Motorcycle Parking Violation Detection Using YOLOv12 Segmentation and ROI-Guided Orientation Analysis

Muh Fajrin Bakri, Shahnaz Tasha Kurnia, Muhammad Fajar B*, Andi Baso Kaswar, Dyah Darma Andayani, Fhatiah Adiba, Sanatang, Syahrul

Department of Computer Engineering, State University of Makassar, Makassar, Indonesia

Received 25 December 2025; revised 20 March 2026; accepted 26 March 2026

DOI: https://doi.org/10.46604/peti.2026.16020

Abstract

This study aims to improve the accuracy of legal and illegal motorcycle parking classification by proposing a computer vision-based detection system using YOLOv12 segmentation and region-of-interest (ROI)-guided orientation analysis. The proposed system integrates object segmentation, ROI mapping, and computation of the angular deviation between vehicle orientation and parking guides. To determine the optimal orientation tolerance, experiments are conducted using multiple angular thresholds under identical datasets and testing scenarios. The system is evaluated across three locations using confusion-matrix-based performance metrics. The results show that a 30° tolerance yields the best performance, achieving an average accuracy of 87.29%, precision of 93.14%, recall of 92.23%, and F1-score of 92.68%. These findings indicate that ROI-guided orientation analysis enhances the reliability of motorcycle parking violation detection.

Keywords: parking violation detection, YOLOv12, image segmentation, orientation analysis, region of interest

1. Introduction

With the continued growth of two-wheeled vehicles in urban areas, computer vision-based systems for detecting motorcycle parking violations have become increasingly important. In Indonesia, motorcycles dominated registered vehicles in 2024, reaching 139,450,013 units, accounting for 83.7% of the total 166,465,914 vehicles [1-2]. This rapid growth contributes to urban challenges such as traffic congestion, increased safety risks, and reduced service quality in public spaces. Prior studies have indicated that illegal parking and suboptimal parking management disrupt traffic flow, hinder emergency vehicle access, and reduce the efficiency of road infrastructure [1, 3]. Consequently, effective and scalable parking violation detection systems are essential for improving urban traffic management.

Motorcycle parking compliance in dense urban environments is determined not only by whether a vehicle is located inside a designated parking area, but also by whether it is aligned with the intended parking direction. In dense campus environments, motorcycle parking compliance is important not only for maintaining parking order but also for preventing space inefficiency and circulation problems. Parking violations in restricted areas can disrupt parking management, reduce space availability, pose safety risks, hinder emergency access, and interfere with pedestrian pathways. Therefore, automated parking monitoring has practical relevance for smart-campus parking management [1, 3].

Advances in deep learning and computer vision have enabled automated detection of traffic and parking violations, reducing reliance on manual surveillance, which is inherently limited in scalability, consistency, and objectivity [3-4]. Object

* Corresponding author. E-mail address: fajarb@unm.ac.id

detection algorithms such as Faster R-CNN, SSD, RetinaNet, and particularly the YOLO family have demonstrated strong real-time performance in road surveillance and urban environments [5-7]. Among these frameworks, YOLO is widely adopted due to its balance between fast inference speed, detection accuracy, and ease of deployment on resource-constrained devices [5, 8]. Comparative studies show that recent YOLO variants achieve favorable trade-offs in mean average precision (mAP), inference time, model size, and energy efficiency, making them well-suited for intelligent transportation applications [5, 9].

Recent work has further strengthened the use of YOLO-based frameworks for transportation-oriented vehicle detection. For example, Bakirci (2024) evaluated YOLO-based vehicle detection in intelligent transportation system applications. This study highlighted trade-offs related to detection accuracy, inference performance, and deployment challenges in traffic-monitoring scenarios. Although that study focuses on general vehicle monitoring rather than parking compliance, it provides a useful methodological context for the present work, which extends a YOLO-based detection backbone with segmentation-driven orientation analysis for structured motorcycle parking violation detection [10].

The YOLO architecture has evolved through successive improvements in backbone design, feature fusion mechanisms, and detection head optimization, leading to higher mAP and faster inference for traffic and vehicle detection tasks [9]. Recent review studies report that YOLOv8–YOLOv12 can achieve over 80% mAP while exceeding 100 frames per second (FPS) in lightweight variants, supporting real-time deployment in urban environments [9, 11]. In the context of parking management, many YOLO-based systems focus on detecting parking slot occupancy and availability using overhead cameras or closed-circuit television (CCTV). However, these approaches primarily emphasize vehicle presence and slot status. They generally overlook vehicle orientation relative to parking markings or guide lines, which is a critical factor in assessing parking compliance and identifying violations [12-13].

Research specifically addressing motorcycle parking violations remains limited, despite motorcycles being the dominant mode of transportation in many developing countries and regions of Southeast Asia [1, 3]. Most studies on two-wheeled vehicle violations focus on helmet non-use, mobile phone usage, bicycle lane encroachment, or traffic signal violations, whereas parking violations are often treated as a single-class detection problem without sophisticated spatial modeling [3, 14]. This limitation reduces the ability of existing systems to distinguish subtle but impactful forms of parking non-compliance, particularly in structured parking environments.

Makmur et al. proposed a YOLOv7-based motorcycle parking violation detection system that incorporates ROI mapping and object area computation to classify violations based on the percentage of bounding box area entering a prohibited zone. Their approach achieved a recall of 89.7% and an F1-score of 86.2%, demonstrating improved performance compared to methods that only assess object presence within the ROI [1]. While this area-based ROI strategy addresses some limitations of simplistic detection methods, it implicitly assumes that violations depend solely on the geometric location of the vehicle relative to restricted zones, regardless of vehicle orientation with respect to parking guide lines or traffic direction [1, 12].

In real-world scenarios, motorcycles parked at perpendicular or acute angles can obstruct pedestrian pathways or evacuation routes. Even when only a small portion of the vehicle overlaps with the restricted area, such violations may escape detection by area-based ROI methods [1, 3]. Conversely, alternative illegal parking detection approaches classify violations based on vehicles stopping within polygons defined as prohibited zones for durations exceeding predefined thresholds [4]. Although effective for detecting illegal parking on public roads, such duration-based methods are less suitable for structured parking environments, where orientation relative to parking markings is a key indicator of compliance and layout regularity [4, 12, 15].

Moreover, many YOLO and multi-object tracking-based traffic violation studies employ ROI primarily to constrain the analysis area and improve computational efficiency, without explicitly incorporating vehicle orientation angle analysis relative to reference lines into violation classification [16-18]. This creates a notable research gap: while existing systems can detect

vehicle presence and movement with high accuracy, they do not exploit orientation information to differentiate legal and illegal parking in structured parking geometries.

Related studies on small object and vehicle detection from aerial or high elevation perspectives emphasize the importance of comprehensive spatial representation. They also highlight the role of object shape and direction modeling for detecting dense targets such as vehicles in parking areas [8, 13]. Optimized YOLO architectures with multi-scale modules have shown improved sensitivity to object scale and angular variations. Nevertheless, these studies primarily focus on detection accuracy and robustness rather than explicitly leveraging orientation angles to enforce compliance rules, such as correct parking orientation [19-20].

Comparative evaluations of road object detection algorithms further indicate that YOLOv4 and other variants remain competitive in performance. However, domain-specific post-processing tasks, including orientation analysis, angle deviation measurement relative to parking markings, and integration of geometric rules, are often left to system developers and are rarely addressed comprehensively in the literature [5, 21]. This highlights the need for a framework that explicitly integrates YOLO-based real-time detection with geometric analytics tailored to structured motorcycle parking scenarios, where parking behavior exhibits greater variability [1, 3].

Beyond methodological aspects, prior evaluations of motorcycle and two-wheeled vehicle violation detection systems are often limited to specific geographic locations and controlled visual conditions, restricting their generalizability across different parking layouts and camera viewpoints [3, 14]. YOLO-based studies have shown that domain shifts caused by variations in lighting, shadows, and image noise can degrade model performance unless explicitly addressed [22]. These challenges are particularly pronounced in structured motorcycle parking areas characterized by limited space and constrained camera viewing angles [14, 23]. This underscores the need for robust system design and evaluation across diverse operational conditions.

Although many studies report standard performance metrics such as accuracy, precision, recall, F1-score, and mAP, operational parameters such as orientation thresholds or angle deviation tolerances relative to parking guide lines are rarely optimized systematically through comparative experiments across multiple configurations [1, 5]. In practical deployments, inappropriate angle threshold selection can increase false positives or false negatives, thereby reducing system reliability in parking enforcement applications [17, 24].

In this study, the term illegal parking is used as an operational classification in the context of structured motorcycle parking management rather than as a direct legal designation corresponding to a fineable traffic offence. The ground-truth labels for legal and illegal parking were assigned manually based on observed parking compliance with the intended parking alignment at the study sites. Furthermore, the 30° tolerance adopted in this work should be interpreted as an empirically derived threshold. It produced the best classification performance on the annotated dataset, rather than as a legally mandated universal standard. This operational definition enables the proposed system to evaluate parking compliance in a practical and measurable manner within the specific environments examined in this study.

Existing computer vision-based motorcycle parking violation detection systems exhibit three primary limitations: the predominance of area-based ROI approaches with minimal use of vehicle orientation information; the lack of integration of angle deviation analysis relative to parking guide lines within detection pipelines; and constrained evaluations that rarely test orientation thresholds across multiple real-world locations. As highlighted in prior studies [1, 5, 12], these limitations motivate the development of advanced systems that utilize YOLOv12 segmentation with ROI-guided orientation analysis, together with multi-location evaluation and systematic angle threshold optimization, to improve the accuracy and reliability of legal and illegal motorcycle parking classification [1, 9].

Therefore, this study aims to develop a YOLOv12-based motorcycle parking violation detection system for accurate and reliable legal and illegal parking classification across multiple real-world locations. The proposed system integrates object segmentation with ROI-guided orientation analysis to evaluate both the spatial position and orientation alignment of motorcycles within structured parking areas.

2. Research Method

This study employs an experimental approach using computer vision and deep learning to develop an automated motorcycle parking violation detection system. The system is built using the YOLOv12 algorithm for object detection, combined with ROI definition and ROI-guided orientation estimation to detect parking violations. The overall research workflow is illustrated in Fig. 1 and consists of the following stages.

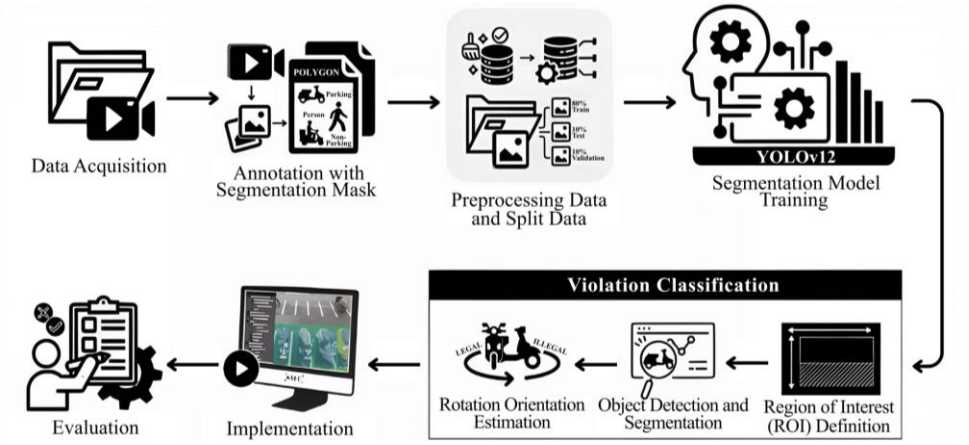


Fig. 1 Research workflow

2.1. Data Acquisition

Motorcycle parking videos were collected from multiple sites using two sources: primary recordings from two locations and a validated secondary dataset from prior work on parking violation detection [18]. The primary videos were captured in daylight using a smartphone camera (1080×2400, 30 fps), totaling 78 min 23 s, while the secondary data contributed 12 min of footage. Detailed data quantities from each location are presented in Table 1.

Table 1 Detailed data quantities

Location	Class			Data Type
	Parking	Non-Parking	Person	
Place A	17,019	1,462	964	Primary
Place B	1,238	752	185	
Place C	26,537	548	2,230	Secondary

The combination of primary and secondary data forms an essential foundation for model training and evaluation, as dataset quality and diversity strongly influence system performance in recognizing objects under a wide range of real-world conditions.

2.2. Annotation with Segmentation Mask

After data collection, an annotation process was conducted using segmentation masks (instance segmentation). Before annotation, the 90 min 23 s video was converted into 5,423 frames using OpenCV at one frame per second. Each frame was then resized to 640 × 480 pixels and saved in JPG format. The results of the data collection process are shown in Fig. 2.

(a) Parking (b) Non-Parking (c) Person

Fig. 2 Object classes for annotation

Object annotation was performed manually on the Roboflow platform using a polygon-based method, assigning labels to three classes: parking, non-parking, and person. Segmentation masks were used to label each object at the pixel level, providing high-precision object boundaries and enabling more accurate and complex object detection compared to traditional bounding box–based methods [25]. Fig. 2 presents examples of annotations for the three classes, where (a) represents a parked motorcycle, (b) shows a motorcycle in motion, and (c) denotes an object labeled as a person.

The annotated data were stored in YOLO format (.txt), where each annotation file contains object class labels along with normalized bounding box coordinates and corresponding segmentation mask coordinates. This format ensures compatibility with the YOLOv12 segmentation training pipeline and enables efficient loading and processing during model training. The use of normalized coordinates allows the model to generalize across different image resolutions and scales. In this study, the annotation files were directly utilized for both training and validation phases without additional conversion. The overall distribution of annotated objects across the three classes is summarized in Table 2.

Table 2 Total annotations object

Class	Annotation Count
Parking	44,794
Non-Parking	2,762
Person	3,379
Total	50,935

2.3. Data Preprocessing and Data Splitting

After annotation, normalization and augmentation were applied to improve training stability and dataset diversity. Augmentation included saturation (−20% to +20%), grayscale (5%), exposure (−15% to +15%), and blur (≤1.5 px), producing 2,488 additional frames and resulting in a total of 7,911 images, as illustrated in Fig. 3.

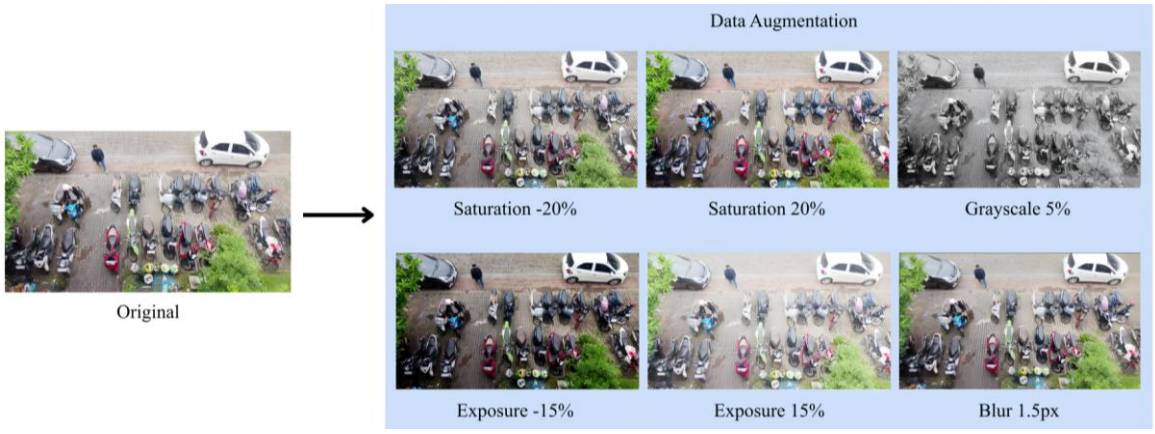


Fig. 3 Illustration of data augmentation using multiple techniques

The processed dataset was divided into three main subsets: training, validation, and testing. This partitioning aims to ensure that the model can learn effectively, be monitored during training, and be evaluated using previously unseen data [26].

The total dataset comprises 7,911 images, with 80% allocated for training (6,329 images), 10% for validation (791 images), and 10% for testing (791 images) [27].

The training dataset was used to enable the model to learn object features and patterns; the validation dataset for tune model parameters such as the learning rate and threshold values; and the testing dataset for final performance under real-world scenarios. Detailed statistics on the number of annotated objects and images for each class in the training, validation, and testing sets are provided in Table 3.

Table 3 Number of objects and images in the training, validation, and testing datasets by class

Class	Total Object	Total Train Image	Total Test Image	Total Val Image
Parking	44,794	6,329	791	791
Non-Parking	2,762			
Person	3,379			
Total Datasets		7,911		

As shown in Table 3, the dataset was imbalanced across classes, with the parking class substantially larger than the non-parking and person classes. This distribution primarily reflects the real-world parking scenes, where parked motorcycles were naturally more prevalent. No explicit class weighting, focal loss, or resampling strategy was applied in the present training configuration. Instead, dataset diversity was partially improved through augmentation and the inclusion of secondary data with richer viewpoints and human-appearance diversity.

Training was conducted on a Windows 11 laptop equipped with an AMD Ryzen 7 4800H processor, 16 GB RAM, and an NVIDIA RTX 3050 GPU. This hardware configuration provided sufficient computational capacity for segmentation-based model training in this study. The selected training parameters were designed to balance model learning stability and computational feasibility under the available resources. In particular, the image size, batch size, and number of epochs were chosen to support efficient training while preserving sufficient visual detail for orientation-based analysis. The hyperparameters applied during training are summarized in Table 4.

Table 4 Hyperparameters used for model training

Hyperparameters	Value
Image size	640
Batch size	8
Epoch	50

The training configuration in Table 4 was selected as a practical experimental setting for the present study, taking into account the available computational resources and the need to maintain stable GPU memory usage during training. The input image size was set to 640 to preserve sufficient visual detail for segmentation-based orientation analysis while remaining computationally feasible on the available hardware. The batch size was set to 8 to ensure safe and stable training on the available GPU, and the maximum number of epochs was set to 50 to allow performance monitoring across the full training process. These settings should therefore be interpreted as study-specific training configurations rather than the outcome of a separate hyperparameter-optimization procedure.

2.4. Segmentation Model Training

This stage describes the core training and inference components of the proposed parking violation detection framework. After dataset preparation, the YOLOv12 segmentation model was trained to identify motorcycles and generate object masks for subsequent spatial analysis. The segmentation output was then integrated with ROI-based filtering and orientation

estimation to support parking status classification. The following subsections explain the ROI construction, detection and segmentation process, and ROI-guided orientation analysis in detail.

2.4.1. Region of Interest (*ROI*)

At this stage, the ROI is defined as a predefined rectangular region that represents the valid motorcycle parking area in each scene. After YOLOv12 detects a motorcycle, the detected object is spatially evaluated against this predefined ROI before further orientation analysis is performed. Using ROI, the system can verify whether a motorcycle's center point lies within a valid parking area before further processing [28]. In this study, the ROI corresponds to the parking boundary area within the roadway region captured in the video. The ROI is defined by pixel coordinates forming a fixed rectangular region, which remains consistent due to the stability of the camera setup. This evaluation is performed by calculating the percentage of the object's bounding box area that overlaps with the valid parking ROI. If less than 50% of the object area lies within the ROI, the object is classified as a parking violation. Otherwise, the system proceeds to orientation analysis.

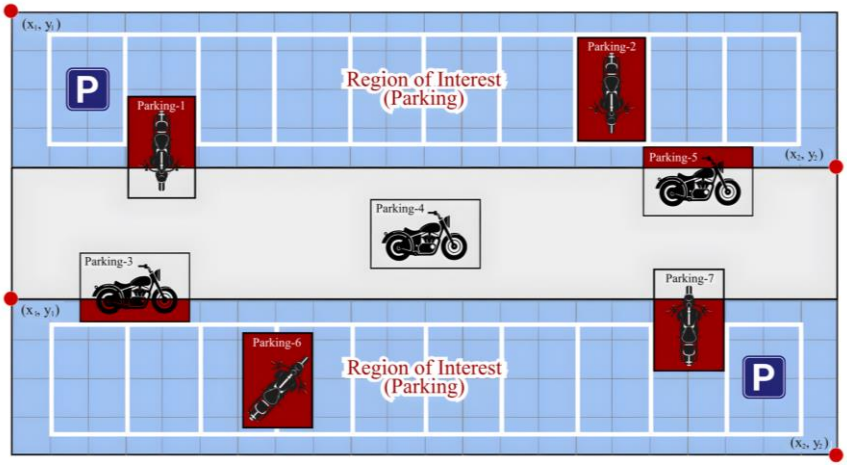


Fig. 4 Illustration of the ROI

In Fig. 4, the ROI is defined as the valid parking area, highlighted in blue, and located on the left and right sides of the main roadway. The coordinates (x_1, y_1) and (x_2, y_2) represent the upper-left and lower-right corner points of the overall ROI. This area includes several designated parking spaces labeled parking 1 to parking 7. However, not all marked spaces are considered valid parking areas. For example, parking 4, which is located in the central traffic lane, is excluded from the ROI because it lies within the vehicle movement area. To mathematically define the spatial boundary of the ROI, the ROI is represented as a four-point polygon, as expressed below:

$$P = \{p_1, p_2, p_3, p_4\} = (x_i, y_i) \tag{1}$$

To determine the orientation of vehicles located within the ROI, the upper and lower edges of the ROI are first identified. The two points with the smallest y-coordinate values are selected as the upper edge, while the two points with the most significant y-coordinate values form the lower edge, as described below:

$$T = \{t_1, t_2\}, y_{t_1}, y_{t_2} = \min(y_1) \tag{2}$$

$$B = \{b_1, b_2\}, y_{t_1}, y_{t_2} = \max(y_1) \tag{3}$$

Next, the ROI is divided into orientation guidelines by applying linear interpolation between the upper and lower edge points. The interpolation of the upper points and the lower points are computed using equations below. Each resulting

orientation guide line G_iconnects the corresponding interpolated upper and lower points can be obtained by:

$$g_i^{top} = t_1 + \frac{i-1}{n-1}(t_2 - t_1) \tag{4}$$

$$g_i^{bottom} = b_1 + \frac{i-1}{n-1}(b_2 - b_1) \tag{5}$$

$$G_i = (g_i^{top}, g_i^{bottom}), i = 1, 2..., n \tag{6}$$

Once a vehicle is detected, the system evaluates whether the parked object violates parking regulations by calculating the percentage of the object's bounding-box area that overlaps with the valid parking ROI. This percentage is computed by the equation below. If less than 50% of the object area lies within the ROI, the object is classified as a parking violation. Otherwise, the object is considered spatially valid and proceeds to orientation analysis.

$$Object\ Area\,(\%) = \frac{Object - ROI\ Intersection}{Total\ Object\ Area} \cdot 100 \tag{7}$$

2.4.2. Object Detection and Segmentation

YOLOv12 is employed to detect objects and produce two primary outputs: bounding boxes and segmentation masks [29]. Bounding boxes provide coarse object localization, whereas segmentation masks provide pixel-level object regions that can be used for contour extraction. In the proposed framework, the segmentation output serves as the geometric basis for orientation estimation. The motorcycle contour is required to determine the farthest contour-point pair, which is then employed to construct the orientation line. This contour-based representation is more appropriate for the current orientation-analysis procedure than relying solely on the rectangular extent of the detected object.

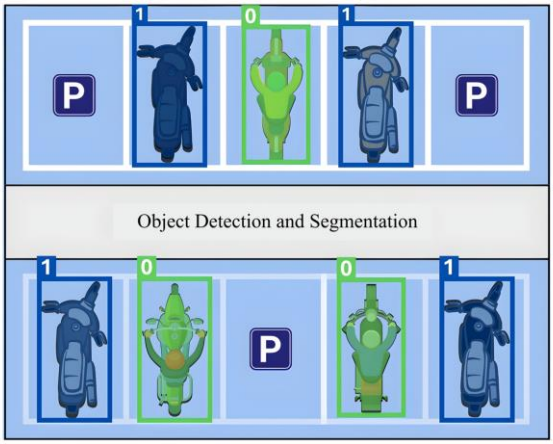


Fig. 5 Illustration of object detection and segmentation

Fig. 5 illustrates the detection and segmentation process, showing how the proposed system identifies motorcycles within the parking area using both bounding boxes and segmentation masks. Bounding boxes provide coarse object localization, while segmentation masks offer finer shape information at the pixel level. This combination is important, as the proposed method relies not only on object presence but also on contour-based orientation analysis. By preserving more detailed object boundaries, the segmentation output supports more accurate estimation of motorcycle direction and enhances the subsequent classification of parking violations.

2.4.3. Rotation Orientation Estimation

This stage evaluates the orientation of a parked motorcycle to determine whether its position complies with parking rules or constitutes a violation. The process begins by identifying the farthest pair of contour points (farthest point pair) on each motorcycle instance, which are then used to construct the vehicle's orientation line. This contour-based procedure requires pixel-level object boundaries obtained from the segmentation mask. It enables the system to estimate a heading that more accurately reflects the motorcycle's actual parking pose in the scene. Conventional axis-aligned bounding boxes were not used as the primary basis for orientation estimation because they mainly represent the outer rectangular extent of the detected object. This representation fails to capture the intrinsic rotational direction of a motorcycle, particularly when the vehicle appears oblique, partially overlapping, or irregular in outline. After the orientation line is obtained, the system computes its inclination angle relative to the horizontal axis by calculating the rotation angle as shown below:

$$\theta = \arctan\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \cdot \frac{180}{\pi} \tag{8}$$

Based on the computed angle, the system evaluates whether the motorcycle's orientation is consistent with the predefined parking direction represented by the ROI guideline. A vehicle is classified as legal parking if its orientation deviation remains within the specified tolerance. Conversely, if the deviation exceeds the threshold, it is categorized as a parking violation. This approach allows the system to assess both the presence of a motorcycle in the parking area and its alignment according to the intended parking layout. In this way, the orientation angle becomes a key decision variable in the final parking status classification.

The overall architecture of the proposed parking violation detection system is presented in Fig. 6. The system comprises three main components: input data, detection process, and detection outputs. The object detection module processes each video frame with YOLOv12 to detect motorcycles and determine whether each object lies within the predefined ROI. If an object is within the ROI, the system verifies whether the vehicle body is fully contained within it. If any part of the object lies outside the ROI boundary, the system directly labels it as a violation. For objects that satisfy the ROI constraint, the system subsequently computes the vehicle's orientation angle and compares it with the ideal direction. A parking violation is flagged if the rotation angle exceeds the predefined tolerance. The detection results are then categorized into three output types: object detection, violation detection, and legal parking for correctly parked vehicles.

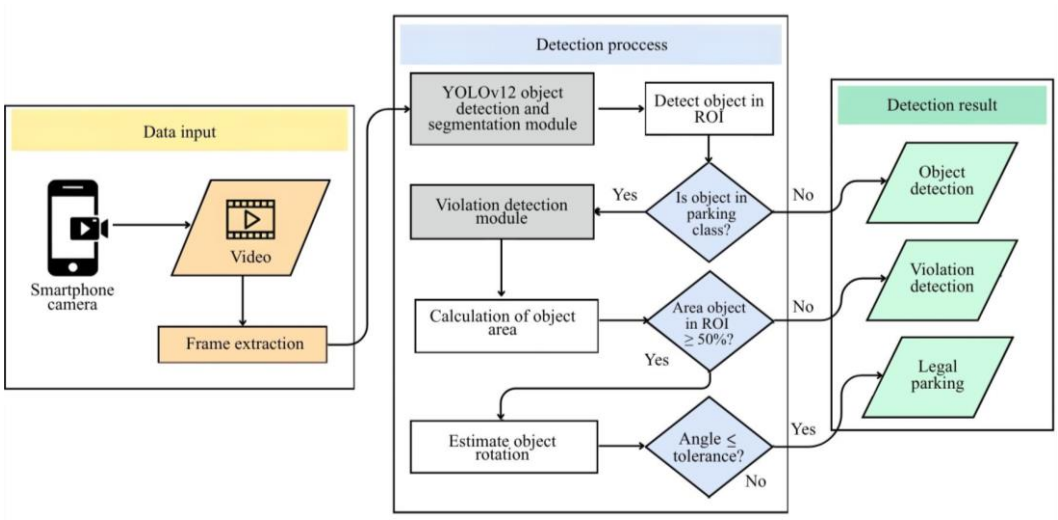


Fig. 6 Overall architecture of the parking violation detection system

2.5. Implementation

In the implementation stage, the developed parking violation detection system was applied to real-world video data collected from three different locations. The objective of this stage is to evaluate the system's effectiveness and reliability under actual environmental conditions. Each video used in the implementation presents diverse parking violation scenarios, as visualized in Fig. 7.

(a) Place A (b) Place B (c) Place C

Fig. 7 Different locations for system implementation

In the present implementation, the system does not process every video frame continuously. Instead, it analyzes selected frames when relevant scene changes are detected, treating each as an individual event for violation assessment. Therefore, the current workflow is better interpreted as an event-based analysis pipeline rather than a conventional full-frame real-time detector. Real-world implementation and inference stages were also executed on the same Windows 11 laptop used for training, equipped with an AMD Ryzen 7 4800H CPU, 16 GB RAM, and an NVIDIA RTX 3050 GPU. The detected parking violations are then visualized by overlaying the vehicle orientation lines along with their corresponding violation status. All results are exported as video files for documentation and further analysis. Detailed information regarding the duration and description of the videos from each implementation location is presented in Table 5.

Table 5 Duration and description of implementation videos

Place	Duration	Description
A	00:20:12	Features the parking area in front of the Graduate School Building at State University of Makassar, recorded from the 2nd floor.
B	00:42:23	Features the parking area in front of Daun Coffee Café, recorded from the 2nd floor.
C	00:11:10	Features the parking lot in front of the Teknol Building of the Department of Informatics and Computer Engineering at the State University of Makassar, recorded from the 3rd floor.

Place A was not included in the training dataset and therefore served as an unseen implementation site. In contrast, the evaluated events at Places B and C were different from the training samples but originated from the same locations used in dataset development. Therefore, although there was no direct frame-level or event-level overlap between training and evaluation events at Places B and C, the present study does not constitute full camera-level separation across all evaluated sites.

2.6. System Evaluation

A comprehensive evaluation was conducted to measure its performance using confusion-matrix-based metrics, including accuracy, precision, recall, and F1-score. The confusion matrix was used to represent classification outcomes in terms of true positive (TP), true negative (TN), false positive (FP), and false negative (FN), as summarized in Table 6.

Table 6 Confusion matrix

	Predicted	
Actual	True negative (TN)	False positive (FP)
	False negative (FN)	True positive (TP)

Based on the confusion matrix in Table 6, four standard classification metrics were calculated to evaluate the effectiveness of the proposed parking violation detection system. Accuracy measures the overall proportion of correct predictions, while precision quantifies the proportion of detected violations that were truly positive. Recall assesses how many actual violations were successfully identified by the system, and the F1-score provides a balanced measure between precision and recall. These metrics were computed below to provide a comprehensive assessment of classification performance.

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \cdot 100\% \tag{9}$$

$$Precision = \frac{TP}{TP + FP} \cdot 100\% \tag{10}$$

$$Recall = \frac{TP}{TP + FN} \cdot 100\% \tag{11}$$

$$F1 - Score = \frac{2 \cdot precision \cdot recall}{precision + recall} \cdot 100\% \tag{12}$$

3. Results and Discussion

The parking violation detection model was designed to classify objects in the motorcycle parking area into three categories: parked motorcycles (parking), moving motorcycles (non-parking), and humans (person). The model was trained with both bounding boxes and segmentation masks to improve detection accuracy, particularly in dense scenes involving object overlap. The prediction results indicate that the model can reliably differentiate the three object categories. The segmentation output captures object shape and boundary details, which are crucial for accurate violation identification in subsequent stages. Examples of model predictions, including object labels and segmentation outputs, are shown in Fig. 8.

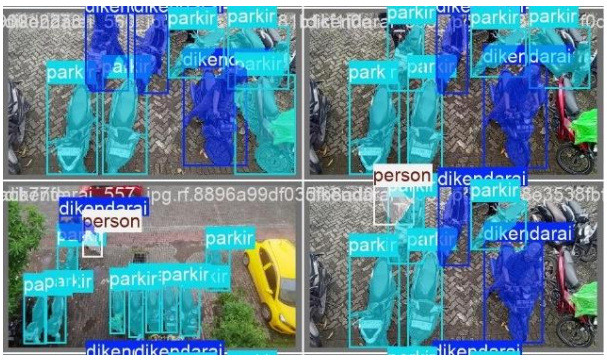


Fig. 8 Prediction results for parking, non-parking, and person objects in a motorcycle parking area

Fig. 8 shows that the model successfully detects and classifies three object categories, namely parking, non-parking, and person, in complex real-world parking scenes. The visual results demonstrate that the segmentation-based approach can capture object boundaries effectively, even under crowded conditions with overlapping motorcycles. In addition, the inclusion of

orientation-varied annotations, as illustrated in Fig. 9, improves the model's robustness across different viewpoints and parking configurations. This combination of detection and orientation-aware annotation supports more reliable classification of parking status in diverse scenarios.

(a) Upward non-parking (b) Upward parking (c) Downward non-parking (d) Downward parking

(e) Left non-parking (f) Left parking (g) Right non-parking (h) Right parking

Fig. 9 Vertical and horizontal orientation variations for parking and non-parking objects

Vertical orientations are divided into upward and downward directions, while horizontal orientations are categorized as left-facing and right-facing. These orientation variations are applied to both the parking and non-parking classes (Fig. 9). After labeling all objects, model training was conducted using the YOLOv12 architecture. Its performance was evaluated under multiple epoch settings, and the results are provided in Table 7.

Table 7 Training experiment results

Test		Epoch				
		10	20	30	40	48
Parking (%)	Precision	0.907	0.921	0.950	0.934	0.972
	Recall	0.978	0.976	0.954	0.970	0.938
	mAP 50	0.983	0.986	0.986	0.987	0.987
	mAP 50:95	0.802	0.830	0.830	0.836	0.845
Non-Parking (%)	Precision	0.888	0.889	0.941	0.930	0.947
	Recall	0.874	0.929	0.916	0.927	0.929
	mAP 50	0.934	0.965	0.971	0.971	0.973
	mAP 50:95	0.664	0.756	0.779	0.785	0.794
Person (%)	Precision	0.779	0.868	0.894	0.901	0.913
	Recall	0.650	0.792	0.781	0.822	0.827
	mAP 50	0.749	0.884	0.909	0.926	0.933
	mAP 50:95	0.435	0.582	0.619	0.647	0.665

Performance improves steadily across epochs, with the most stable results observed at epoch 48. Thus, the epoch-48 model was selected for deployment because it achieved a favorable balance between precision and recall without evident signs of overfitting. As shown in the confusion matrix in Fig. 10, the chosen model achieves high classification performance across non-parking (0.97), parking (0.99), and person (0.92). These results indicate good generalization without significant overfitting.

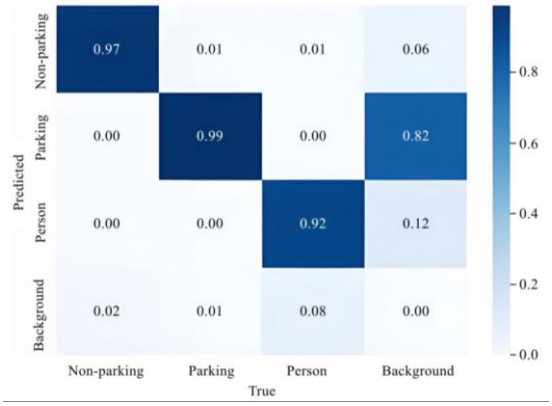


Fig. 10 Confusion matrix of the selected model at epoch 48

The parking class achieves the highest actual positive rate at 99%, followed by non-parking (97%) and person (92%). However, background false positives remain an important issue in the current evaluation. Although some of these errors may be influenced by unlabeled instances during dataset preparation, they still represent practically relevant detection errors from an operational perspective. Therefore, the present results should be interpreted with the understanding that annotation limitations may have contributed to the observed false-positive behavior. Nevertheless, such errors would still affect system reliability in real-world deployment.

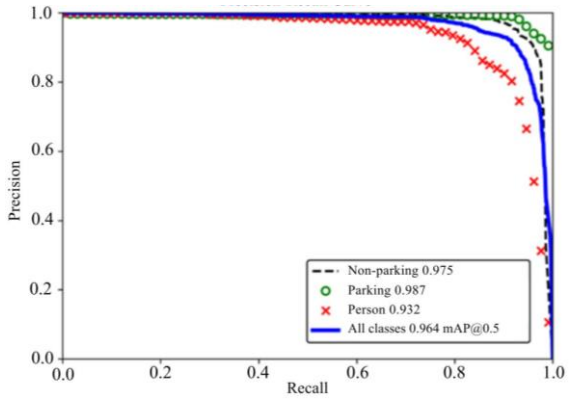


Fig. 11 Precision-recall curve

The experimental results confirm that the model differentiates the three object categories effectively within the evaluated dataset. The segmentation output captures object shape and boundary details that are beneficial for the subsequent violation-analysis stage. As summarized in Fig. 11, the model achieved strong detection performance, with an overall mAP@0.5 of 0.965. The parking and non-parking classes showed particularly strong results, while the person class remained slightly lower. Based on this balanced performance, the epoch-48 model was selected for real-world video implementation testing, as demonstrated in Fig. 12.

Fig. 12 Object detection results using YOLOv12

Real-world testing initially revealed several detection errors, particularly under conditions with limited viewpoint and orientation diversity, as well as insufficient variation in the appearance of human objects. These limitations reduced the model's ability to generalize to more complex visual patterns encountered during real-world deployment. To address this issue,

additional secondary data with richer visual characteristics were incorporated into the training process. The inclusion of these data improved object diversity in terms of viewpoint, pose, and scene composition, which in turn enhanced detection quality. The resulting improvement is illustrated in Fig. 13.

(a) Wrong Parking (b) Correct Parking (c) Wrong Non-Parking

(d) Correct Non-Parking (e) Wrong Person (f) Correct Person

Fig. 13 Detection results before and after adding secondary data

After incorporating secondary data with richer visual characteristics, the refined model was employed for implementation-level testing across three different locations. This stage was intended to evaluate whether the improved training data could support more reliable detection under varied environmental layouts and parking densities. By testing the model in multiple real-world scenes, the study further examined the consistency of object detection, orientation analysis, and parking status classification beyond the original training examples. The implementation results from these three locations are presented in Fig. 14.

(a) Place A (b) Place B (c) Place C

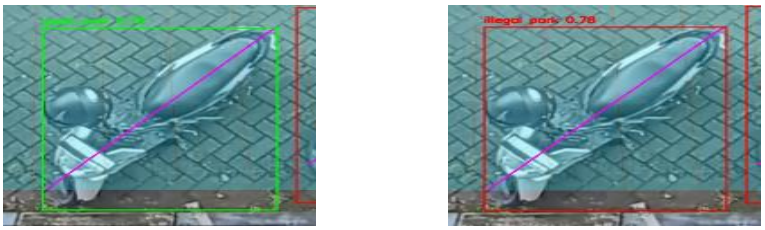
Fig. 14 Implementation results of the motorcycle parking violation detection system in three locations

An additional experiment was conducted to determine the optimal rotation tolerance angle for motorcycle orientation. The evaluation was performed in a dedicated setting to evaluate tolerance under different camera perspective configurations. Video data were extracted into frames at four frames per second, and each frame was compared against the ground truth. Performance comparisons across rotation tolerance settings are provided in Table 8.

Table 8 Performance comparison under different motorcycle rotation tolerances

Angle	System		Error Detection		Recall (%)	Precision (%)	F1-score (%)	Accuracy (%)
	TP	TN	FN	FP				
25°	39	3	0	26	60	100	75	61.76
30°	66	1	0	1	98.50	100	99.24	98.52
35°	63	3	0	2	96.92	100	98.43	97.05
40°	43	3	22	0	100	66.15	79.62	67.64
45°	43	3	22	0	100	66.15	79.62	67.64

The threshold comparison in Table 8 also serves as a sensitivity analysis of the orientation tolerance parameter. The results indicate that 30° provides the best balance between false positives and false negatives under the structured parking conditions examined in this study. However, this value should not be interpreted as a universal threshold for all parking environments. Instead, it should be considered as an empirically derived operational tolerance that is dependent on the parking layout, camera viewpoint, and reference-line configuration used in the present system. As illustrated in Fig. 15, a large tolerance may fail to detect violations, while a tighter tolerance improves illegal-parking detection.



(a) Rotation Within Tolerance (45°) (b) Rotation Without Tolerance (30°)

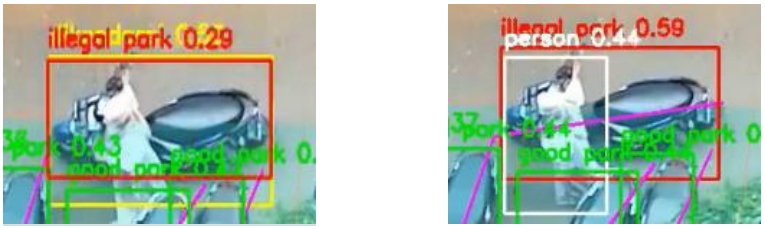
Fig. 15 Comparison of motorcycle orientation classification under different rotation tolerance thresholds (45° vs 30°)

System performance is also influenced by object density and the proximity of motorcycles. Fig. 16 compares incorrect and correct detections for closely spaced motorcycles. In Fig. 16(a), two motorcycles are detected as a single object due to overlap and similar orientations, preventing the system from separating them. In contrast, Fig. 16(b) shows that each motorcycle is correctly detected as an individual object and classified accurately. These cases suggest that the segmentation-based orientation pipeline is useful for distinguishing closely spaced motorcycles and supporting violation classification in crowded scenes. To evaluate the contribution of orientation analysis, a direct quantitative comparison against an ROI area-only baseline was conducted.

(a) Wrong Detection (b) Correct Detection

Fig. 16 Illustration of incorrect and correct detections for closely spaced motorcycles

In real-world conditions, detection errors may also arise from interactions among object classes. Fig. 17 presents examples of dual detection outcomes. In Fig. 17(a), the system fails to detect a person dismounted from a motorcycle, misclassifying the motorcycle as non-parking. This error stems from overlapping bounding boxes that prevent proper separation between the person and the motorcycle. In Fig. 17(b), the person is successfully detected as a separate object, enabling the motorcycle to be correctly classified as a violation, since it is entirely within the ROI. At the same time, a dedicated robustness test under artificially partial masks or controlled segmentation degradation was not included in the present study. Therefore, the discussion of occlusion sensitivity should be interpreted as qualitative evidence from real implementation cases, rather than as a formal robustness benchmark.



(a) Wrong Detection (b) Correct Detection

Fig. 17 Dual detection results

To quantitatively examine the contribution of object orientation analysis, an ROI area-only baseline was evaluated using the same event set and the same predefined parking regions. In this baseline configuration, parking compliance was determined solely based on spatial overlap with the ROI, without using motorcycle orientation information. The resulting event-based performance is presented in Table 9.

Table 9 Event-based performance of the ROI area-only baseline without orientation-based parking analysis

Place	Total Events	System		Error Detection		Recall (%)	Precision (%)	F1-score (%)	Accuracy (%)
		TP	TN	FN	FP				
A	58	42	0	0	16	100	72.41	83.99	72.41
B	38	31	6	0	2	100	93.94	96.88	97.37
C	22	2	7	0	13	100	13.33	23.53	40.91
Total Average						100	70.75	82.86	74.58

As shown in Table 9, the ROI area-only baseline achieved an average accuracy of 74.58%, precision of 70.75%, recall of 100%, and F1-score of 82.86%. These results indicate that spatial ROI filtering alone was able to capture most violation events; however, it produced many false positives, particularly at Place C, where precision and F1-score dropped substantially. This behavior suggests that area-only reasoning is insufficient for structured motorcycle parking scenes where vehicles may remain inside the parking region while still violating the intended parking direction.

Using the selected 30° rotation tolerance, the final evaluation was conducted to assess the overall performance of the proposed system across three different implementation locations. This stage examined whether the integration of ROI-based spatial filtering and orientation analysis could provide reliable parking violation classification under varied real-world conditions. The evaluation was performed at the event level, where each reviewed parking event was compared against the manually assigned ground truth labels. The resulting performance metrics, including accuracy, precision, recall, and F1-score, are summarized in Table 10.

Table 10 Proposed system with orientation

Place	Total Events	System		Error Detection		Recall (%)	Precision (%)	F1-score (%)	Accuracy (%)
		TP	TN	FN	FP				
A	58	36	8	7	7	83.72	83.72	83.72	75.86
B	38	38	0	0	0	100	100	100	100
C	22	21	0	0	1	95.45	100	97.67	95.45
Total Average						92.23	93.14	92.68	87.29

Table 10 presents event-based evaluation results rather than frame-based results. The Total Events column indicates the number of manually reviewed parking events used for final system evaluation at each implementation site. Using the selected 30° tolerance, the proposed system achieved average values of 87.29% accuracy, 93.14% precision, 92.23% recall, and 92.68% F1-score across the three evaluated locations. Location B achieved perfect event-level performance, while Location C also showed high performance, with only one detection error. In contrast, the relatively lower performance at Place A indicates that generalization remains more challenging when the system is applied to a location outside the training dataset.

Among the evaluated sites, Place A represents the most relevant unseen-location case because it was not included in the training dataset. In contrast, the evaluated events at Places B and C were different from the training samples but originated from the same locations used during dataset development, meaning that although there was no direct event-level overlap, full camera-level separation across all sites. Accordingly, the present study should be interpreted as an event-based evaluation with partial unseen-location testing, rather than full camera-level cross-location validation. A limitation of the present study is that the dataset remains restricted to daylight conditions, fixed-camera viewpoints, and limited structural diversity.

4. Conclusions

This study developed a motorcycle parking violation detection framework that combines YOLOv12 segmentation with ROI-guided orientation analysis to classify compliant and violating parking behavior in structured parking environments. By integrating spatial ROI validation and angular deviation measurement, the proposed system extends conventional parking detection beyond area-based reasoning alone. Under the evaluated conditions, a 30° tolerance was identified as the most suitable operational threshold, and the final event-based evaluation across three implementation sites achieved average performance values of 87.29% accuracy, 93.14% precision, 92.23% recall, and 92.68% F1-score. The main conclusions of this study are as follows:

(1) The proposed framework successfully combines segmentation-based contour extraction, ROI validation, and orientation analysis to detect motorcycle parking violations in structured parking areas.

(2) Comparative threshold testing showed that the selected 30° tolerance yielded the most favorable balance for the evaluated parking configuration, indicating that threshold selection is important for reliable classification.

(3) The inclusion of orientation analysis improved performance compared with area-only ROI reasoning, showing that parking compliance cannot be determined solely from spatial overlap.

(4) The system demonstrated promising results across three real-world implementation sites, although performance varied depending on scene complexity and location shift.

(5) The current findings should be interpreted as evidence of method feasibility under controlled daylight and fixed-camera conditions. Broader validation across different parking layouts, camera geometries, and more challenging visual conditions remains necessary for future work.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] H. Makmur, M. Wulandari, M. F. B, A. B. Kaswar, D. D. Andayani, F. Adiba, et al., "Motorcycle Parking Violation Detection System Using YOLOv7 with Region of Interest Mapping and Object Area Calculation," Advances in Technology Innovation, vol. 10, no. 1, pp. 29-43, 2025.

[2] Badan Pusat Statistik, Statistical Yearbook of Indonesia 2024, Jakarta, Indonesia: BPS-Statistics Indonesia, 2024.

[3] R. S. Charran and R. K. Dubey, "Two-Wheeler Vehicle Traffic Violations Detection and Automated Ticketing for Indian Road Scenario," IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 11, pp. 22002-22007, 2022.

[4] S. S. Kathait, A. Kumar, S. Sawal, R. Patidar, and K. Agrawal, "Computer Vision and Deep Learning Based Approach for Violations due to Illegal Parking Detection," International Journal of Computer Applications, vol. 186, no. 70, pp. 9-13, 2025.

[5] B. Mahaur, N. Singh, and K. K. Mishra, "Road Object Detection: A Comparative Study of Deep Learning-Based Algorithms," Multimedia Tools and Applications, vol. 81, no. 10, pp. 14247-14282, 2022.

[6] Y. Zhang, Z. Guo, J. Wu, Y. Tian, H. Tang, and X. Guo, "Real-Time Vehicle Detection Based on Improved YOLO v5," Sustainability, vol. 14, no. 19, article no. 12274, 2022.

[7] E. Güney and C. Bayılmış, "An Implementation of Traffic Signs and Road Objects Detection Using Faster R-CNN," Sakarya University Journal of Computer and Information Sciences, vol. 5, no. 2, pp. 216-224, 2022.

[8] M. Zhou, X. Wan, Y. Yang, J. Zhang, S. Li, S. Zhou, et al., "EBR-YOLO: A Lightweight Detection Method for Non-Motorized Vehicles Based on Drone Aerial Images," Sensors, vol. 25, no. 1, article no. 196, 2025.

[9] J. Wei, A. As'arry, K. A. M. Rezali, M. Z. M. Yusoff, H. Ma, and K. Zhang, "A Review of YOLO Algorithm and Its Applications in Autonomous Driving Object Detection," IEEE Access, vol. 13, pp. 93688-93711, 2025.

[10] M. Bakirci, "Utilizing YOLOv8 for Enhanced Traffic Monitoring in Intelligent Transportation Systems (ITS) Applications," Digital Signal Processing, vol. 152, article no. 104594, 2024.

[11] R. Sapkota, M. Flores-Calero, R. Qureshi, C. Badgujar, U. Nepal, A. Poulose, et al., "YOLO Advances to Its Genesis: A Decadal and Comprehensive Review of the You Only Look Once (YOLO) Series," Artificial Intelligence Review, vol. 58, no. 9, article no. 274, 2025.

[12] A. Ahad and F. A. Kidwai, "YOLO Based Approach for Real-Time Parking Detection and Dynamic Allocation: Integrating Behavioral Data for Urban Congested Cities," Innovative Infrastructure Solutions, vol. 10, no. 6, article no. 252, 2025.

[13] D. P. Carrasco, H. A. Rashwan, M. Á. García, and D. Puig, "T-YOLO: Tiny Vehicle Detection Based on YOLO and Multi-Scale Convolutional Neural Networks," IEEE Access, vol. 11, pp. 22430-22440, 2021.

[14] S. Bose, M. H. Kolekar, S. Nawale, and D. Khut, "LoLTV: A Low Light Two-Wheeler Violation Dataset with Anomaly Detection Technique," IEEE Access, vol. 11, pp. 124951-124961, 2023.

[15] A. Ahad and F. A. Kidwai, "Navigating Parking Choices: Drivers' Perspectives on Parking Guidance and Information Systems: A Survey-Based Case Study," Innovative Infrastructure Solutions, vol. 10, no. 2, article no. 66, 2025.

[16] S. S. Kathait, A. Kumar, S. Sawal, R. Patidar, and K. Agrawal, "Computer Vision and Deep Learning Based Approach for Traffic Violations Due to Over-Speeding and Wrong Direction Detection," International Journal of Computer Applications, vol. 186, no. 66, pp. 7-13, 2025.

[17] N. Sharma, S. Baral, M. P. Paing, and R. Chawuthai, "Parking Time Violation Tracking Using YOLOv8 and Tracking Algorithms," Sensors, vol. 23, no. 13, article no. 5843, 2023.

[18] F. M. Alwafi, B. M. Pratama, P. T. Le, B. Prihasto, and J. C. Wang, "Enhanced Detection of Illegally Parked Vehicles Using YOLO and Good Feature to Track Methods," 2024 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Macau, Macao, 2024.

[19] I. Mir, and K. J. Giri, "SO-YOLOv8: A Novel Deep Learning-Based Approach for Small Object Detection with YOLO beyond COCO," Expert Systems with Applications, vol. 280, article no. 127447, 2025.

[20] X. Luo and X. Zhu, "YOLO-SMUG: An Efficient and Lightweight Infrared Object Detection Model for Unmanned Aerial Vehicles," Drones, vol. 9, no. 4, article no. 245, 2025.

[21] Y. Hou, G. Shi, Y. Zhao, F. Wang, X. Jiang, R. Zhuang, et al., "R-YOLO: A YOLO-Based Method for Arbitrary-Oriented Target Detection in High-Resolution Remote Sensing Images," Sensors, vol. 22, no. 15, article no. 5716, 2022.

[22] H. Wang and H. Qian, "SDG-YOLOv8: Single-Domain Generalized Object Detection Based on Domain Diversity in Traffic Road Scenes," Displays, vol. 87, article no. 102948, 2025.

[23] Y. Qiu, Y. Lu, Y. Wang, and H. Jiang, "IDOD-YOLOV7: Image-Dehazing YOLOV7 for Object Detection in Low-Light Foggy Traffic Environments," Sensors, vol. 23, no. 3, article no. 1347, 2023.

[24] R. Grbić and B. Koch, "Automatic Vision-Based Parking Slot Detection and Occupancy Classification," Expert Systems with Applications, vol. 225, article no. 120147, 2023.

[25] S. E. Ghazouali, Y. Mhirit, A. Oukhrid, U. Michelucci, and H. Nouira, "FusionVision: A Comprehensive Approach of 3D Object Reconstruction and Segmentation from RGB-D Cameras Using YOLO and Fast Segment Anything," Sensors, vol. 24, no. 9, article no. 2889, 2024.

[26] I. E. Tampu, A. Eklund, and N. Haj-Hosseini, "Inflation of Test Accuracy Due to Data Leakage in Deep Learning-Based Classification of OCT Images," Scientific Data, vol. 9, no. 1, article no. 580, 2022.

[27] D. Kuo, Q. Gao, D. Patel, M. Pajic, and M. Hadziahmetovic, "How Foundational Is the Retina Foundation Model? Estimating RETFound's Label Efficiency on Binary Classification of Normal versus Abnormal OCT Images," Ophthalmology Science, vol. 5, no. 3, article no. 100707, 2025.

[28] G. Pradhan, M. R. Prusty, V. S. Negi, and S. Chinara, "Advanced IoT-Integrated Parking Systems with Automated License Plate Recognition and Payment Management," Scientific Reports, vol. 15, no. 1, article no. 2388, 2025.

[29] İ. Çetinkaya, E. D. Çatmabacak, and E. Öztürk, "Detection of Fractured Endodontic Instruments in Periapical Radiographs: A Comparative Study of YOLOv8 and Mask R-CNN," Diagnostics, vol. 15, no. 6, article no. 653, 2025.