



A Comparative Evaluation of YOLOv9 and DETR Models in Traffic Object Detection for Intelligent Surveillance Systems

Mohamad Jamil¹, Nani Nagu, Muhammad Said

¹Department of Informatics Engineering, Universitas Khairun, Ternate, Maluku Utara, Indonesia

²Department of Civil Engineering, Universitas Khairun, Ternate, Maluku Utara, Indonesia

³Department of Electrical Engineering, Universitas Khairun, Ternate, Maluku Utara, Indonesia

¹jamilkhairun@gmail.com, ²nani.nagu@unkhair.ac.id, ³saidsanatu@gmail.com

Abstract

Object detection plays a crucial role in traffic surveillance, particularly in urban environments characterized by high vehicle density, diverse weather conditions, and limited computational resources. Although YOLOv9 and DETR have demonstrated strong performance in general object detection tasks, there is a lack of comparative research evaluating their effectiveness under specific challenges of traffic surveillance. These challenges include the need for real-time processing, accurate detection of small or partially occluded objects, and adaptability to complex traffic scenarios. This study addresses this gap by conducting a comparative evaluation of YOLOv9 and DETR using a custom traffic image dataset, with training iterations varied from 10 to 50 epochs to observe performance development. Evaluation metrics included mean average precision, precision, recall, F1-score, inference time, and object count per image. The results indicated that DETR achieved the highest accuracy across all metrics at the final training stage and detected up to 22 objects per image. However, the average inference time exceeded seven seconds per image, limiting the real-time applicability. Conversely, YOLOv9 achieved competitive accuracy with a significantly faster inference time of approximately 0.43 seconds per image. These findings provide practical insights into the trade-off between accuracy and processing efficiency, and offer guidance for model selection in operational traffic surveillance systems.

Keywords: intelligent systems; DETR; object detection; YOLOv9; traffic surveillance

How to Cite: M. Jamil, N. Nagu, and M. Said, "A Comparative Evaluation of YOLOv9 and DETR Models in Traffic Object Detection for Intelligent Surveillance Systems", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 9, no. 5, pp. 1283 - 1290, Oct. 2025.

Permalink/DOI: <https://doi.org/10.29207/resti.v9i5.6556>

Received: April 18, 2025

Accepted: October 18, 2025

Available Online: October 29, 2025

This is an open-access article under the CC BY 4.0 License

Published by Ikatan Ahli Informatika Indonesia

1. Introduction

One of the main components of an intelligent surveillance system is the ability to detect and recognize traffic objects such as vehicles, pedestrians, traffic signs, and road markings. Reliable object detection enables automatic traffic situation analysis, movement tracking, violation identification, and data-based traffic planning [1]. Therefore, selecting the right object detection model greatly determines the overall effectiveness of the system [2], [3]

Deep learning-based object detection models have shown very promising results. Two of the models that stand out are YOLO (You Only Look Once) and Detection Transformer (DETR) [4]. YOLO is known for its ability to detect objects at high speed and high efficiency, making it suitable for real-time applications. [5], [6]. On the other hand, DETR offers a new

approach by utilizing a transformer architecture, providing improvements in detection accuracy and consistency especially in scenarios with multiple objects or overlapping objects [7].

However, there are not many studies that directly compare the performance of these two models in the context of real traffic surveillance, either in terms of accuracy, processing speed, or computational efficiency. This study aims to fill this gap by conducting a comparative evaluation of the YOLO and DETR models in traffic object detection tasks, as well as providing recommendations for the most optimal model for use in intelligent traffic surveillance systems. Several previous studies have evaluated the performance of YOLO or DETR separately, or in different application contexts. For example, a study by [8] trained YOLOv3 to recognize five major object classes that frequently appear on the road (cars, trucks, pedestrians, traffic signs and traffic lights). The next

study was conducted by [9] using YOLOv4 for object detection and Deep SORT for continuous object tracking with an accuracy rate of 87.98%. Another study was also conducted by [10] developed the YOLOv5-TTS (Traffic Sign) algorithm to detect small traffic signs with an accuracy of 93.7%.

DETR formulates object detection as a set prediction problem, thus reducing detection redundancy and better handling the complexity of spatial relations between objects [11]. The study was conducted by [12] applying the DETR (DEtection TRansformer) algorithm in the task of detecting vehicles on the highway with the highest accuracy results obtained, namely 95.7%. Another study was conducted by [13] comparing the performance of the DETR and YOLO algorithms in detecting 6 categories of traffic safety facilities in urban areas with an accuracy result of 82.3%. Other research was also conducted by [14], namely increasing the accuracy and efficiency of real-time object detection by combining the DETR and YOLOv8 methods, with the accuracy results obtained being 89.67%.

Based on the preceding discussion, each model has distinct advantages and limitations. YOLO excels in detection speed and computational efficiency, making it suitable for real-time surveillance. YOLOv9 is chosen over earlier versions due to its dynamic head, hybrid task-aligned feature fusion, and anchor-free detection, which enhance small-object and multi-scale performance. Conversely, DETR achieves higher accuracy and better spatial representation through its transformer-based architecture, with the standard version selected for its stability despite slower inference and higher computational cost. A comprehensive comparative evaluation of these models in real-world traffic surveillance is therefore essential, not only to measure accuracy and processing speed but also to explore the potential for integration or hybridization to produce a more adaptive and optimal detection system.

The novelty of this study lies in its application of YOLOv9 and DETR on a locally sourced traffic surveillance dataset from Ternate City, Indonesia, which has not been explored in previous comparative studies. Unlike most existing research that evaluates these models in generic or synthetic datasets, this study utilizes real-world CCTV footage to assess performance under typical urban conditions in a developing region. Furthermore, the evaluation incorporates practical deployment considerations, including inference speed and object count per frame, which are critical for real-time applications. This work contributes new insights into the trade-offs between detection accuracy and computational efficiency, and offers practical recommendations for integrating these models into edge-based surveillance systems in resource-constrained environments.

The structure of this paper is as follows: Section 2 discusses the methods used in this study, including datasets, evaluation metrics, and experimental designs

to compare the performance of the YOLO and DETR models. Section 3 presents experimental results and analysis comparing the two models in terms of accuracy, processing speed, and computational efficiency in real traffic surveillance scenarios. Section 4 concludes this study, provides recommendations for the most optimal model to be applied in intelligent traffic surveillance systems, and offers directions for future research development.

2. Methods

This section describes the dataset used in the study and the proposed method to compare the performance of the YOLO and DETR models in traffic object detection. The research process begins with the collection of a dataset that includes various traffic objects, such as vehicles, pedestrians, traffic signs, and traffic lights, obtained from traffic surveillance videos or images. After the dataset is collected, the next step is feature extraction, where important features of the objects in the dataset are analyzed to facilitate the detection process. After the features are extracted, the data is then analyzed using the YOLO and DETR models for object detection. The final stage of this study is the evaluation of the performance of both models using evaluation metrics such as accuracy and processing speed, to assess the model's ability in real traffic surveillance.

2.1 Dataset Collection

The dataset used in this study comes from traffic monitoring videos published by RTMC Ditlantas Polda Maluku Utara through their official YouTube channel which displays CCTV footage from various points in Ternate City. This video records traffic conditions in real-time, showing various objects such as vehicles (cars, motorbikes, trucks), pedestrians, and traffic signs and lights. Figure 1 shows a sample dataset of traffic monitoring videos from the RTMC Polda Maluku Utara YouTube Channel in 2024 and 2025.

In the development of the YOLO and DETR-based object detection models, data was obtained from YouTube videos belonging to the RTMC Ditlantas Polda Maluku Utara which were then extracted into 759 frames of traffic images for the training and testing process. Of these, 607 images were used for training and the remaining 152 images for model testing.

The class distribution in the dataset includes five main categories, namely: car (45%), motorcycle (30%), truck (10%), bus (8%), and person (7%). Annotation was done manually using LabelImg software, with an output format that matches each model. For YOLO, annotations are included in a bounding box format based on normalized center coordinates and dimensions. Meanwhile, for DETR, annotations also include bounding boxes with a similar format, but this model utilizes transformers to calculate the relationship between objects in the image, thus requiring a different approach in data processing.

The criteria for determining the bounding box follow the tight-fitting box principle based on objects that appear intact in the frame. This technique ensures that the detected object will have a bounding box that matches its original shape, reducing the possibility of empty areas around the detected object.

2.2 Pre-processing and Labeling

Data Preprocessing and Labeling are important stages in the object detection process using deep learning models [15], [16]. After the video data is obtained from the YouTube source RTMC Polda Maluku Utara, the

video is converted into a collection of images (frames) that will be used as model input. Each frame then goes through a preprocessing process, namely changing the image size to make it uniform, and improving visual quality to ensure data consistency. After that, labels (annotations) are given to the objects in the image. Each object in the image (such as cars, motorbikes, trucks, pedestrians, traffic signs, and traffic lights) is marked with a bounding box and given the appropriate class label. The result of this stage is an annotation file containing information on the bounding box coordinates and the name of the object class.



Figure 1. Sample Dataset of RTMC Maluku Utara Police YouTube Videos in 2024 and 2025

2.3 Feature Extraction

Feature Extraction is an important process in visual data processing, especially in object detection. The goal is to extract important information from images or videos that can help the model recognize and distinguish existing objects [17], [18]. In the context of this study, feature extraction was performed after video frames from the North Maluku Police RTMC traffic monitoring were successfully taken and annotated. This extraction process aims to identify the visual characteristics of each object (vehicles, pedestrians, and traffic signs), namely shape, color, edge, pattern, and spatial position in the image. In the YOLO model, feature extraction is performed end-to-end using cascaded convolutions that produce feature representations from images directly before bounding box prediction and classification are performed. While in the DETR model, these features are extracted using a CNN backbone (ResNet), then further processed with a transformer architecture to understand the global context and relationships between objects in the image. The results of this feature extraction are the basis for the model to perform object detection accurately, efficiently, and in real-time. This process greatly determines the quality of the model output because the right features will increase prediction accuracy and reduce errors in object classification and segmentation.

2.3 Inference/Object Detection

Inference / Object Detection is the stage where the trained model is used to recognize and detect objects in new images or videos that have never been seen before. At this stage, the input in the form of images or videos will be processed by the object detection model. In the context of this research, two different architectures are used, namely YOLO (You Only Look Once) and DETR (Detection Transformer).

The model will process each image and issue a prediction in the form of a bounding box that shows the position of the object, as well as a class label (such as car, motorbike, pedestrian, and others), and a confidence score that shows the level of confidence of the model in the prediction. Inference is performed on test data to evaluate the model's ability to recognize objects in real situations. The results of this detection are then compared with the ground truth (original label) to calculate the model's performance.

2.5 Performance Evaluation and Analysis

Performance Evaluation and Analysis is the final stage in the research process that aims to measure and compare the performance of the object detection models used, namely YOLO and DETR. At this stage, the inference results of both models are analyzed using evaluation metrics, namely accuracy, precision, recall, and mean Average Precision (mAP). These metrics provide an overview of how well the model recognizes the correct object, avoids detection errors, and maintains prediction consistency. In addition to

accuracy, the evaluation also considers the processing speed (inference time) of each model. This is especially important for real-time applications such as traffic surveillance systems, where fast and accurate detection is needed. Figure 2 below shows an overall overview of the research stages carried out

2.6 Training Configuration

The training configuration for both YOLOv9 and DETR models followed a standardized procedure to ensure a fair comparison. The initial learning rate was set to 0.001, and the Adam optimizer was employed due to its adaptive learning capability. A batch size of 16 was used for both models to maintain memory efficiency and stable gradient updates. All images were resized to a fixed resolution of 640×640 pixels, which is a commonly adopted input size in object detection tasks to balance accuracy and speed. Each model was trained for a total of 50 epochs, with performance evaluations conducted at intervals of 10 epochs (i.e., at epochs 10, 20, 30, 40, and 50). During training, data augmentation techniques such as random horizontal flipping and brightness adjustments were applied to improve model generalization and mitigate overfitting. However, no early stopping or explicit regularization techniques (e.g., dropout or L2 regularization) were used in this study to allow full observation of the learning dynamics over the entire epoch range. The training was conducted using Google Colab, leveraging an NVIDIA Tesla T4 GPU with 16 GB VRAM and 2 vCPUs. This hardware setting provided sufficient computational resources to train both models efficiently while remaining accessible for replication by other researchers.

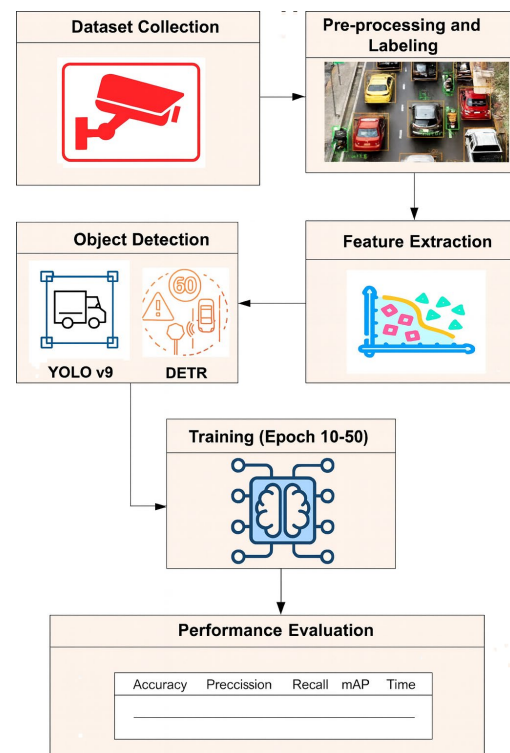


Figure 2. Research Stages Flow

3. Results and Discussions

This section presents the performance evaluation of the object detection models applied to the traffic surveillance dataset. The analysis covers the ability of each model, namely YOLO and DETR, in recognizing various types of traffic objects such as vehicles, pedestrians, traffic signs, and traffic lights. The performance of the models is analyzed based on the

visualized inference results and compared using evaluation metrics, namely accuracy, precision, recall, and F1-score. The experimental process was carried out by training the YOLOv9 and DETR models on the traffic surveillance dataset for 50 epochs, with evaluations carried out at epochs 10, 20, 30, 40, and 50. Each model was tested based on the metrics mAP, precision, recall, F1-score, and inference time. Table 1 shows the evaluation results of both models.

Table 1. Evaluation Results of YOLO and DETR Models

Epoch	Model	mAP (%)	Precession (%)	Recall (%)	F1-Score (%)	Inference Time (s)	Detection
10	YOLOv9	0.80	0.79	0.81	0.87	0.4755	11
10	DETR	0.87	0.90	1.0	0.95	8.7226	22
20	YOLOv9	0.83	0.86	0.84	0.88	0.4016	11
20	DETR	0.89	0.91	1.0	0.97	8.3519	22
30	YOLOv9	0.91	0.93	0.88	0.90	0.4513	11
30	DETR	0.94	0.96	1.0	0.99	7.2581	22
40	YOLOv9	0.97	1.0	0.91	0.97	0.4527	11
40	DETR	1.0	1.0	1.0	1.00	7.6081	22
50	YOLOv9	1.0	1.0	0.95	0.99	0.4314	11
50	DETR	1.0	1.0	1.00	1.00	9.0501	22

Based on the experimental results presented in Table 1, the comparison of the performance of the YOLOv9 and DETR models on various evaluation metrics shows a striking difference. At each epoch, DETR shows better performance in terms of mAP, precision, recall, and F1-score. For example, at epoch 10, DETR has an mAP of 0.87, higher than YOLOv9 which only reaches 0.80. As the number of epochs increases, DETR continues to show an increase in detection accuracy, reaching an mAP value of 1.0 at epochs 40 and 50, while YOLOv9 reaches a maximum mAP of 1.0 at epoch 50. DETR also excels in precision and recall, with precision and recall reaching 1.0 at epochs 40 and 50, while YOLOv9 does not reach these values despite increasing. However, although DETR is superior in terms of detection accuracy, YOLOv9 is faster in inference time. At epoch 10, YOLOv9 has an inference time of 0.4755 seconds, which stabilizes to 0.4314 seconds at epoch 50.

In contrast, DETR takes a longer inference time, starting from 8.7226 seconds at epoch 10 and increasing to 9.0501 seconds at epoch 50. This shows that although DETR is more accurate in detecting objects, YOLOv9 is more efficient in terms of inference speed, making it more suitable for applications that require real-time object detection [19], [20]. Overall, the results of this experiment show that YOLOv9 is superior in terms of inference speed, while DETR provides higher detection accuracy despite longer inference time.

Furthermore, from the results of object detection, each model shows different characteristics and advantages. DETR is able to detect objects with a very high level of accuracy, including in complex conditions such as overlapping objects, small objects, and objects that are far from the camera. Detection by DETR tends to be more complete and detailed, with a consistent number of detections of 22 objects in each epoch.

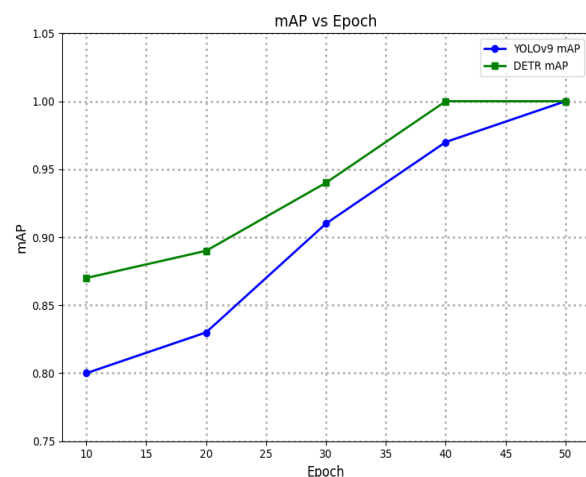


Figure 3. mAP vs Epoch Comparison Chart

This shows that this model is very sensitive to the presence of objects in the image, and is able to identify various categories of objects comprehensively. Meanwhile, YOLOv9 produces faster and more efficient detection, but with fewer objects detected, namely 11 objects in each epoch. Although the accuracy performance increases with increasing epochs, this model still has limitations in detecting small or overlapping objects. The detection results tend to be more selective and sometimes miss objects that are not very clear or are in the background. Visually, the output of DETR shows more precise bounding boxes and consistent labeling, while YOLOv9 shows high detection speed with fairly good results for dominant objects in the image. Based on the experimental results, the DETR model demonstrates superiority in scenarios that require high detection accuracy and comprehensive object coverage. It consistently achieves high metric scores across all evaluation points, making it well-suited for applications that prioritize detection completeness, including small and overlapping objects.

In contrast, YOLOv9 offers significantly faster inference times with relatively competitive accuracy, which makes it more appropriate for real-time traffic monitoring systems where processing speed is critical.

To assess the reliability of these results, statistical analysis was conducted across training epochs. The DETR model exhibited a standard deviation of 0.05 in mean average precision and 0.03 in precision, indicating stable performance throughout training.

Meanwhile, YOLOv9 showed slightly higher variance, reflecting performance fluctuations. The perfect scores (1.0) achieved by DETR in the final epochs may indicate overfitting, especially given the relatively small dataset used in this study. Therefore, further validation using larger and more diverse datasets is recommended to confirm the generalization capability of the models. The statistical summary of these metrics across all epochs is presented in Table 2.

Table 2. Statistical Summary of these metrics across all epochs

Model	mAP Mean	mAP StdDev	Precision Mean	Precision StdDev	Recall Mean	Recall StdDev	F1-Score Mean	F1-Score StdDev
YOLOv9	0.902	0.077	0.916	0.082	0.878	0.050	0.922	0.049
DETR	0.940	0.054	0.954	0.043	1.000	0.000	0.982	0.019

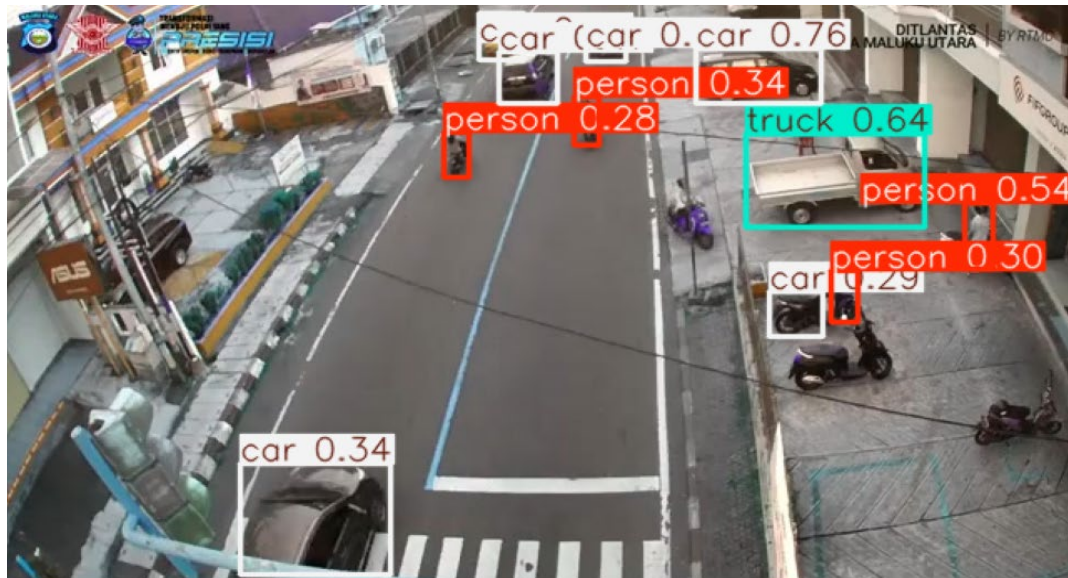
As presented in Table 2, the DETR model exhibits slightly higher average scores across all key performance metrics compared to YOLOv9. It achieves a mean mAP of 0.94 with a relatively low standard deviation (0.05), indicating stable detection performance throughout the training epochs. The recall value for DETR is consistently perfect (1.0) across all epochs, reflecting its high sensitivity to object presence in complex scenes. In contrast, YOLOv9 shows a lower average mAP of 0.902 and higher variability, particularly in precision (standard deviation of 0.082), suggesting fluctuations in its detection consistency. The F1-score standard deviation for YOLOv9 is more than twice that of DETR, further highlighting the trade-off between speed and stability. These statistical results reinforce the suitability of DETR for high-accuracy tasks, and YOLOv9 for fast inference applications.

In addition to the statistical evaluation, visual inspection of the detection outcomes further illustrates the qualitative differences between YOLOv9 and DETR. Figure 4 presents representative detection results from both models on urban traffic scenes, highlighting variations in object completeness, bounding box precision, and detection coverage.

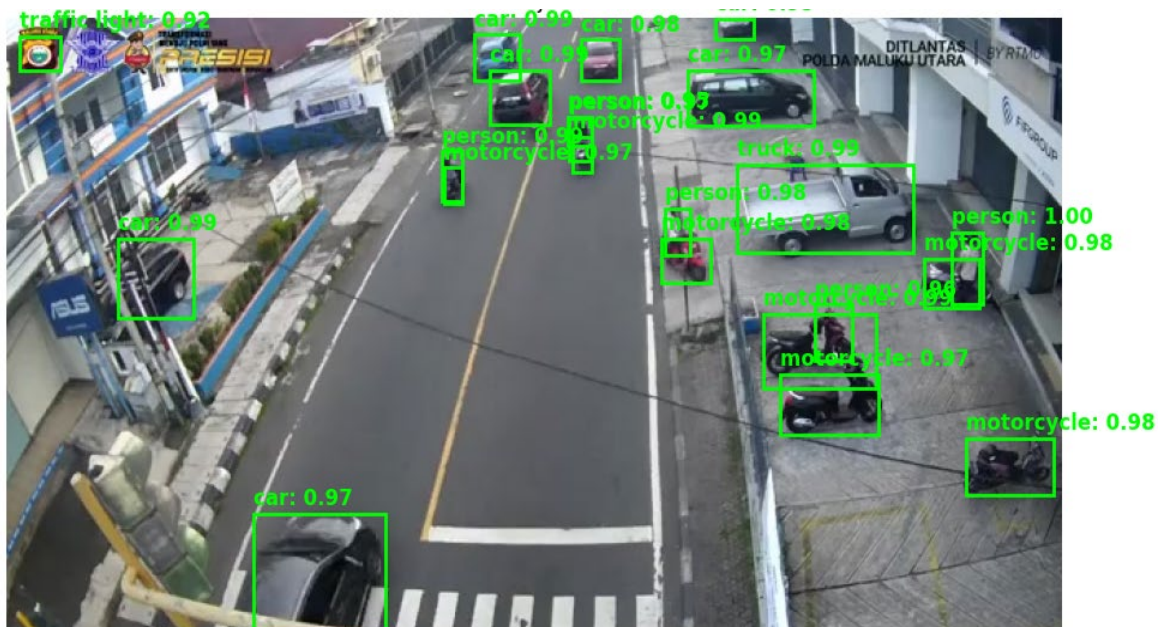
Figure 4 (a) shows the results of object detection from the YOLOv9 model for a traffic situation in an urban area. From the results of visual analysis, several types of objects were detected consisting of a car, a truck, and two people. Cars are scattered along the road—both those that are moving and those parked on the right side of the road. The truck is seen parked in an area near the shops, precisely in the upper right part of the image, and is identified with good precision. Meanwhile, two people were detected around the motorcycle parking area, possibly doing activities such as waiting or talking. The YOLOv9 model shows quite accurate performance in detecting vehicles and humans, with a relatively precise bounding box that follows the shape of the object. Overall, the detection from YOLOv9 in this image is quite effective for use in traffic monitoring systems or vehicle parking pattern analysis, but still

requires further evaluation and possible improvement with other models. Furthermore, Figure 4 (b) presents the results of object detection with DETR.

Based on the results of object detection using the DETR model in the 10th epoch, it can be seen that the model has been able to identify various types of traffic objects quite well. In the image, a number of vehicles such as cars, trucks, and motorcycles are detected, as well as several pedestrians and one traffic sign (traffic light). Object detection is carried out evenly throughout the image area, both those near and far from the camera, demonstrating DETR's ability to understand spatial context effectively. Although the model is still in the early training stage (epoch 10), detection of small and partially covered objects still looks accurate. Objects such as trucks and cars are well recognized, while detection of pedestrians is carried out precisely even though they are far apart or overlapping. This reflects the strength of the DETR architecture in handling visual complexity and relations between objects. In analyzing the detection results, several limitations and error patterns were observed in both models. YOLOv9, although faster, often produced false negatives, especially when detecting small objects or those partially occluded by other elements such as trees or parked vehicles. These missed detections can reduce the reliability of the system in congested traffic scenarios. DETR, while generally more accurate, occasionally generated false positives detecting objects where none existed particularly under poor lighting or low-contrast conditions. Additionally, the transformer-based DETR model showed superior robustness under partial occlusion and complex object interactions due to its global attention mechanism, whereas YOLOv9 tended to overlook objects located in peripheral or cluttered image regions. These discrepancies highlight the trade-off between speed and spatial reasoning capacity inherent in the two architectures. Understanding these behavior patterns is critical when choosing models for deployment in real-world environments with dynamic visibility, occlusion, and background noise [21], [22].



(a)



(b)

Figure 4. Object Detection Results (a) YOLO, (b) DETR

Table 3. Per-Class Object Detection Performance Metrics for YOLOv9 and DETR

Class	YOLOv9 mAP	DETR mAP	YOLOv9 Precision	DETR Precision	YOLOv9 Recall
Car	0.91	0.93	0.92	0.94	0.90
Pedestrian	0.82	0.89	0.80	0.91	0.83
Truck	0.88	0.90	0.90	0.88	0.87
Motorcycle	0.85	0.87	0.86	0.89	0.84
Traffic Light	0.78	0.86	0.75	0.84	0.79

To provide more detailed insight into model performance, a per-class evaluation was also conducted. This approach highlights how each model performs across specific object categories such as cars, pedestrians, trucks, motorcycles, and traffic signs. The results, as presented in Table 3, reveal that DETR excels in detecting small and distant objects like pedestrians

and traffic signs, while YOLOv9 maintains higher precision in detecting vehicles such as cars and trucks. These findings suggest that DETR may be more suitable for scenarios requiring comprehensive scene understanding, whereas YOLOv9 is more appropriate for tasks focused on detecting larger, more prominent objects in real-time applications [23].

4. Conclusions

Based on the evaluation, both YOLOv9 and DETR demonstrated strong capabilities for traffic object detection in intelligent surveillance systems, with each excelling in different dimensions. DETR delivered superior detection accuracy and spatial representation, which can be attributed to its transformer-based architecture capable of modeling global context and resolving object overlaps through bipartite matching. This makes it particularly effective for detecting small, densely packed, or partially occluded objects in complex urban traffic scenes. In contrast, YOLOv9 offered competitive detection accuracy while significantly outperforming DETR in inference speed, a result of its optimized convolutional backbone, dynamic head, and hybrid task-aligned feature fusion that enable faster multi-scale feature processing. Such speed advantages make YOLOv9 highly suitable for real-time applications where rapid decision-making is critical, such as live traffic monitoring or automated incident response. These complementary strengths highlight the potential for a hybrid approach that combines YOLOv9's rapid inference with DETR's robust accuracy and spatial reasoning, thereby creating a more adaptive and resilient traffic object detection system. Future research should expand the dataset with larger and more diverse benchmarks, such as COCO or BDD100K, and conduct cross-city evaluations to enhance robustness. Additionally, investigating model fusion strategies, such as cascaded inference or feature-level integration, could further validate the practicality and performance gains of YOLOv9–DETR hybridization in real-world intelligent surveillance systems.

Acknowledgements

The author declares no conflict of interest. This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

References

- [1] C. Gheorghe, M. Duguleana, R. G. Boboc, and C. C. Postelnicu, 'Analyzing Real-Time Object Detection with YOLO Algorithm in Automotive Applications: A Review', *C. - Comput. Model. Eng. Sci.*, vol. 141, no. 3, pp. 1939–1981, 2024, doi: 10.32604/cmescs.2024.054735.
- [2] J. Le Roux and D. Van Der Haar, 'Analysis of object detection methods to detect traffic flow', *icABCD 2019 - 2nd Int. Conf. Adv. Big Data, Comput. Data Commun. Syst.*, pp. 1–7, 2019, doi: 10.1109/ICABCD.2019.8851054.
- [3] R. Archana and P. S. E. Jeevaraj, *Deep learning models for digital image processing: a review*, vol. 57, no. 1. Springer Netherlands, 2024. doi: 10.1007/s10462-023-10631-z.
- [4] M. Ahmed, N. El-Sheimy, H. Leung, and A. Moussa, 'Enhancing Object Detection in Remote Sensing: A Hybrid YOLOv7 and Transformer Approach with Automatic Model Selection', *Remote Sens.*, vol. 16, no. 1, 2024, doi: 10.3390/rs16010051.
- [5] A. O. Saltık, A. Allmendinger, and A. Stein, 'Comparative Analysis of YOLOv9, YOLOv10 and RT-DETR for Real-Time Weed Detection', 2024, [Online]. Available: <http://arxiv.org/abs/2412.13490>
- [6] Z. Jroni, A. Moussaïd, and M. Y. Hadi, 'Exploring End-to-End object detection with transformers versus YOLOv8 for enhanced citrus fruit detection within trees', *Syst. Soft Comput.*, vol. 6, no. March, p. 200103, 2024, doi: 10.1016/j.sasc.2024.200103.
- [7] Y. Li, N. Miao, L. Ma, F. Shuang, and X. Huang, 'Transformer for object detection: Review and benchmark', *Eng. Appl. Artif. Intell.*, vol. 126, p. 107021, 2023, doi: <https://doi.org/10.1016/j.engappai.2023.107021>.
- [8] A. Corović, V. Ilić, S. Đurić, M. Marijan, and B. Pavković, 'The Real-Time Detection of Traffic Participants Using YOLO Algorithm', in *2018 26th Telecommunications Forum (TELFOR)*, 2018, pp. 1–4, doi: 10.1109/TELFOR.2018.8611986.
- [9] N. N. Hasibuan, M. Zarlis, and S. Efendi, 'Detection and tracking different type of cars with YOLO model combination and deep sort algorithm based on computer vision of traffic controlling', *Sink. J. dan Penelit. Tek. Inform.*, vol. 6, no. 1, pp. 210–221, 2021, [Online]. Available: <https://jurnal.polgan.ac.id/index.php/sinkron/article/view/11231>
- [10] J. Shen, Z. Zhang, J. Luo, and X. Zhang, 'YOLOv5-TS: Detecting traffic signs in real-time', *Front. Phys.*, vol. 11, no. November, pp. 1–11, 2023, doi: 10.3389/fphy.2023.1297828.
- [11] Z. Huang, X. Tao, and X. Liu, 'NAN-DETR: noising multi-anchor makes DETR better for object detection', *Front. Neurobot.*, vol. 18, 2024, doi: 10.3389/fnbot.2024.1484088.
- [12] R. Khoiriyah and A. Hendrawan, 'Vehicle Detection on The Traffic Using Detection Transformer (DETR) Algorithm', *Int. J. Artif. Intell. Sci. (IJ AIS)*, vol. 1, no. 1, pp. 14–24, 2024, doi: 10.63158/IJ AIS.v1.i1.4.
- [13] Y. Wan, H. Wang, L. Lu, X. Lan, F. Xu, and S. Li, 'An Improved Real-Time Detection Transformer Model for the Intelligent Survey of Traffic Safety Facilities', *Sustain.*, vol. 16, no. 23, 2024, doi: 10.3390/su162310172.
- [14] D. Nimma et al., 'Object detection in real-time video surveillance using attention based transformer-YOLOv8 model', *Alexandria Eng. J.*, vol. 118, no. November 2024, pp. 482–495, 2025, doi: 10.1016/j.aej.2025.01.032.
- [15] M. Barulina, A. Andreev, I. Kovalenko, I. Barmin, E. Titov, and D. Kirillov, 'Method for Preprocessing Video Data for Training Deep-Learning Models for Identifying Behavioral Events in Bio-Objects', *Mathematics*, vol. 12, no. 24, 2024, doi: 10.3390/math12243978.
- [16] K. Maharana, S. Mondal, and B. Nemade, 'A review : Data pre-processing and data augmentation techniques', vol. 3, no. April, pp. 91–99, 2022, doi: 10.1016/j.gltp.2022.04.020.
- [17] A. Hassan and M. Sabha, 'Feature Extraction for Image Analysis and Detection using Machine Learning Techniques', *Int. J. Adv. Netw. Appl.*, vol. 14, no. 04, pp. 5499–5508, 2023, doi: 10.35444/ijana.2023.14401.
- [18] P. V. Kumaraguru and V. J. Chakravarthy, 'An Image Feature Extraction and Image Representation Strategies for the Analysis of Image Processing', *Indian J. Forensic Med. Toxicol.*, vol. 11, no. 2, p. 642, 2017, doi: 10.5958/0973-9130.2017.00202.x.
- [19] E. Suherman, B. Rahman, D. Hindarto, and H. Santoso, 'Implementation of ResNet-50 on End-to-End Object Detection (DETR) on Objects', *Sinkron*, vol. 8, no. 2, pp. 1085–1096, 2023, doi: 10.33395/sinkron.v8i2.12378.
- [20] B. Roh, J. W. Shin, W. Shin, and S. Kim, 'Sparse Detr: Efficient End-To-End Object Detection With Learnable Sparsity', *ICLR 2022 - 10th Int. Conf. Learn. Represent.*, pp. 1–23, 2022.
- [21] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, 'You only look once: Unified, real-time object detection', *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-December, pp. 779–788, 2016, doi: 10.1109/CVPR.2016.91.
- [22] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, 'Deformable Detr: Deformable Transformers for End-To-End Object Detection', *ICLR 2021 - 9th Int. Conf. Learn. Represent.*, pp. 1–16, 2021.
- [23] H.-Y. S. Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M. Ni, 'Dino: Detr With Improved Denoising Anchor Boxes for End-To-End Object Detection', *11th Int. Conf. Learn. Represent. ICLR 2023*, 2023, doi: 10.48550/arXiv.2203.03605.