Computer Science (CO-SCIENCE)
Volume 6 Issue 1 January 2026
Accreditation **Sinta 4** No. SK : 230/E/KPT/2022

Performance Evaluation of YOLOv8 for Railway Switching Operation Safety Monitoring

Aulya Anggita Putri Selendra[1*], Teguh Arifianto[2], Fathurrozi Winjaya[3]

[1,2,3]Railway Electrical Technology, Politeknik Perkeretaapian Indonesia, Madiun, Indonesia
Jl. Tirta Raya, Nambangan Lor, Maguharjo, Kota Madiun, Jawa Timur, Indonesia

e-mail: [1]aulya.tep2220042@taruna.ppi.ac.id, [2]teguh@ppi.ac.id, [3]fathurrozi@ppi.ac.id

(*) Corresponding Author

Article Info: Received: 10-01-2026 | Revised : 21-01-2026 | Accepted : 27-01-2026

Abstracts - Safety in railway shunting operations requires continuous monitoring of train distance and speed to reduce the risk of operational accidents. In practice, shunting activities are still highly dependent on manual observation and verbal communication, while the performance of vision based safety systems under real operational conditions remains uncertain. In addition, comprehensive performance evaluations of deep learning based object detection models in real shunting environments, particularly under different hardware capabilities and lighting conditions, are still limited. This study aims to evaluate the performance of the YOLOv8 algorithm for real-time distance and speed monitoring during railway shunting operations. The system was tested using a camera-based detection approach under different processor configurations, namely an internal CPU and an RTX GPU, and under morning, daytime, and nighttime lighting conditions. System performance was evaluated based on accuracy, precision, and real-time detection capability across these conditions. The results show that the system achieved an average accuracy of 87.32% when operating on a CPU which increased to 91.30% when using a GPU. Optimal performance was observed under adequate daylight conditions, while reduced lighting led to a decline in performance, particularly on CPU-based processing. These findings indicate that hardware configuration and lighting conditions play a critical role in determining the reliability of YOLOv8-based safety monitoring systems for railway shunting operations.

Keywords : safety, railway, shunting, YOLOv8, real-time

INTRODUCTION

Safety in railway shunting operations is a strategic element in railway operations, as it directly relates to the management of rolling stock movement within confined areas that carry a high risk potential (Dwiatmoko, 2025; Handoko, 2023; Wibisono & Zidan, 2023). The shunting process requires precise coordination between the locomotive driver and ground personnel to ensure that train movements are conducted safely, controlled, and in accordance with operational regulations (Aditiatmoko & Latifah, 2022; Arifianto et al., 2024; Marsusiadi et al., 2023). Therefore, strengthening safety measures during shunting activities is essential to support the reliability of railway operational systems.

Ideally, shunting operations should be carried out through well-coordinated operational procedures that ensure safe, controlled, and efficient train movements (Pfaff, 2023; Reichmann et al., 2025). In this context, shunting activities involve close interaction between locomotive drivers and ground personnel, particularly in managing train speed, distance, and movement direction during coupling or separation processes. Effective communication and accurate perception of train position are essential elements to support decision-making during shunting, as improper control of distance and speed may increase the risk of operational incidents (Kim et al., 2025). Therefore, consistent monitoring of distance and speed becomes a fundamental requirement to support safety during shunting activities.

However, field conditions reveal several technical and operational challenges that hinder the effective implementation of these safety standards. Noise levels inside locomotive cabins can exceed 90 dBA which may reduce concentration and disrupt communication between operators (Asri et al., 2025; Sangadi & Ratrikaningtyas, 2024). In addition, signal interference in handheld communication devices often results in delayed or incomplete information exchange (Sumarahardhi & Santoso, 2023). Limited visibility caused by blind spots in shunting areas further complicates the driver's ability to accurately monitor train positions, increasing the potential for operational errors.

These conditions indicate a gap between prescribed safety regulations and actual operational practices in the field. Conventional safety approaches that rely on manual visual observation and verbal communication have not fully ensured optimal safety performance. Previous studies have proposed the use of computer vision and deep learning techniques to support automatic object detection in railway environments (Guan et al., 2022; Kapoor et al., 2022; Mauri et al., 2022; Wu et al., 2021). Deep learning based object detection algorithms, particularly those from the YOLO family, have demonstrated advantages in terms of detection speed and accuracy (Brintha & Jawhar, 2024; Guan et al., 2022; Meng et al., 2023).

Nevertheless, most existing studies are conducted in controlled or simulated environments and primarily focus on algorithmic development or detection accuracy, with limited attention to real railway operating conditions (Zhang et al., 2024). In particular, comprehensive evaluations of YOLOv8 performance in actual shunting operations, considering variations in hardware specifications and lighting conditions, remain limited. Such factors are critical, as shunting activities are performed under diverse environmental conditions and with varying computational resources in practical applications (Lema et al., 2024; Zhou et al., 2025).

Based on this research gap, this study aims to evaluate the performance of the YOLOv8 algorithm for real-time distance and speed monitoring in railway shunting operations under different hardware configurations and lighting conditions. The scientific contribution of this study lies in providing empirical performance evaluation results of YOLOv8 implemented in a real shunting environment, highlighting the influence of processor capability and environmental lighting on detection accuracy and reliability. The findings are expected to support the optimization of vision-based safety monitoring systems for practical railway shunting operations.

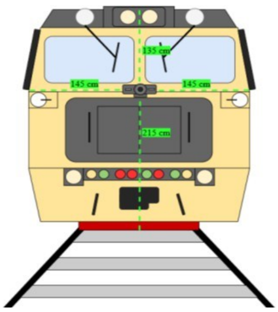
RESEARCH METHOD

1. Research Design

    This study employs an experimental research design to evaluate the performance of the YOLOv8 algorithm for distance and speed monitoring in railway shunting operations. The research stages consist of dataset preparation, model training, system implementation, and performance evaluation. The evaluation focuses on analyzing system performance under different hardware configurations and lighting conditions in real shunting operations.

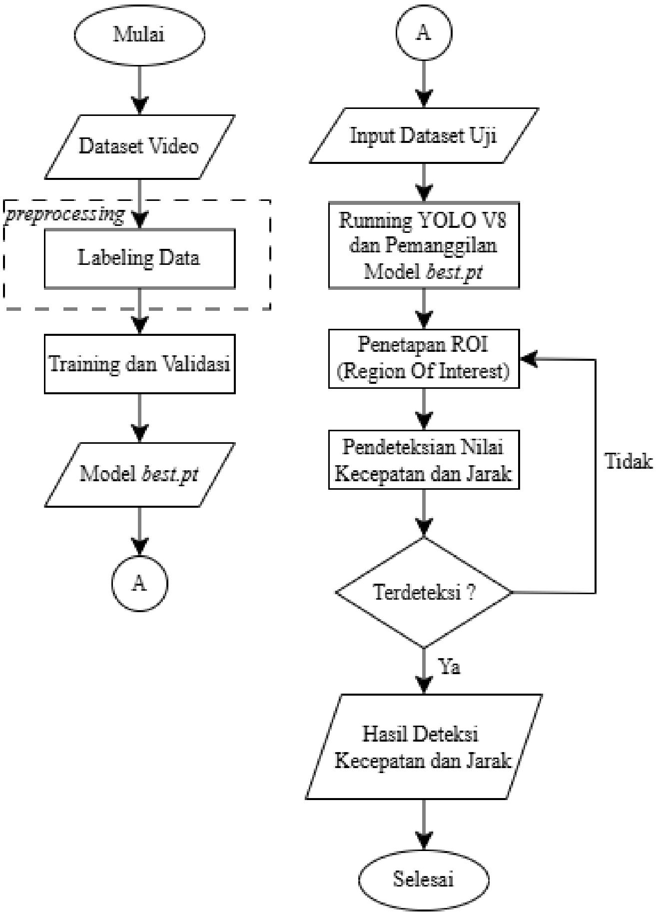
2. Dataset Preparation and Annotation

    The dataset used in this study consists of video recordings collected from real railway shunting operations conducted under varying lighting conditions, including morning, daytime, and nighttime scenarios. The recorded videos were extracted into image frames and annotated using the Computer Vision Annotation Tool (CVAT). Each frame was labeled by defining bounding boxes for the “train” object as the primary detection target.



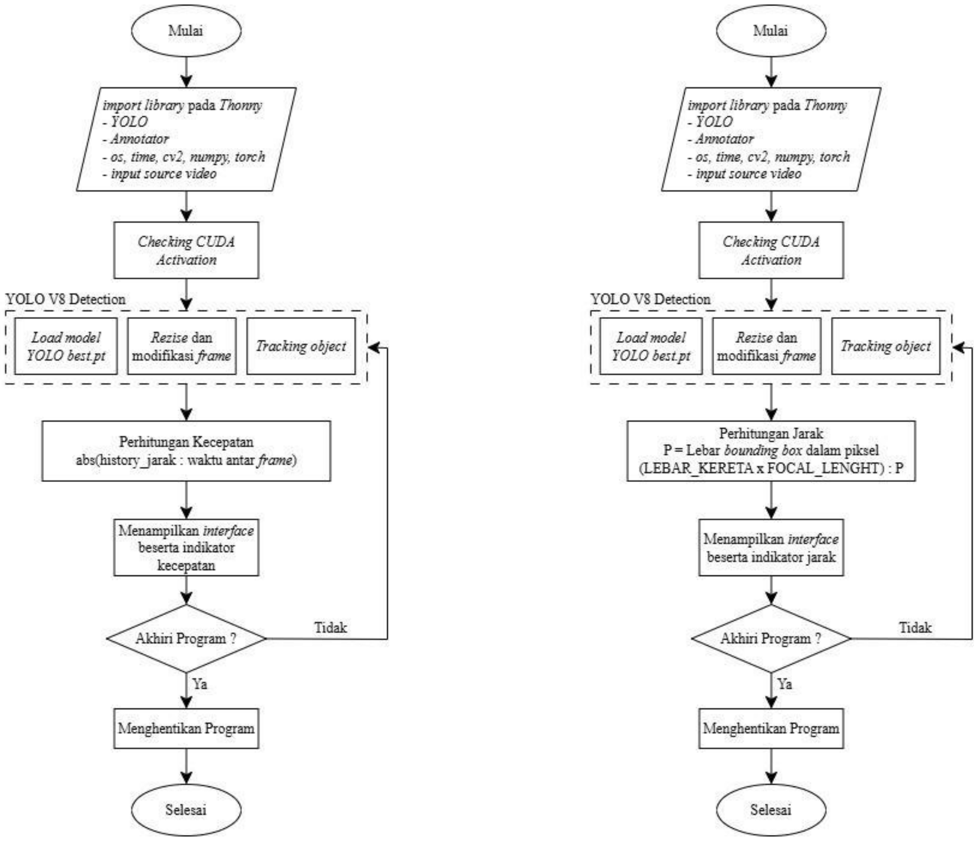
Source: Research Result(2025)
Figure 1. Webcam placement on the locomotive

The overall dataset preparation and annotation process is illustrated in Figure 2. The annotated dataset was subsequently divided into training and validation sets to support supervised learning during the model training phase. This dataset reflects real operational environments, allowing the evaluation of model robustness under practical conditions.



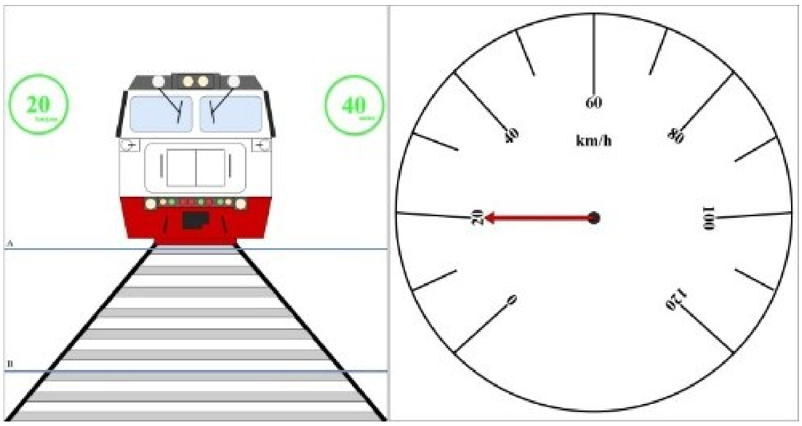
Source: Research Result (2025)
Figure 2. Dataset design flowchart

3. Model Training Configuration
   Model training was conducted using the YOLOv8 architecture on the Google Colaboratory platform. The training process utilized the annotated dataset to develop an object detection model capable of recognizing train objects during shunting operations. After completing the training process, the model with the best performance was selected and used for subsequent system implementation and evaluation.
4. System Implementation
   The trained YOLOv8 model was integrated into a Python-based detection system for real-time processing. During system operation, video frames captured from the camera were processed sequentially. The system automatically adjusted the computational process based on the available hardware, performing inference using either an internal CPU or an RTX GPU. The detection results were displayed as bounding boxes overlaid on the video stream.
5. Distance and Speed Estimation Method
   The workflow for distance and speed estimation implemented in this study is presented in Figure 3. Speed estimation was performed by calculating the displacement of the detected object between consecutive frames divided by the corresponding time interval. Distance estimation was based on the apparent width of the detected object's bounding box in pixel units. This method assumes a constant actual width of the train and utilizes the relationship between object size in the image and its distance from the camera. Although this approach provides an approximate distance estimation, it is suitable for real-time shunting safety monitoring, where relative distance information is required to support operational decision-making.



Source: Research Result (2025)
Figure 3. (a) Speed detection flowchart and (b) distance detection flowchart

6. Interface Design
   The real-time monitoring interface used during system evaluation is shown in Figure 4. The interface displays the live video stream with detected objects marked by bounding boxes, along with numerical information on estimated distance and speed. This interface was used during testing to support performance evaluation and to assess the system's capability to provide real-time visual information during shunting operations.



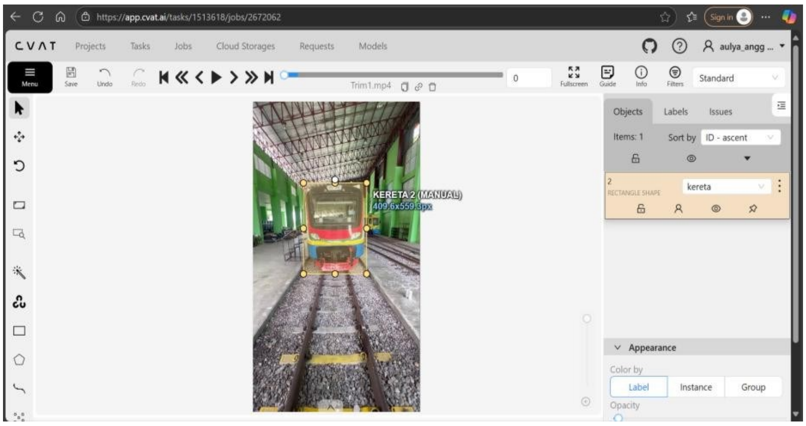
Source: Research Result (2025)
Figure 4. Design of the speed and distance interface

7. Performance Evaluation Metrics
   System performance was evaluated using accuracy, error percentage, and precision metrics for both distance and speed estimation. Performance testing was conducted under different processor configurations (CPU and GPU) and lighting conditions. These metrics were used to assess detection reliability, estimation consistency, and real-time performance in actual shunting operations.

RESULTS AND DISCUSSION

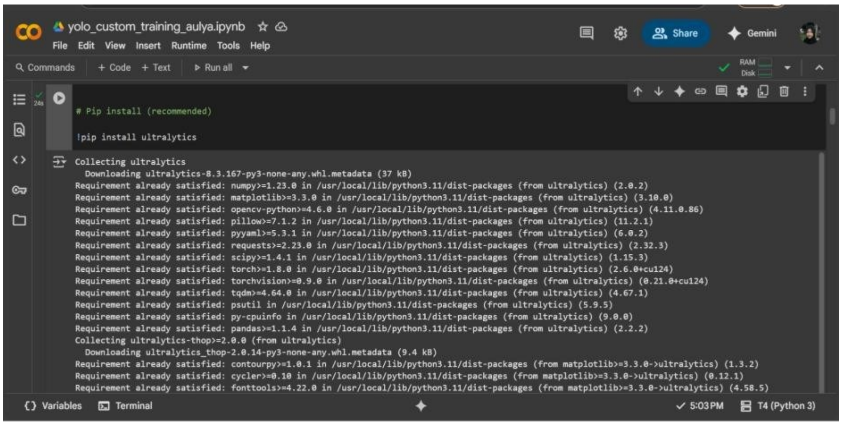
1. Preprocessing

The preprocessing stage aims to recognize and extract patterns based on previously learned patterns within the system. New patterns obtained from input data are subsequently classified into specific categories according to the results of the prior training process. This classification process utilizes a neural network as the primary approach during the training phase. The training data is derived from features extracted from a prepared dataset. Through this approach, the system is expected to consistently recognize and differentiate object patterns while maintaining a high level of accuracy (Zhao et al., 2024).



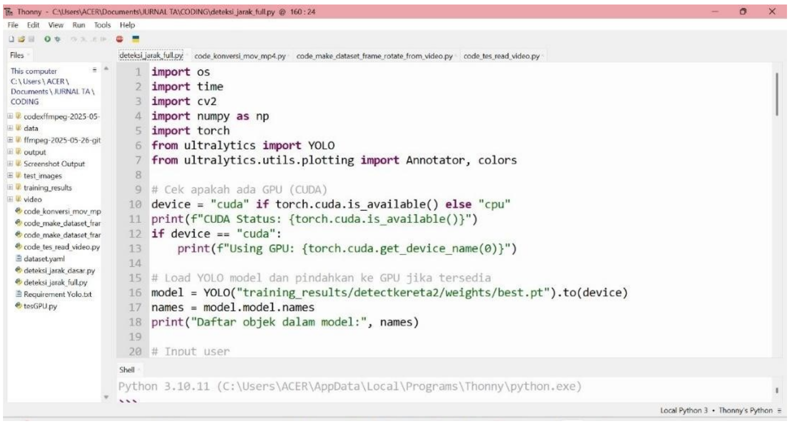
Source: Research Result(2025)
Figure 5. Bounding box determination

As part of this stage, preprocessing begins with collecting a dataset consisting of video recordings of shunting operations captured at several different time intervals. The acquired videos are then uploaded to the Computer Vision Annotation Tool (CVAT) platform for data annotation, as shown in Figure 5. At this stage, the video is divided into multiple frames per second, and each frame is labeled according to the target detection object. All labeled data are stored in a dedicated folder on Google Drive for efficient management and retrieval. Furthermore, bounding boxes are assigned to the “train” object which serve as references for calculating the distance and speed of the rolling stock.



Source: Research Result(2025)
Figure 6. View on google colaboratory

The dataset stored on Google Drive is then used as input for model training on the Google Colaboratory platform. During this phase, the Ultralytics package is installed to enable the YOLOv8 algorithm in the Colaboratory environment. Training and validation are subsequently conducted on the prepared dataset, as illustrated in Figure 6. The trained model, after completing the specified number of epochs, is saved as a module named *best.pt*. This module is then integrated into a Thonny Python-based detection program to process video input from the webcam, allowing the system to perform real-time object detection, as demonstrated in Figure 7.



Source: Research Result(2025)
Figure 7. Thonny python interface

2. Program Interface

Figure 8 presents the design of the interface for the shunting distance and speed detection system which is used to visually display processed data. The interface presents a real-time video feed from the webcam, with detected objects highlighted using bounding boxes to clearly indicate their position and presence. In addition to the visual display, the system provides several supporting indicators essential for monitoring, such as FPS (Frame Per Second) to represent processing performance, object detection status, and the estimated object distance in meters. Moreover, the interface displays object speed in meters per second (m/s) and kilometers per hour (km/h), allowing users or train operators to gain a comprehensive and accurate overview of the shunting conditions in real time.

Source: Research Result(2025)
Figure 8. Program interface display

3. Definition of Evaluation Metrics

In this study, system performance was evaluated using three main metrics, namely Error (%), Accuracy (%), and Precision (%), to assess both detection reliability and estimation quality under different operating conditions. Error (%) represents the percentage deviation between the estimated value produced by the system and the reference value obtained during testing, indicating the magnitude of estimation inaccuracy. Accuracy (%) reflects the closeness of the estimated distance or speed values to the reference measurements, providing an overall indication of system correctness. Precision (%) describes the consistency of the system in producing similar estimation results across repeated measurements, indicating the stability of detection performance. These metrics were selected to comprehensively evaluate the system’s ability to estimate distance and speed under varying hardware configurations and lighting conditions during railway shunting operations.

4. Speed Test Results

Speed tests conducted during the morning period were carried out under low-light conditions, as the sun had not fully risen. The measured light intensity ranged from 1,000 to 8,100 lux. According to Table 1, the system running on an internal CPU produced an average error of 20.71% with an accuracy of 79.28%. In contrast, utilizing an RTX GPU reduced the error to 9.66% while increasing accuracy to 90.33%.

Table 1. Morning speed test

Morning Speed Test (06.00-09.00)				
Session	CPU Internal		GPU RTX	
	Error (%)	Accuracy (%)	Error (%)	Accuracy (%)
Morning 1	12,92	87,07	11,85	88,14
Morning 2	28,02	71,97	4,21	95,78
Morning 3	21,19	78,80	12,92	87,07
Mean ($\bar{X}$)	20,71	79,28	9,66	90,33

Source: Research result (2025)

During the afternoon, testing was performed under optimal lighting conditions with high sunlight intensity, ranging between 73,000 and 82,000 lux. As shown in Table 2, the CPU-based system produced an average error of 10.37% with an accuracy of 89.62%, whereas the RTX GPU demonstrated superior performance with an error of 8.32% and accuracy reaching 91.67%. This indicates that adequate lighting significantly contributes to improved system performance.

Table 2. Afternoon speed test

Afternoon Speed Test (11.00-14.00)				
Session	CPU Internal		GPU RTX	
	Error (%)	Accuracy (%)	Error (%)	Accuracy (%)
Afternoon 1	9,618	90,38	6,621	93,3
Afternoon 2	5,582	94,41	9,654	90,3
Afternoon 3	15,91	84,08	8,711	91,2
Mean ($\bar{X}$)	10,37	89,62	8,32	91,67

Source: Research result (2025)

Nighttime testing was performed under severely limited lighting, with no natural sunlight. Measured light intensity ranged from 54 to 71 lux. Table 3 shows that the system using an internal CPU resulted in an average error of 12.57% with an accuracy of 87.42%, while the RTX GPU achieved an error of 8.38% and an accuracy of 91.61%. To support detection during nighttime, additional lighting from train lamps was utilized.

Table 3. Night speed test

Night Speed Test (17.00-20.00)				
Session	CPU Internal		GPU RTX	
	Error (%)	Accuracy (%)	Error (%)	Accuracy (%)
Night 1	5,801	94,19	5,982	94,0
Night 2	9,569	90,43	11,92	88,0
Night 3	4,549	95,45	3,588	96,4
Mean ($\bar{X}$)	12,57	87,42	8,38	91,61

Source: Research result (2025)

5. Distance Test Results

Morning distance tests were conducted under low-light conditions, with light intensity ranging from 1,000 to 8,100 lux. As presented in Table 4, using an internal CPU produced an average error of 4.12%, with accuracy of 95.87% and precision of 97.53%. The RTX GPU reduced the error to 3.81%, while accuracy increased to 96.18% and precision reached 99.23%. These results indicate that sufficient lighting significantly improves detection performance.

Table 4. Morning distance test

Morning Distance Test (06.00-09.00)						
Session	CPU Internal			GPU RTX		
	Error (%)	Accuracy (%)	Precision (%)	Error (%)	Accuracy (%)	Precision (%)
Morning 1	4,91	95	97	3,40	96	99
Morning 2	2,32	97	97	3,38	96	99
Morning 3	5,13	94	97	4,64	95	99
Mean ($\bar{X}$)	4,12	95	97	3,81	96	99

Source: Research result (2025)

Afternoon testing occurred under optimal sunlight exposure (73,000-82,000 lux). Table 5 shows that the internal CPU system achieved an average error of 4.65%, accuracy of 95.34%, and precision of 98.93%. The RTX GPU again outperformed the CPU with an error of 3.59%, accuracy of 96.40%, and precision of 99.23%.

Table 5. Afternoon distance test

Afternoon Distance Test (11.00-14.00)						
Session	CPU Internal			GPU RTX		
	Error (%)	Accuracy (%)	Precision (%)	Error (%)	Accuracy (%)	Precision (%)
Afternoon 1	4,21	95	98	3,66	96	98
Afternoon 2	4,85	95	98	3,65	96	99
Afternoon 3	4,88	95	99	3,48	96	99
Mean ($\bar{X}$)	4,65	95	98	3,59	96	99

Source: Research result (2025)

Nighttime distance testing was conducted under severely limited lighting (54-71 lux). Table 6 shows that the internal CPU produced an average error of 23.44%, accuracy of 76.37%, and precision of 98.11%, whereas the RTX GPU achieved a lower error of 18.42%, accuracy of 81.57%, and precision of 99.74%. Nighttime performance was enhanced using additional lighting from train lamps which improved image quality and detection accuracy.

Table 6. Nighttime distance test

Nighttime Distance Test (17.00-20.00)						
Session	CPU Internal			GPU RTX		
	Error (%)	Accuracy (%)	Precision (%)	Error (%)	Accuracy (%)	Precision (%)
Night 1	23,7	76,23	97,87	18,3	81,61	99,7
Night 2	24,1	75,35	98,04	18,6	81,32	99,7
Night 3	22,4	77,53	99,83	18,2	81,78	99,7
Mean ($\bar{X}$)	23,4 4	76,37	98,11	18,42	81,57	99,74

Source: Research result (2025)

CONCLUSION

his study demonstrates that the YOLOv8-based vision system is capable of supporting distance and speed monitoring during railway shunting operations in real operational environments. The results indicate that the performance of the system is strongly influenced by hardware capability and environmental lighting conditions, confirming that computational resources and illumination play a critical role in ensuring reliable real-time detection. The findings highlight that YOLOv8 can function effectively as a shunting safety support tool when appropriate operational conditions are met.

From a practical perspective, the study shows that the use of higher-performance processors contributes to more stable detection, particularly in low-light conditions commonly encountered during nighttime shunting operations. Adequate lighting conditions also remain essential to maintain detection reliability, indicating that system deployment should consider both hardware readiness and environmental factors to achieve optimal performance. Despite these findings, this study has several limitations. Distance estimation is based on visual information from a single camera and relies on bounding box geometry which may be affected by perspective distortion, vibration, and extreme lighting variations. In addition, system evaluation was conducted under specific operational scenarios and did not yet include adverse weather conditions or a wider range of shunting speeds.

Future research should focus on improving distance estimation accuracy by integrating additional sensors such as LiDAR or ultrasonic sensors and by applying camera calibration techniques. Further testing under diverse environmental and operational conditions is also recommended to enhance system robustness and support its implementation as a practical safety monitoring solution in railway shunting operations.

REFERENCES

Aditiatmoko, A., & Latifah, K. (2022). Assesmen Digital Untuk Menentukan Status Kelayakan Awak Kereta Api. *Science And Engineering National Seminar 7 (SENS 7)*, *7*, 1–7. https://conference.upgris.ac.id/index.php/sens/article/view/3611

Arifianto, T., Pradjojowaty, I. S., Feryando, D. A., Arifidin, M. A. A., Pratiwi, D. I., & Prasetijo, J. (2024). Wisata Edukasi Pengenalan Petugas Pelayanan Penumpang di Atas Kereta Api pada Siswa RA Babussalam Madiun. *JURPIKAT (Jurnal Pengabdian Kepada Masyarakat)*, *5*(1), 170–180.

https://doi.org/10.37339/jurpikat.v5i1.1500

Asri, A. F., Bachtiar, V. S., Lestari, R. A., & Ridwan, R. (2025). Analisis dan Rekayasa Kebisingan Akibat Aktivitas Alat Transportasi Kereta Api Pada Permukiman (Studi Kasus: Jalur Kereta Api Stasiun Alai-Air Tawar). *Dampak: Jurnal Teknik Lingkungan*, *22*(1), 19–35. https://doi.org/10.25077/dampak.22.1.19-35.2025

Brintha, K., & Jawhar, S. J. (2024). FOD-YOLO NET: Fasteners Fault and Object Detection in Railway Tracks using Deep YOLO Network. *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology*, *46*(4), 8123–8137. https://doi.org/10.3233/JIFS-236445

Dwiatmoko, H. (2025). *Keselamatan Perkeretaapian*. Perkumpulan Masyarakat Perkeretaapian Indonesia (Maska).

Guan, L., Jia, L., Xie, Z., & Yin, C. (2022). A Lightweight Framework for Obstacle Detection in the Railway Image Based on Fast Region Proposal and Improved YOLO-Tiny Network. *IEEE Transactions on Instrumentation and Measurement*, *71*, 1–16. https://doi.org/10.1109/TIM.2022.3150584

Handoko, H. (2023). *Sistem dan Manajemen Stasiun*. CV. Mega Press Nusantara.

Kapoor, R., Goel, R., & Sharma, A. (2022). An Efficient Object and Railway Track Recognition in Thermal Images Using Deep Learning. *Multimedia Tools and Applications*, *841*, 21083–21109. https://doi.org/10.1007/978-981-16-8774-7_20

Kim, T.-Y., Kim, C.-N., Park, C.-W., & Yum, B.-S. (2025). Human Error Accident Analysis of Railway Transport Workers in Shunting Operations : Application of the HEAR Methodology. *Journal of Korean Society of Occupational and Environmental Hygiene*, *35*(3), 292–302. https://doi.org/10.15269/JKSOEH.2025.35.3.292

Lema, D. G., Usamentiaga, R., & García, D. F. (2024). Quantitative Comparison and Performance Evaluation of Deep Learning-Based Object Detection Models on Edge Computing Devices. *Integration*, *95*, 102127. https://doi.org/10.1016/j.vlsi.2023.102127

Marsusiadi, E. N. S., Wiarco, Y., Wijayanti, L. M., & Iswanto, A. P. (2023). Kajian Penggunaan Bahasa Baku dalam Mewujudkan Komunikasi Efektif Awak Sarana Prasarana Melayani Perjalanan Kereta Api. *EDUKASIA: Jurnal Pendidikan Dan Pembelajaran*, *4*(2), 1331–1346. https://doi.org/10.62775/edukasia.v4i2.439

Mauri, A., Khemmar, R., Decoux, B., Haddad, M., & Boutteau, R. (2022). Lightweight Convolutional Neural Network for Real-Time 3D Object Detection in Road and Railway Environments. *Journal of Real-Time Image Processing*, *19*, 499–516. https://doi.org/10.1007/s11554-022-01202-6

Meng, C., Wang, Z., Shi, L., Gao, Y., Tao, Y., & Wei, L. (2023). SDRC-YOLO: A Novel Foreign Object Intrusion Detection Algorithm in Railway Scenarios. *Electronics*, *12*(5), 1256. https://doi.org/10.3390/electronics12051256

Pfaff, R. (2023). Braking Distance Prediction for Vehicle Consist in Low-Speed On-Sight Operation: A Monte Carlo Approach. *Railway Engineering Science*, *31*(2), 135–144. https://doi.org/10.1007/s40534-023-00303-7

Reichmann, M., Stephan, G., Wagner, A., Zajicek, J., Stadlmann, B., Wancura, H., Furian, N., & Vössner, S. (2025). Introducing the Concept of Grades of Automation for Shunting Operations. *Journal of Rail Transport Planning & Management*, *33*, 100500. https://doi.org/10.1016/j.jrtpm.2024.100500

Sangadi, F. J., & Ratrikaningtyas, P. D. (2024). Analisis Fungsi Pendengaran Pada Masinis PT Kereta Api Indonesia (Persero): Kajian Tingkat Kebisingan Lokomotif. *Jurnal Semesta Sehat*, *4*(2), 58–66. https://doi.org/10.58185/j-mestahat.v4i2.133

Sumarahardhi, P. C., & Santoso, D. B. (2023). Identifikasi dan Pemetaan Gangguan Komponen Sistem Persinyalan PT Kereta Api Indonesia (Persero) Resort Karawang. *JATI (Jurnal Mahasiswa Teknik Informatika)*, *7*(3), 1574–1578. https://doi.org/10.36040/jati.v7i3.6875

Wibisono, R. E., & Zidan, M. Y. (2023). Identifikasi Perawatan dan Pemeliharaan Wesel pada Wesel 209 di Stasiun Surabaya Gubeng. *Mitrans: Jurnal Media Publikasi Terapan Transportasi*, *1*(1), 11–18. https://doi.org/10.26740/mitrans.v1n1.p11-18

Wu, Y., Qin, Y., Qian, Y., & Guo, F. (2021). Automatic Detection of Arbitrarily Oriented Fastener Defect in High-Speed Railway. *Automation in Construction*, *131*, 1–15. https://doi.org/10.1016/j.autcon.2021.103913

Zhang, B., Yang, Q., Chen, F., & Gao, D. (2024). A Real-Time Foreign Object Detection Method Based on Deep Learning in Complex Open Railway Environments. *Journal of Real-Time Image Processing*, *21*(166). https://doi.org/10.1007/s11554-024-01548-z

Zhao, X., Wang, L., Zhang, Y., Han, X., Deveci, M., & Parmar, M. (2024). A Review of Convolutional Neural Networks in Computer Vision. *Artificial Intelligence Review*, *57*(99), 1–43. https://doi.org/10.1007/s10462-024-10721-6

Zhou, L., Dong, Y., Ma, B., Yin, Z., & Lu, F. (2025). Object Detection in Low-Light Conditions based on DBS-YOLOv8. *Cluster Computing*, *28*(55). https://doi.org/10.1007/s10586-024-04829-1