of 19.8 milliseconds per frame, which translates to an impressive frame rate of 30 FPS. This performance is broken down into three distinct phases: preprocessing requires 2.0 milliseconds, the core inference process takes 10.5 milliseconds, and post-processing completes in 7.3 milliseconds. The test results show that the object detection model has a fairly high processing speed when running on the NVIDIA GTX 1060 Ti GPU, with an average speed of 30 FPS. This speed meets the requirements for real-time applications, which generally require a minimum of 30 FPS to run smoothly. This fast-processing speed enables timely object detection and classification in operational environments.

3.2.3 Consistency and stability

Consistency and stability refer to the ability of a model to produce similar results and performance under different conditions and over time. The results of the detection evaluation in various conditions can be seen in Fig. 7.

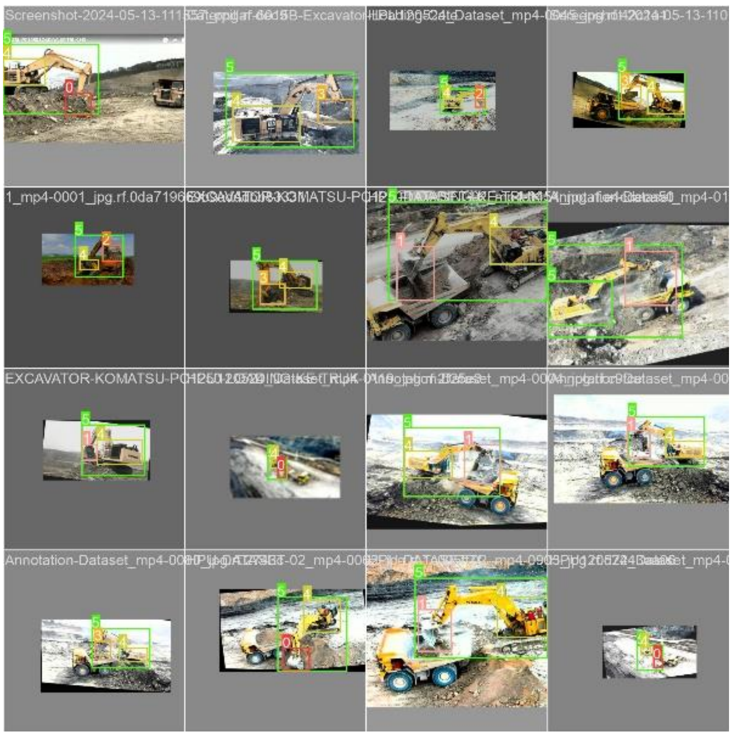

Fig. 7. Results of detection evaluation in various conditions

Based on Fig. 7 shows the results of object detection in various excavator operating conditions. The figure shows that the detection system can be accessed from all directions, allowing mining operations to be monitored in real time. Object detection was successfully carried out in various environments, including mining areas with different soil conditions, varying lighting, and various camera angles and distances. Despite variations in environmental conditions, the object detection system is still able to identify and track the excavator and its components well.

Consistent detection results are seen in various images with different backgrounds and conditions. The object detection system can identify the excavator, upper structure, bucket-digging, bucket-load, bucket-empty, and bucket-dumping with a high level of confidence. In addition, the image shows that the object detection system has good performance stability, with the ability to detect and track objects accurately even though there are changes in operational conditions. This shows that the developed object detection model has good resilience to environmental and operational variations.

3.2.4 Ease of use

The user-friendliness aspect is also evaluated based on feedback from operators who use the application during testing. The application interface is designed to be intuitive and easy to evaluate the accuracy of the object detection system used, allowing operators in the field to easily understand and monitor excavator activities. Features such as bounding box visualization and detection data logging can be seen in Fig. 8. In addition, there are features for setting confidence and also setting movement tolerance which are very helpful in facilitating monitoring. This application also provides a fast response by displaying detection results in real time. The confidence and movement tolerance setting interface can be seen in Fig. 8**.**

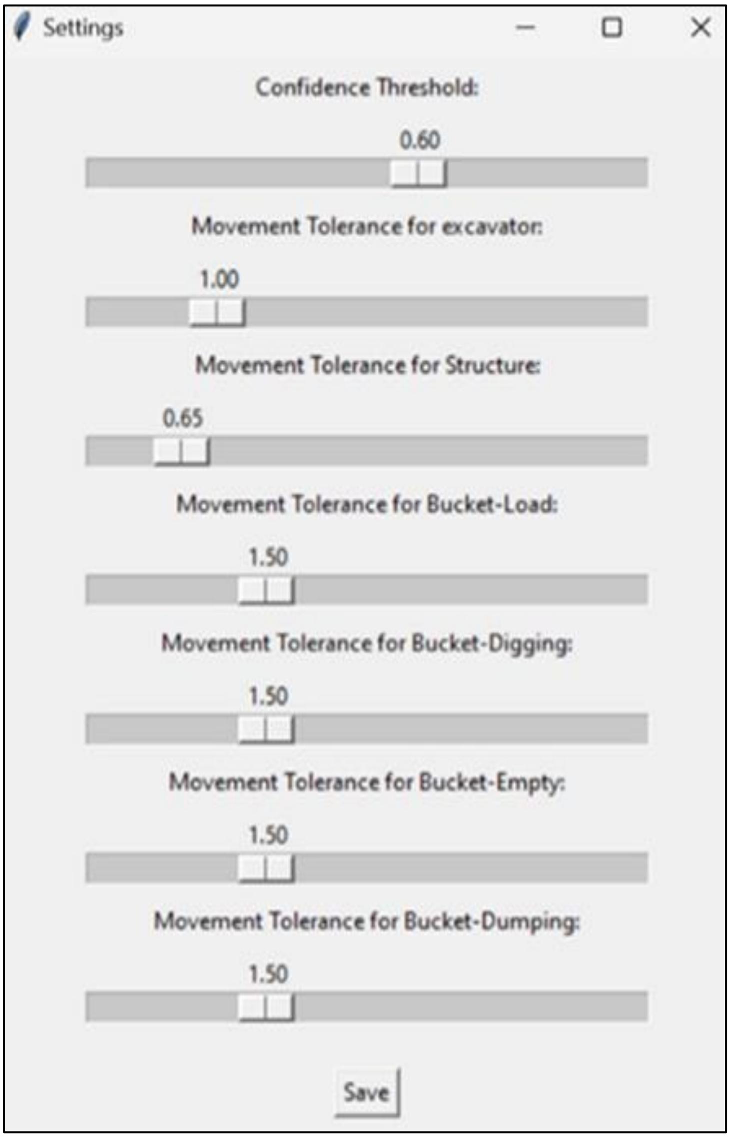

Fig. 8. Confidence and movement tolerance setting interface

3.2.5 Comparison with manual method

Based on the developed object detection system results, a comparison was made with the manual method using a stopwatch. Testing was carried out by taking data at the same time and environmental conditions to ensure consistency of results. The comparison of excavator cycle time results can be seen in Table 1.

Table 1. Comparison of Excavator Cycle Time Results (Object Detection vs Manual Stopwatch)

Operation phase	Manual method (s)	Object detection method (s)	Difference
AVG DIG TO LOAD	12.79	10.94	1.85
AVG SWING LOAD	4.91	4.99	-0.08
AVG DUMP	3.66	2.19	1.47
AVW SWING EMPTY	5.08	4.31	0.77
AVG CYCLE TIME	26.91	23.44	3.47

Based on Table 1, it is found that the average cycle time has increased by 12% from manual to a detection system. Based on the object detection method**,** it shows slightly faster results in some phases of operation. The average difference between the two results shows that object detection has a fairly high accuracy and is reliable. The YOLO-based object detection system can automate the measurement of excavator cycle time with accuracy comparable to manual methods. The use of this system reduces dependence on manual measurements that are prone to human error and provides more consistent and efficient results automatically.

4 Conclusion

The real-time excavator cycle time detection system developed in this study achieved a detection accuracy of 94.2% mAP and a processing speed of 30 FPS, validating its effectiveness for deployment in mining operations. The system provides accurate and timely information on excavator activity, allowing field operators to monitor performance more efficiently. The reduction in manual monitoring time and improved data accuracy contribute directly to increased equipment productivity and operational efficiency. Additionally, the intuitive interface of the system supports broader adoption by workers. Therefore, it is important to integrate this detection system with a centralized dashboard for real-time visualization, maintenance tracking, and safety monitoring to further enhance operational control in mining environments.

References

[1] Erwin Syahrudin, Ema Utami, and Anggit Dwi Hartanto, “Enhanced Yolov8 with OpenCV for Blind-Friendly Object Detection and Distance Estimation,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 8, no. 2, pp. 199–207, 2024, doi: 10.29207/resti.v8i2.5529.

[2] G. L. Nugraha, M. Ajis, H. S. Adhi, and D. Zakaria, “Performance Improvement of Hydraulic Excavator Efficiency: A Literature Review,” J. Mechatronics Artif. Intell., vol. 1, no. 1, pp. 27–44, 2024, doi: 10.17d509/jmai.v1i1.69947.

[3] E. A. Nugroho, J. D. Setiawan, M. Munadi, and M. Diki, “Enhancing textile quality control with the application of teachable machine and Raspberry Pi as machine learning-based image processing,” J. Polimesin, vol. 20, no. 2, pp. 121–127, 2023, doi: 10.30811/jpl.v22i5.5308.

[4] A. Yolov8 et al., “Analisa Kemampuan Algoritma YOLOv8 Dalam Deteksi Objek Manusia Dengan Metode Modifikasi Arsitektur,” J. Sains Komput. Inform., vol. 7, no. 2, pp. 891–901, 2023, doi: 10.30645/j-sakti.v7i2.694.

[5] N. Zaghden, E. Ibrahim, M. Safaldin, and M. Mejdoub, “Integrating Attention Mechanisms in YOLOv8 for Improved Fall Detection Performance,” Comput. Mater. Contin., vol. 83, no. 1, pp. 1117–1147, 2025, doi: 10.32604/cmc.2025.061948.

[6] D. Wang, Y. Zhang, R. Zhang, G. Nie, and W. Wang, “Detection and assessment of post-earthquake functional building ceiling damage based on improved YOLOv8,” *J. B*uild. Eng., vol. 98, p. 111315, 2024, doi: 10.1016/j.jobe.2024.111315.

[7] G. Wen, M. Li, Y. Tan, C. Shi, Y. Luo, and W. Luo, "Enhanced YOLOv8 algorithm for leaf disease detection with lightweight GOCR-ELAN module and loss function: WSIoU," Comput. Biol. Med., vol. 186, p. 109630, 2025, doi: 10.1016/j.compbiomed.2024.109630.
[8] Y. Liu, W. Qiu, K. Fu, X. Chen, L. Wu, and M. Sun, "An improved YOLOv8 model and mask convolutional autoencoder for multi-scale defect detection of ceramic tiles," Measurement, vol. 248, p. 116847, 2025, doi: 10.1016/j.measurement.2025.116847.
[9] Z. Wang et al., "Detection of small foreign objects in Pu-erh sun-dried green tea: An enhanced YOLOv8 neural network model based on deep learning," Food Control, vol. 168, p. 110890, 2025, doi: 10.1016/j.foodcont.2024.110890.
[10] E. U. Armin, A. Purnama Edra, F. I. Alifin, I. Sadidan, I. P. Sary, and U. Latifa, "Performa Model YOLOv8 untuk Deteksi Kondisi Mengantuk pada pengendara mobil," BRAHMANA J. Penerapan Kecerdasan Buatan, vol. 5, no. 1, pp. 67–76, 2023, [Online]. Available: https://doi.org/10.30645/brahmana.v5i1.279.
[11] I. P. Sary, S. Andromeda, and E. U. Armin, "Performance Comparison of YOLOv5 and YOLOv8 Architectures in Human Detection using Aerial Images," Ultim. Comput. J. Sist. Komput., vol. 15, no. 1, pp. 8–13, 2023, doi: 10.31937/sk.v15i1.3204.
[12] M. Ikbal and R. A. Saputra, "Pengenalan Rambu Lalu Lintas Menggunakan Metode Yolov8," JIKA (Jurnal Inform., vol. 8, no. 2, pp. 204–212, 2024, doi: 10.31000/jika.v8i2.10609.
[13] E. Sulzbach et al., "Deep learning model optimization methods and performance evaluation of YOLOv8 for enhanced weed detection in soybeans," Comput. Electron. Agric., vol. 232, p. 110117, 2025, doi: 10.1016/j.compag.2025.110117.
[14] W. Wu et al., "Enhancing placental pathology detection with GAMatrix-YOLOv8 model," Heliyon, vol. 11, no. 4, p. e42441, 2025, doi: 10.1016/j.heliyon.2025.e42441.
[15] Iqra and K. J. Giri, "SO-YOLOv8: A novel deep learning-based approach for small object detection with YOLO beyond COCO," Expert Syst. Appl., vol. 280, p. 127447, 2025, doi: 10.1016/j.eswa.2025.127447.
[16] Z. Xu, F. Luo, F. Chen, H. Wu, and M. Yu, "Surgical tool detection in open surgery based on improved-YOLOv8," Biomed. Signal Process. Control, vol. 105, p. 107548, 2025, doi: 10.1016/j.bspc.2025.107548.
[17] X. Bao et al., "A novel method of void detection in rebar-affected areas based on transfer learning and improved YOLOv8," Tunn. Undergr. Sp. Technol., vol. 158, p. 106440, 2025, doi: 10.1016/j.tust.2025.106440.
[18] J. Xu et al., "Transillumination imaging for detection of stress cracks in maize kernels using modified YOLOv8 after pruning and knowledge distillation," Comput. Electron. Agric., vol. 231, p. 109959, 2025, doi: 10.1016/j.compag.2025.109959.
[19] Y. Yanto, F. Aziz, and I. Irmawati, "Yolo-V8 Peningkatan Algoritma Untuk Deteksi Pemakaian Masker Wajah," JATI (Jurnal Mhs. Tek. Inform., vol. 7, no. 3, pp. 1437–1444, 2023, doi: 10.36040/jati.v7i3.7047.
[20] H. Xia et al., "Bridge crack detection algorithm designed based on YOLOv8," Appl. Soft Comput., vol. 171, p. 112831, 2025, doi: 10.1016/j.asoc.2025.112831.
[21] Q. N. Huy Minh, N. N. Dinh, L. V. Ho, and C. P. Huu, "Real-time traffic accident detection using yolov8," Transp. Res. Procedia, vol. 85, pp. 68–75, 2025, doi: https://doi.org/10.1016/j.trpro.2025.03.135.
[22] E. Casas, L. Ramos, C. Romero, and F. Rivas-Echeverría, "A comparative study of YOLOv5 and YOLOv8 for corrosion segmentation tasks in metal surfaces," Array, vol. 22, p. 100351, 2024, doi: 10.1016/j.array.2024.100351.
[23] T. Erlina and M. Fikri, "Yolo Algorithm-Based Visitor Detection System for Small Retail Stores Using Single Board Computer," J. Appl. Eng. Technol. Sci., vol. 4, no. 2, pp. 908–920, 2023, doi: 10.37385/jaets.v4i2.1872.
[24] R. Adrian, "Implementation of YOLO (You Only Look Once) Algorithm for Drowsiness Detection as An Additional Safety Feature in the Operation of Crane Equipment in Real Time," J. Inov. Teknol. dan Rekayasa, vol. 10, no. 1, pp. 113–120, 2025, doi: 10.31572/inotera.Vol10.Iss1.2025.ID458.
[25] S. K. Inayah, W. Wahyudin, and D. Herwanto, "Analisis Kualitas Produk Toolbox Menggunakan Metode Seven Tools di PT. KSKB," J. Asiimetrik J. Ilm. Rekayasa Inov., vol. 5, no. 2, pp. 263–272, 2023, doi: 10.35814/asiimetrik.v5i2.4877.
[26] M. Yamin and M. Z. Alfasha, "Numerical study of downwash flow on rice plant protection drone using computational fluid dynamics method," J. Polimesin, vol. 22, no. 4, pp. 432–440, 2024, doi: 10.30811/jpl.v22i4.5036.
[27] A. A. Rasjid, B. Rahmat, and A. N. Sihananto, "Implementasi YOLOv8 Pada Robot Deteksi Objek," J. Technol. Syst. Inf., vol. 1, no. 3, pp. 1–9, 2024, doi: 10.47134/jtsi.v1i3.2969.
[28] R. Winarso, Slamet Khoeron, R. Wibowo, and Darmanto, "Hardening of bucket teeth made from creusabro 8000 steel by using the induction hardening method," J. Polimesin, vol. 20, no. 2, pp. 121–127, 2023, doi: 10.30811/jpl.v21i3.3672.
[29] F. Marpaung, H. Purnama, I. N. Artana, N. Kurniawati, D. Samodrawati, and T. Surawan, "Design and simulation of offshore crane structure with capacity of 400 ton," J. Polimesin, vol. 22, no. 2, pp. 186–190, 2024, doi: 10.30811/jpl.v22i2.4162.
[30] C. Li, Y. Zhu, and M. Zheng, "A multi-objective dynamic detection model in autonomous driving based on an improved YOLOv8," Alexandria Eng. J., vol. 122, pp. 453–464, 2025, doi: 10.1016/j.aej.2025.03.020.
[31] A. T. Khan, S. M. Jensen, and A. R. Khan, "Advancing precision agriculture: A comparative analysis of YOLOv8 for multi-class weed detection in cotton cultivation," Artif. Intell. Agric., vol. 15, no. 2, pp. 182–191, 2025, doi: 10.1016/j.aiia.2025.01.013.
[32] J. M. Polgan et al., "Evaluasi Performa Yolov8 Dalam Deteksi Objek Di Depan Kendaraan Dengan Variasi Kondisi Lingkungan," J. Minfo Polgan, vol. 13, no. 2, pp. 1755–1773, 2024, doi: 10.33395/jmp.v13i2.14228 e-ISSN.
[33] E. Panja, H. Hendry, and C. Dewi, "YOLOv8 Analysis for Vehicle Classification Under Various Image Conditions," Sci. J. Informatics, vol. 11, no. 1, pp. 127–138, 2024, doi: 10.15294/sji.v11i1.49038.
[34] Husnan, Fatichah, and Dikairono, "Deteksi Objek Menggunakan Metode YOLO dan Implementasinya pada Robot Bawah Air," J. Teknol. Sepuluh Nop., vol. 12, no. 3, pp. 1–9, 2023, doi: 10.12962/j23373539.v12i3.122326.