SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 3, July 2025
DOI : https://doi.org/10.33395/sinkron.v9i3.15071
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Lightweight YOLO Models for Real-Time Multi-Vehicle Detection

Imam Ahmad Ashari[1)*], Iis Setiawan Mangku Negara[2)], Arif Setia Sandi A[3)]
[1)]Teknologi Informasi, Universitas Harapan Bangsa, Indonesia
[2)]Informatika, Universitas Harapan Bangsa, Indonesia
[3)]Sistem Informasi, Universitas Harapan Bangsa, Indonesia
[1)]imamahmadashari@uhb.ac.id, [2)]iissetiawan@uhb.ac.id, [3)]arifsetiasandi@gmail.com

Submitted : Jul 7, 2025 | **Accepted** : Aug 6, 2025 | **Published** : Aug 10, 2025

Abstract: This study presents a comparative evaluation of three lightweight YOLO architectures: YOLOv5n, YOLOv8n, and YOLO11n for multi-class vehicle detection using CCTV imagery under dense traffic conditions in Semarang, Indonesia. The models were assessed on their ability to detect four vehicle classes: motorcycle, car, bus, and truck. To improve generalization across lighting and environmental variations, six augmentation techniques were applied during training. Among them, Blur augmentation consistently yielded the highest detection accuracy. YOLOv8n with Blur augmentation showed the best performance, achieving a precision of 0.875, recall of 0.655, and mAP@0.5 of 0.756. Larger vehicles such as buses and trucks were detected more accurately, while motorcycles were more difficult due to their smaller size and visual similarity to other objects. Despite the growing use of lightweight YOLO models, few studies have conducted a direct comparison of recent nano variants under real-world urban congestion. This study addresses that gap by providing empirical insights into effective model and augmentation pairings suitable for real-time deployment. The results underscore the importance of architectural selection and data enhancement strategies to improve object detection in resource-constrained settings. These findings offer practical implications for developing intelligent traffic surveillance systems in smart cities.

Keywords: YOLO object detection, vehicle detection, traffic surveillance, data augmentation, real-time detection.

INTRODUCTION

The significant growth in the number of vehicles in urban areas has led to increased traffic congestion, impacting various aspects such as travel delays, air pollution, and transportation efficiency. Traffic congestion reduces vehicle speed, causes incomplete fuel combustion, and increases exhaust gas emissions by two to three times compared to normal conditions (Guo et al., 2020). This situation exacerbates urban air quality and poses negative consequences for public health (Reche et al., 2022).

The integration of CCTV with computer vision techniques has proven effective for real-time traffic monitoring and analysis (Ansh Sakhuja, 2023). Several studies have utilized surveillance cameras to accurately estimate traffic volume and classify vehicles, achieving an error rate of less than 10% (Fedorov et al., 2019). Moreover, real-time traffic control systems leveraging video feeds are capable of instantly calculating vehicle flow density, enabling faster responses during traffic congestion (Gupta et al., 2023). Other research has demonstrated the effectiveness of deep learning in optimizing traffic signal control based on visual CCTV data (Almukhalfi et al., 2024). In addition, AIoT systems powered by CCTV have successfully reduced vehicle waiting times by up to 34% in real-world simulations (Elbasha & M. Abdellatif, 2025). Deep convolutional neural networks have also been employed to automatically count vehicles, detect queues, and monitor stationary vehicles (MANDAL, 2019).

Among single-stage object detection methods, YOLO (You Only Look Once) is widely adopted for its real-time performance and efficient architecture, making it suitable for deployment in traffic surveillance systems (Redmon et al., 2016). YOLO-based models have been successfully used to detect and classify vehicles in CCTV footage, enabling faster incident response and improved traffic flow management (Karim et al., 2024). Recent versions such as YOLOv5n, YOLOv8n, and YOLO11n offer lightweight design and fast inference speeds, making them highly relevant for implementation in dense traffic scenarios with limited computational resources (Khanam et al., 2025). However, their comparative performance under real-world urban conditions, especially in Indonesia, remains underexplored.

*Imam Ahmad Ashari

e-ISSN : 2541-2019
p-ISSN : 2541-044X

This study contributes by providing a comprehensive evaluation of three Ultralytics YOLO models (YOLOv5n, YOLOv8n, YOLO11n) in the context of multi-vehicle detection under dense traffic conditions. While previous research has explored YOLO's application in traffic monitoring, to the best of our knowledge, no existing study has systematically compared the nano versions of these models within Indonesian urban settings using real CCTV data. Furthermore, this study uniquely focuses on the impact of different data augmentation techniques in enhancing model performance. The analysis emphasizes comparative detection accuracy (mAP) and runtime latency, with implications for real-time, edge-based traffic management systems.

LITERATURE REVIEW

Automatic vehicle detection plays a vital role in intelligent transportation systems, real-time surveillance, and urban traffic management. YOLO (You Only Look Once) is one of the most widely adopted deep learning models for this task because it can detect objects in a single pass with high speed and accuracy (Redmon et al., 2016). As Ultralytics continuously improves the YOLO family, including YOLOv5, YOLOv8, and YOLOv11, researchers have evaluated their performance in various scenarios. For example, YOLOv5 enhanced with Flip Mosaic augmentation improved small object recognition, resulting in a 0.5 percent increase in mean Average Precision (Zhang et al., 2022). YOLOv5 has been applied for multi-vehicle detection and tracking, achieving a mean Average Precision (mAP) of 99.39 percent (Tchanchou et al., 2022). Its robustness under rainy and crowded conditions has also been demonstrated in challenging environments (Lin & Zhu, 2024), while further optimization for fisheye camera input led to a 13.7 percent improvement in mAP.

However, most existing studies have focused on a single YOLO version or were conducted under relatively simple traffic conditions. An ensemble approach was proposed to enhance YOLOv5 performance in congested traffic, resulting in a mAP50 of 0.458 (Rahman et al., 2022). Further investigations into YOLOv7 and YOLOv11 revealed improvements in detecting occluded or small vehicles and achieving faster inference times (Hong et al., 2024). Lightweight models based on YOLOv5s architecture and enhanced with attention modules were developed to maintain high detection accuracy while minimizing model complexity (Rana et al., 2024) (Wang et al., 2024). Despite these advancements, no prior study has systematically compared YOLOv5n, YOLOv8n, and YOLO11n under real-world dense traffic conditions in Indonesia, particularly with attention to the role of augmentation techniques. A summary of previous studies on YOLO-based vehicle detection, including the models used, datasets, traffic conditions, and reported outcomes, is presented in Table 1.

Table 1. Comparison of Previous Studies on YOLO-based Vehicle Detection

Study	Model	Dataset / Input	Traffic Condition	Result (mAP or mAP50)	Notes
(Zhang et al., 2022)	YOLOv5 + Flip Mosaic	Custom dataset (small objects)	Moderate	+0.5% mAP	Improved small object detection
(Tchanchou et al., 2022)	YOLOv5	Multi-vehicle dataset	Urban, medium traffic	99.39% mAP	High accuracy for tracking
(Lin & Zhu, 2024)	YOLOv5	Rainy, crowded conditions	Challenging	Substantial improvement	Adapted for adverse environments
(Rahman et al., 2022)	YOLOv5 (Ensemble)	Congested roads	Heavy traffic	0.458 mAP50	Ensemble approach
(Hong et al., 2024)	YOLOv7	Traffic surveillance	City traffic	Not specified	Suitable for traffic systems
(Rana et al., 2024)	YOLOv5s + attention modules	Lightweight custom config	Not specified	High accuracy	Fewer parameters with strong performance

The studies summarized above confirm the relevance of YOLO-based models for vehicle detection. However, several limitations remain. Many studies focus on a single version of YOLO and are often conducted under specific or limited traffic scenarios. Very few investigations address the challenges of dense traffic, occlusion, and lighting variation in urban environments. In addition, while lightweight versions such as YOLOv5n, YOLOv8n, and YOLO11n are increasingly used for edge computing, there is a lack of comparative evaluation among them under real-world conditions. Most prior research also does not analyze the impact of various augmentation techniques, which are crucial for generalization in uncontrolled settings. Therefore, this study fills an important gap by comparing the performance of three lightweight YOLO models under dense traffic conditions in Semarang, Indonesia, while also evaluating the contribution of six different augmentation methods. The results aim to guide practitioners in selecting optimal YOLO models for practical traffic surveillance applications.

*Imam Ahmad Ashari

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 3, July 2025
DOI : https://doi.org/10.33395/sinkron.v9i3.15071
e-ISSN : 2541-2019
p-ISSN : 2541-044X

METHOD

YOLO Model Training and Testing Process

To evaluate the performance of YOLO-based models in vehicle detection tasks, a structured training and testing pipeline was implemented. This process involved a series of steps including data preparation, annotation, model training, and validation. The overall workflow adopted in this study is illustrated in Figure 1. The models selected for evaluation are YOLOv5n, YOLOv8n, and YOLO11n. These models were chosen based on their lightweight architecture, fast inference speed, and growing adoption in edge computing applications. YOLOv5n serves as a widely used baseline with efficient performance in real-time tasks. YOLOv8n introduces anchor-free detection and improved flexibility in feature extraction. YOLO11n is the latest release from Ultralytics, offering enhancements in latency optimization and small object detection. The comparative evaluation of these three versions aims to determine the most suitable model for real-world vehicle detection under dense traffic conditions, particularly in settings with limited computational resources.

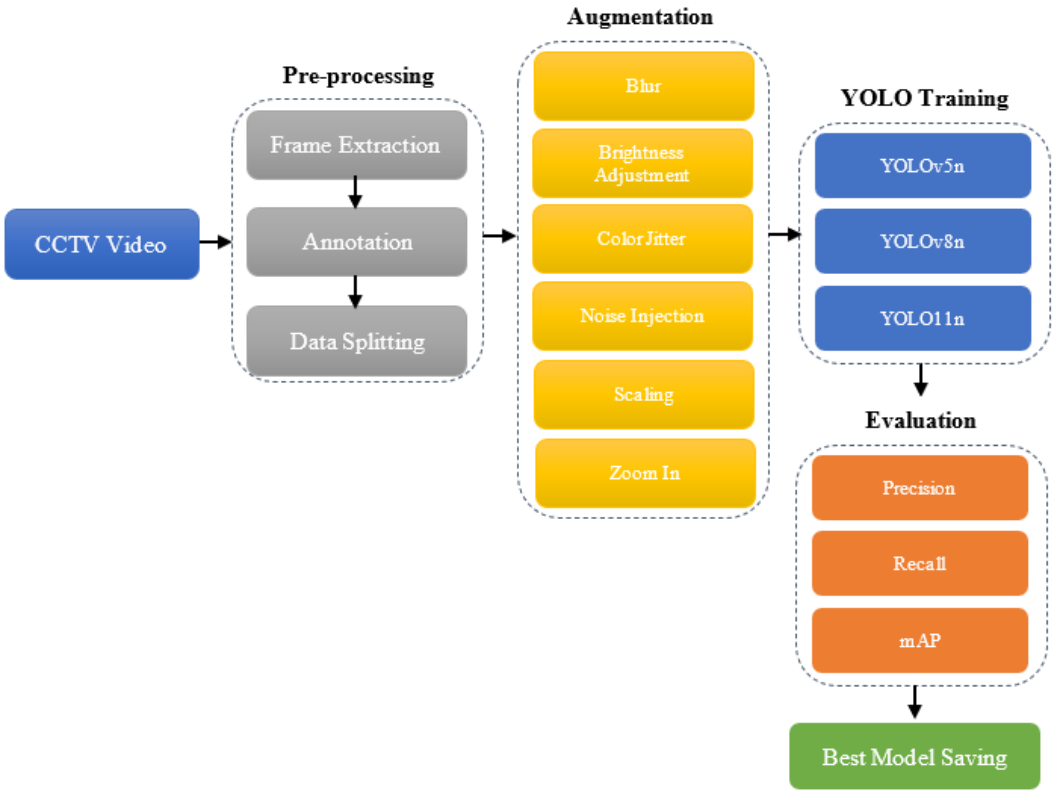

Fig. 1 YOLO Training Workflow

Figure 1 illustrates the YOLO training workflow, which begins with the data pre-processing stage. This step includes extracting frames from CCTV recordings, annotating vehicle objects within each frame, and splitting the dataset into training and validation sets. These procedures are essential to prepare the data in a format compatible with object detection models. To enhance model generalization and dataset diversity, various data augmentation techniques were applied, such as rotation, flipping, brightness adjustment, and other composite transformations.

The object detection models were trained using three versions of Ultralytics YOLO: YOLOv5, YOLOv8, and YOLOv11. These versions were selected to enable comparative analysis of model performance in detecting vehicles under dense traffic conditions. Training was conducted using the pre-processed and augmented datasets, and model performance was evaluated using metrics such as Precision, Recall, mAP50, and mAP50-95. The best-performing model from each version was saved in the best.pt format for deployment in the testing phase.

In the testing phase, the best model (best.pt) was employed to perform inference on the test dataset. The evaluation was carried out under various real-world conditions, including bright lighting, small objects located at a distance from the camera, and high traffic density. The inference results were assessed using the same performance metrics as in the training phase to measure the model's ability to accurately detect vehicles under practical conditions.

*Imam Ahmad Ashari

SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 3, July 2025
DOI : https://doi.org/10.33395/sinkron.v9i3.15071
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Data Augmentation Strategies for YOLO Models

To enhance the performance of YOLO models in detecting objects under various lighting conditions, object sizes, and image qualities, several data augmentation techniques were employed. The primary goal of these techniques is to generate more diverse training samples, thereby improving the model's robustness to real-world scenarios. The augmentation methods applied in this study include blur, brightness adjustment, color jitter, noise injection, scaling, and zoom-in. Each technique was designed to simulate different environmental conditions and image qualities likely encountered in practice, ultimately aiming to improve model generalization. The parameters used for each augmentation technique are summarized in Table 2.

Table 2. Data Augmentation Parameters

No	Augmentation	Value	Augmentation Factor		Reference
			1	2	
1	Blur	Kernel Size	1	2	(Khan et al., 2023)
2	Brightness Adjustment	Brightness Factor	0.8	1.2	(Raimondo et al., 2024)
3	Color Jitter	Brightnes	Rand(0.6, 1.4)	Rand(0.6,1.4)	(Pei et al., 2023)
		Contrast	Rand(0.6, 1.4)	Rand(0.6, 1.4)	
		Saturation	Rand(0.6, 1.4)	Rand(0.6, 1.4)	
		Hue	Rand(-25.5, 25.5)	Rand(-25.5, 25.5)	
4	Noise Injection	Gaussian Noise	Rand(0,0.1)	Rand(0,0.1)	(Eldeen et al., 2022)
5	Scaling	Scale Image	Rand(0.8, 1.2)	Rand(0.8, 1.2)	(Awaluddin & Chao, 2023)
6	Zoom In	Zoom In	1.2	1.5	(Firdaus & Arif, 2023)

In this study, each augmentation technique was implemented using two parameter variations to generate more diverse training samples and further strengthen the generalization capability of the YOLO model. The first technique, blur, was applied with kernel sizes of 1 and 2. These values simulate various degrees of image blurriness to reflect conditions such as motion blur or camera defocus.

Brightness adjustment was performed using brightness factors of 0.8 and 1.2, allowing the model to adapt to both low-light and overexposed environments. For color jitter, random alterations were made to several color components. Brightness, contrast, and saturation factors were randomly varied within the range of 0.6 to 1.4, while the hue shift was randomly applied within the range of -25.5 to 25.5. This combination helps the model recognize objects under diverse color distortions caused by lighting, camera sensors, or environmental influences.

Noise injection was implemented by adding Gaussian noise to the images, with noise levels randomly selected between 0 and 0.1. This approach improves the model's tolerance toward poor-quality or visually noisy images. Scaling augmentation was performed by resizing objects in the images using a random scale between 0.8 and 1.2. This simulates the variability in object sizes due to differences in distance from the camera. Lastly, zoom-in augmentation was applied by enlarging the image with scaling factors of 1.2 and 1.5, simulating conditions in which objects are very close to the camera and partially visible within the frame. By introducing parameter variations across all augmentation techniques, the YOLO model is expected to achieve greater robustness and adaptability when confronted with diverse real-world data conditions. The augmentation workflow is illustrated in Figure 2.

Each augmentation technique was executed twice for the entire training set, once for each parameter variation. As a result, two additional sets of augmented images were generated, each containing 576 images, totaling 1,152 new samples. When combined with the original 576 unaugmented images, the final dataset used for YOLO model training consisted of 1,728 images. This expansion significantly increases the diversity of the training data space, aiming to enhance the model's ability to generalize across variations in lighting, object size, and image quality. By employing this strategy, the YOLO training process does not rely solely on raw data but is enriched with augmented images that simulate real-world scenarios, such as motion-induced blur, variable lighting conditions, visual noise, and partial zoom-in on objects. This comprehensive approach is expected to improve overall object detection performance, particularly on more complex test data.

*Imam Ahmad Ashari

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 3, July 2025
DOI : https://doi.org/10.33395/sinkron.v9i3.15071
e-ISSN : 2541-2019
p-ISSN : 2541-044X

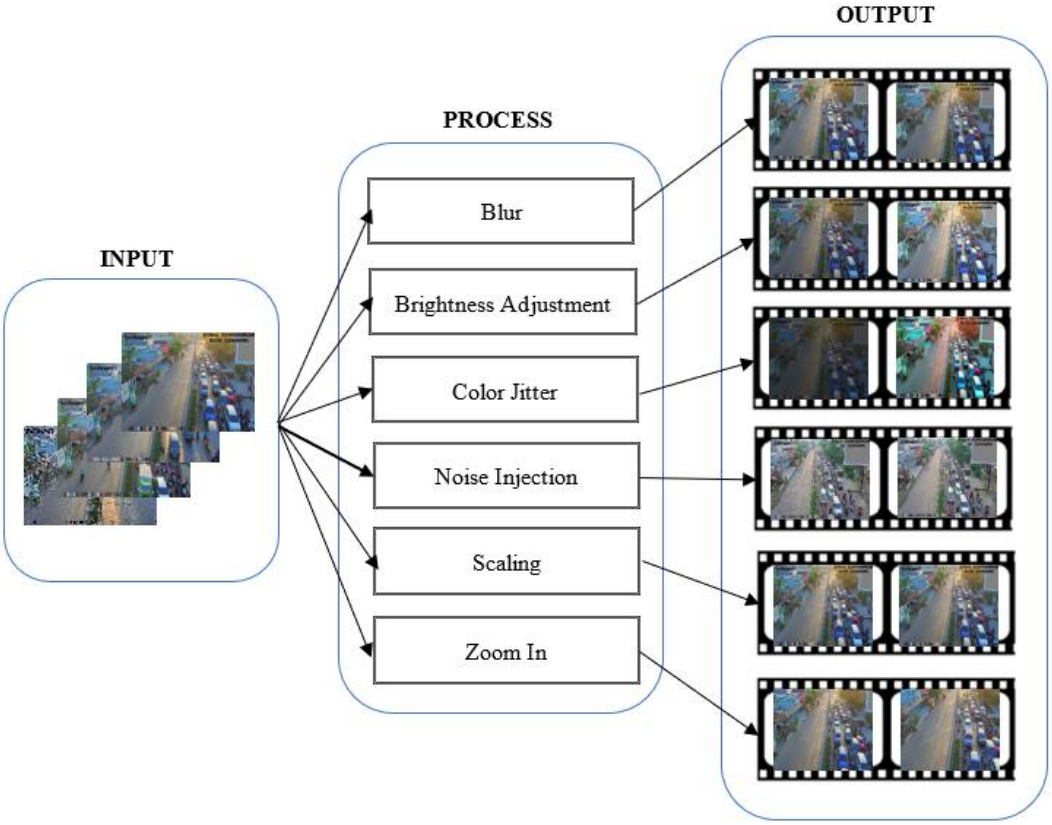

Fig. 2 Augmentation Workflow

Dataset and Annotation Process

The dataset used in this study was derived from CCTV video frame extractions captured in the Fatwamati area of Semarang, Indonesia. A total of 720 images were obtained, comprising 45,347 annotated objects across four vehicle categories: motorcycle, car, truck, and bus. Detailed annotation statistics revealed a dominance of motorcycle objects, with 31,481 annotations, followed by 12,402 for cars, 1,184 for trucks, and 280 for buses.

For model training and evaluation purposes, the dataset was split into two subsets: 80% for training and 20% for validation. In the training set, annotations consisted of 22,136 motorcycles, 8,804 cars, 839 trucks, and 199 buses. The validation set contained 9,345 motorcycle annotations, 3,598 for cars, 345 for trucks, and 81 for buses. This split ensures that the model learns from the majority of available data while being evaluated on previously unseen data to assess its generalization performance.

Object annotations were manually performed using the Roboflow platform, which provides a web-based interface for frame-by-frame object labeling. During the annotation process, users manually drew bounding boxes around each visible vehicle in a frame and assigned appropriate category labels. An example of an annotated frame is presented in Figure 3.

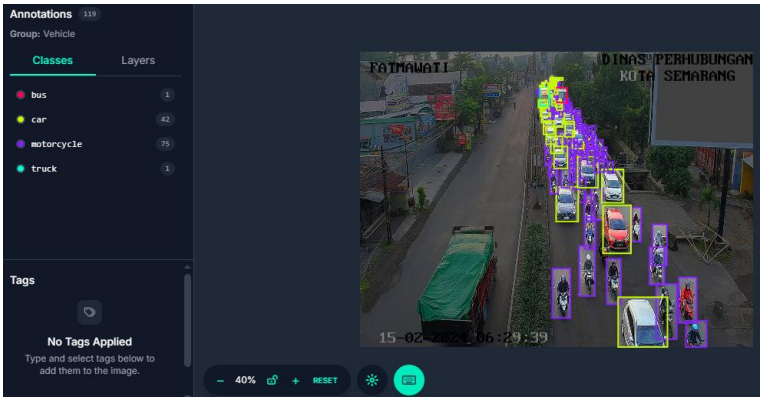

Fig. 3 Image Annotation Example

*Imam Ahmad Ashari

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 3, July 2025
DOI : https://doi.org/10.33395/sinkron.v9i3.15071
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Figure 3 presents the results of manual image annotation using color-coded bounding boxes for each vehicle category: purple for motorcycles, yellow for cars, and teal green for trucks. In this representative frame, captured during peak traffic conditions at 06:29 AM local time (WIB), a total of 75 motorcycles, 42 cars, 1 bus, and 1 truck were manually labeled. This image serves as a representative example of the annotation process applied to frames with high vehicle density.

Experimental Environment

The model training and testing procedures in this study were conducted in a cloud-based environment using Google Colab Pro, which offers access to high-performance GPUs such as the NVIDIA Tesla T4. This computational setup is well-suited for deep learning tasks, particularly for training object detection models at scale. Google Colab Pro is also integrated with Google Drive, allowing efficient storage, loading, and retrieval of datasets and trained models.

The programming environment was based on Python version 3.10+, which is the current default in Colab Pro. Model training was carried out using the Ultralytics YOLO library, an official and streamlined Python API implementation of the YOLO (You Only Look Once) architecture. This library enables training, validation, and inference with just a few lines of code and supports multiple YOLO versions, including YOLOv5, YOLOv8, and YOLOv11. In addition to the Ultralytics package, PyTorch (torch) was used as the primary deep learning backend. PyTorch was responsible for tensor computations, model optimization, and GPU memory management during training and evaluation processes.

YOLO Model Performance Evaluation

The performance of the YOLO models was assessed using four standard evaluation metrics: Precision, Recall, mean Average Precision at IoU threshold 0.5 (mAP@50), and mean Average Precision across IoU thresholds from 0.5 to 0.95 (mAP@50–95). These metrics are widely adopted in object detection benchmarks and research, such as COCO (Common Objects in Context) and PASCAL VOC challenges. They provide a comprehensive assessment of the model's ability to accurately detect and classify objects, especially in complex urban traffic scenarios.

Precision measures the proportion of true positive predictions among all positive predictions made by the model, essentially quantifying how accurate the model is in identifying actual objects present in the image. The formulation for Precision is provided in Equation 1.

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

where TP (True Positive) refers to the number of correctly detected objects, and FP (False Positive) denotes the number of detected objects that do not exist in the ground truth annotation. Meanwhile, Recall measures the proportion of actual objects that are correctly detected by the model from the total number of ground truth objects. The formula for Recall is presented in Equation 2.

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

In this context, FN (False Negative) represents the number of objects that are present in the image but were not detected by the model. Precision and Recall often exhibit a trade-off, where improving one may lead to a decline in the other. Therefore, a comprehensive evaluation using mean Average Precision (mAP) is essential.

Mean Average Precision (mAP) is a key metric in object detection evaluation. It represents the average area under the precision-recall curve across all object classes. In the context of YOLO, two common variants of this metric are mAP@50 and mAP@50–95. The mAP@50, also referred to as mAP@IoU=0.50, computes the average precision when the Intersection over Union (IoU) threshold is set at 0.50. This means that a predicted bounding box is considered correct if it overlaps with the ground truth box by at least 50%.

In contrast, mAP@50–95 is the average mAP calculated over ten IoU thresholds ranging from 0.50 to 0.95 in increments of 0.05. This metric provides a more rigorous and comprehensive assessment of the model's detection performance. The general formula for Average Precision (AP) per class, which corresponds to the integral of the precision-recall curve, is shown in Equation 3, and mAP is obtained by averaging the AP scores across all classes.

$$\text{mAP} = \frac{1}{n}\sum_{j=1}^{n} AP(j) \quad (3)$$

*Imam Ahmad Ashari

e-ISSN : 2541-2019
p-ISSN : 2541-044X

Validation and Generalization Strategy

To ensure the robustness and generalizability of the trained models, this study employed a validation scheme using a dedicated testing dataset that was strictly separated from the training and validation phases. The test data consisted of annotated images representing a wide range of real-world traffic scenarios in urban environments. These scenarios were selected to reflect common challenges encountered in actual CCTV surveillance systems, such as lighting variability, weather disturbances, traffic density, and class diversity.

The evaluation was conducted across ten distinct test conditions, each designed to simulate a specific combination of traffic flow, lighting condition, weather, and vehicle type distribution. The conditions are as follows:

1. Heavy traffic with light reflection, simulating overexposed or glare-affected daytime conditions.
2. Heavy traffic with proper lighting exposure, representing ideal surveillance conditions during the day.
3. Smooth or low-density traffic with light reflection, to evaluate performance when vehicles are sparse but lighting is suboptimal.
4. Smooth traffic with proper exposure, used to observe detection behavior under optimal low-congestion settings.
5. Congested traffic during rainfall, which tests model resilience under adverse weather conditions.
6. Smooth traffic during rainfall, where visibility is reduced but congestion is minimal.
7. All vehicle classes present, heavy traffic with light reflection, to assess multi-class detection when lighting causes visual interference.
8. All vehicle classes present, heavy traffic with proper exposure, considered the most comprehensive high-load condition under good visibility.
9. All vehicle classes present, smooth traffic with light reflection, to examine detection when class diversity is high but lighting is challenging.
10. All vehicle classes present, smooth traffic with proper exposure, representing the ideal baseline for evaluation.

This multi-condition testing strategy allows for a comprehensive assessment of each model's ability to generalize across real-world variations. By including both controlled and adverse conditions—such as rain and glare—this study ensures that performance metrics reflect practical deployment challenges. The results from each scenario contribute to understanding the strengths and limitations of each YOLO variant under different environmental and operational constraints.

RESULTS

Object Detection Results Using YOLO Models

Object detection was performed using three versions of the YOLO (You Only Look Once) model: YOLOv5, YOLOv8, and YOLOv11, with each utilizing the lightweight nano variant (YOLOv5n, YOLOv8n, and YOLO11n). The dataset consisted of frames extracted from CCTV footage recorded in the Fatwamati area of Semarang, Indonesia. The test frames depict dense traffic conditions involving various vehicle types, including motorcycles, cars, buses, and trucks.

Upon completion of the training process, each model was evaluated on the validation set for its ability to detect vehicle objects. Detection outputs were visualized using colored bounding boxes according to object labels: purple for motorcycles, yellow for cars, teal green for trucks, and blue for buses. Sample results from the best-performing YOLOv5n, YOLOv8n, and YOLO11n models are presented in Figure 4. The selected test frame represents a traffic congestion scenario captured under normal lighting conditions and includes all four observed vehicle classes.

*Imam Ahmad Ashari

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 3, July 2025
DOI : https://doi.org/10.33395/sinkron.v9i3.15071
e-ISSN : 2541-2019
p-ISSN : 2541-044X

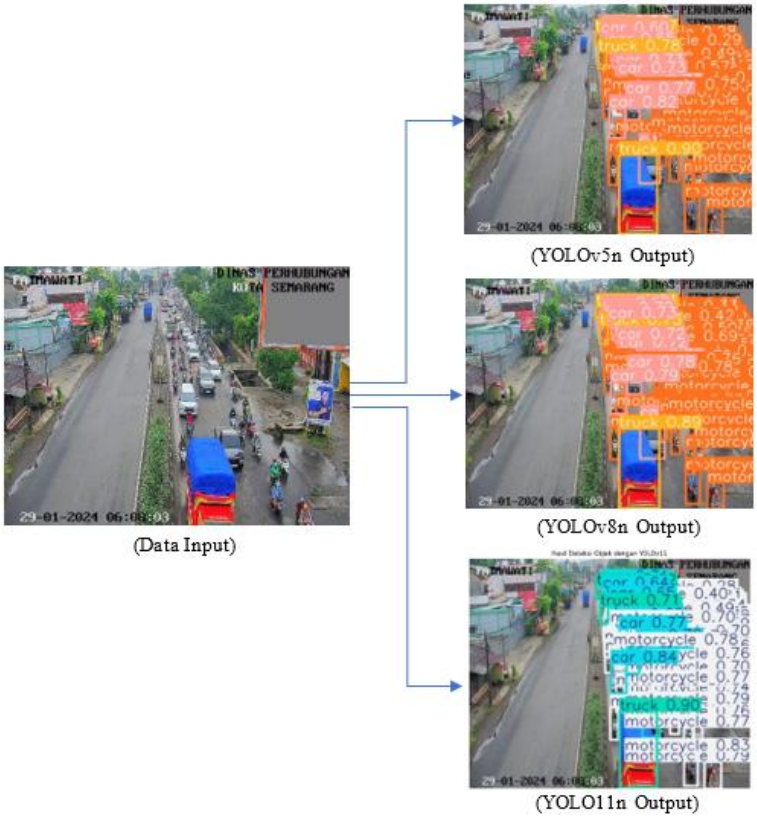

Fig. 4 Sample Results from the Best-Performing YOLO Models

Table 3 presents a comparison of the number of detected objects for each vehicle class, as predicted by the YOLOv5, YOLOv8, and YOLOv11 models, against the actual ground truth values. The observed classes include bus, car, motorcycle, and truck.

Table 3. a Comparison of the Number of Detected Objects for each Vehicle Class

Class	Actual Data	Detection Result		
		YOLOv5n	YOLOv8n	YOLO11n
Bus	1	0	0	0
Car	15	15	15	13
Motorcycle	65	43	61	55
Truck	2	3	3	3

Based on the data in Table 3, the YOLOv8 model demonstrated the closest detection performance to the actual values, particularly for the car and motorcycle classes. Although all models failed to detect any bus objects, they successfully detected the truck class, either accurately or with slight overestimation. In general, YOLOv11 performed better than YOLOv5, but it still fell short of YOLOv8 in terms of overall detection accuracy relative to the ground truth.

To further assess the real-world generalization capabilities of each model, additional testing was conducted across ten traffic scenarios that simulate varying environmental and operational conditions. These scenarios were designed to reflect realistic variations in traffic density, lighting, and weather, including situations such as heavy congestion with glare, smooth flow under rainfall, and frames containing all vehicle classes under both well-lit and challenging lighting conditions. Each model was evaluated on the same set of test frames for consistency. The results of model performance under these diverse test scenarios are summarized in Table 4.

*Imam Ahmad Ashari

SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 3, July 2025
DOI : https://doi.org/10.33395/sinkron.v9i3.15071
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Table 4. The results of model performance under these diverse test scenarios

No.	Testing Scenario	Result			
		Actual Objects Detected	YOLOv5n Result	YOLOv8n Result	YOLO11n Result
1	Heavy traffic with light reflection (glare daytime)	45 cars, 74 motorcycles, 8 trucks	38 cars, 37 motorcycles, 4 trucks	35 cars, 47 motorcycles, 4 trucks	37 cars, 39 motorcycles, 6 trucks
2	Heavy traffic with proper lighting exposure	25 cars, 121 motorcycles, 3 trucks	40 cars, 102 motorcycles, 2 trucks	42 cars, 107 motorcycles, 2 trucks	41 cars, 113 motorcycles, 2 trucks
3	Smooth traffic with light reflection	4 cars, 20 motorcycles	6 cars, 21 motorcycles	6 cars, 22 motorcycles	7 cars, 22 motorcycles
4	Smooth traffic with proper exposure	9 cars, 22 motorcycles	9 cars, 18 motorcycles	9 cars, 18 motorcycles	9 cars, 20 motorcycles
5	Congested traffic during rainfall	50 cars, 97 motorcycles, 6 trucks	36 cars, 84 motorcycles, 7 trucks	49 cars, 85 motorcycles, 7 trucks	46 cars, 93 motorcycles, 7 trucks
6	Smooth traffic during rainfall	4 cars, 9 motorcycles	5 cars, 9 motorcycles	5 cars, 7 motorcycles	5 cars, 9 motorcycles
7	All classes, heavy traffic with light reflection	20 cars, 59 motorcycles, 6 trucks	19 cars, 45 motorcycles, 4 trucks	19 cars, 39 motorcycles, 6 trucks	20 cars, 46 motorcycles, 8 trucks
8	All classes, heavy traffic with proper exposure	15 cars, 65 motorcycles, 2 trucks	16 cars, 44 motorcycles, 3 trucks	13 cars, 51 motorcycles, 3 trucks	12 cars, 57 motorcycles, 4 trucks
9	All classes, smooth traffic with light reflection	1 bus, 10 cars, 18 motorcycles, 2 trucks	1 bus, 11 cars, 19 motorcycles, 2 trucks	1 bus, 11 cars, 20 motorcycles, 2 trucks	1 bus, 11 cars, 16 motorcycles, 2 trucks
10	All classes, smooth traffic with proper exposure	1 bus, 6 cars, 25 motorcycles, 4 trucks	1 bus, 6 cars, 23 motorcycles, 4 trucks	1 bus, 6 cars, 19 motorcycles, 4 trucks	1 bus, 6 cars, 20 motorcycles, 4 trucks

Testing was conducted on ten traffic scenarios involving variations in traffic density, lighting conditions, weather, and the diversity of vehicle classes. Each scenario was analyzed by comparing the actual object counts (manually observed) with the detection results from three models: YOLOv5n, YOLOv8n, and YOLO11n. In the first scenario, which involved heavy traffic with light reflection (glare), YOLOv8n detected the highest number of motorcycles (47 units), while YOLO11n provided a more balanced estimation across vehicle classes (37 cars, 39 motorcycles, 6 trucks), closely matching the actual count. This suggests that YOLO11n demonstrates robustness under visually challenging glare conditions. In the second scenario, which featured heavy traffic under proper lighting exposure, all models successfully detected a large number of motorcycles. YOLO11n achieved the highest detection with 113 motorcycles, indicating its strong performance in high-visibility and high-density environments. For low-density traffic scenarios with challenging lighting (scenarios 3 and 4), all three models performed relatively similarly. However, YOLO11n consistently produced slightly more accurate detections of motorcycles, although the differences were marginal.

Under rainy conditions, particularly in congested traffic (scenario 5), YOLO11n detected 93 motorcycles and 46 cars. These numbers were higher than those produced by YOLOv5n and closely aligned with the actual count. This result highlights YOLO11n's adaptability to adverse weather and reduced visibility conditions. In complex scenarios involving all vehicle classes (scenarios 7 to 10), both in heavy and smooth traffic, YOLO11n consistently produced detections that more closely matched the ground truth compared to the other models. For example, in scenario 10 (smooth traffic, proper exposure, and all vehicle classes present), YOLO11n correctly detected all expected classes including 1 bus, 6 cars, 20 motorcycles, and 4 trucks. In summary, YOLO11n demonstrated the most stable and accurate performance across different lighting conditions, weather variations, and traffic complexities. While YOLOv5n tended to under-detect certain classes and YOLOv8n frequently over-detected motorcycles, YOLO11n showed higher resilience to lighting interference and better recognition of minority classes such as trucks and buses. These results suggest that YOLO11n is a strong candidate for real-time implementation in complex traffic surveillance systems.

*Imam Ahmad Ashari

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 3, July 2025
DOI : https://doi.org/10.33395/sinkron.v9i3.15071
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Model Performance Evaluation

The performance evaluation was conducted to measure the effectiveness and accuracy of each model in detecting vehicle objects. The metrics used in this study include Precision, Recall, mAP@50, and mAP@50–95. Precision measures how accurate the model is in predicting only true positive detections out of all positive predictions. Recall measures the model's ability to detect all relevant objects present in the dataset. Mean Average Precision (mAP) serves as a comprehensive metric for detection performance. mAP@50 represents the average precision at an Intersection over Union (IoU) threshold of 0.50, while mAP@50–95 averages the precision across ten IoU thresholds ranging from 0.50 to 0.95. In addition to detection accuracy, the models were also evaluated in terms of computational speed. The detailed results of the model performance evaluation are summarized in Table 5.

Table 5. Performance Evaluation Results of YOLO Models

No	Augmentation	Model	Precision	Recall	mAP50	mAP50-95	Runtime
1	Without Augmentation	YOLOv5n	0.823	0.557	0.669	0.370	18,72
		YOLOv8n	0.804	0.572	0.679	0.387	18,27
		YOLO11n	0.801	0.548	0.672	0.377	19,58
2	Blur	YOLOv5n	0.816	0.645	0.738	0.445	26,82
		YOLOv8n	0.875	0.655	0.756	0.467	27.48
		YOLO11n	0.850	0.641	0.740	0.453	34,67
3	Brightness Adjustment	YOLOv5n	0.835	0.662	0.742	0.458	31,11
		YOLOv8n	0.773	0.701	0.750	0.462	30,23
		YOLO11n	0.730	0.678	0.741	0.458	32,32
4	Color Jitter	YOLOv5n	0.845	0.637	0.749	0.452	31,55
		YOLOv8n	0.861	0.648	0.754	0.460	30,54
		YOLO11n	0.860	0.660	0.751	0.459	32,22
5	Noise Injection	YOLOv5n	0.795	0.638	0.708	0.396	31,19
		YOLOv8n	0.785	0.633	0.719	0.410	31,33
		YOLO11n	0.779	0.645	0.703	0.407	32,39
6	Scaling	YOLOv5n	0.839	0.675	0.740	0.449	30,88
		YOLOv8n	0.836	0.668	0.752	0.464	29,42
		YOLO11n	0.834	0.681	0.748	0.460	31,33
7	Zoom In	YOLOv5n	0.782	0.640	0.712	0.415	30,82
		YOLOv8n	0.785	0.64	0.722	0.426	29,31
		YOLO11n	0.785	0.655	0.717	0.424	31,56

From Table 5, model performance was evaluated by testing three lightweight YOLO architectures: YOLOv5n, YOLOv8n, and YOLO11n, both without augmentation and with six augmentation techniques applied. These techniques included Blur, Brightness Adjustment, Color Jitter, Noise Injection, Scaling, and Zoom In. The evaluation metrics consisted of Precision, Recall, mAP@0.5, mAP@0.5:0.95, and runtime measured in seconds. Without any augmentation, YOLOv8n outperformed both YOLOv5n and YOLO11n in terms of mAP@0.5:0.95, achieving a value of 0.387, although its Precision was slightly lower than that of YOLOv5n (0.804 vs 0.823). This suggests that, as a baseline, YOLOv8n has a slightly better generalization ability. However, overall, none of the three models achieved optimal performance without data augmentation, confirming the importance of training data diversity in enhancing detection robustness.

Significant performance improvements were observed after applying data augmentation, especially with the Blur, Brightness Adjustment, and Color Jitter techniques. Among all configurations, the best result was obtained by YOLOv8n with Blur augmentation, reaching a Precision of 0.875, Recall of 0.655, mAP@0.5 of 0.756, and mAP@0.5:0.95 of 0.467. The effectiveness of the Blur technique can be attributed to its ability to simulate motion blur and reduce overfitting to high-frequency noise, making the model more tolerant to real-world image imperfections, such as glare or low-resolution CCTV frames. Brightness Adjustment and Color Jitter also delivered

*Imam Ahmad Ashari

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 3, July 2025
DOI : https://doi.org/10.33395/sinkron.v9i3.15071
e-ISSN : 2541-2019
p-ISSN : 2541-044X

competitive results on YOLOv8n, with mAP@0.5:0.95 values of 0.462 and 0.460, respectively. These techniques likely improved robustness to varying lighting conditions, which are common in outdoor traffic scenes captured under natural light. By exposing the model to a range of brightness and color shifts, the augmentation helped the network generalize better across overexposed and underexposed scenarios.

In contrast, the Noise Injection technique resulted in the lowest detection performance across all models. This was particularly evident in YOLOv5n, where mAP@0.5:0.95 dropped to 0.396, and even YOLOv8n only reached 0.410. The added pixel-level noise may have interfered with the model's ability to extract meaningful spatial features, especially in small and partially occluded vehicles, thus degrading accuracy. Scaling and Zoom In showed moderate improvements. Scaling introduced size variability that helped models recognize vehicles at different distances, but excessive scaling can also distort spatial proportions. Similarly, Zoom In augmentation forced the model to learn localized features, which can be useful for detecting small objects, though it might reduce context awareness in wider scenes.

In terms of computational time, YOLOv8n exhibited relatively fast and efficient runtime performance. Its runtime ranged from 18.27 seconds (without augmentation) to 30.54 seconds (with Color Jitter). While YOLO11n achieved competitive performance under some augmentation conditions, its computational time was generally higher, peaking at 34.67 seconds when using the Blur technique. This could be due to its increased model complexity and parameter count compared to YOLOv5n and YOLOv8n. Overall, this evaluation confirms that applying appropriate data augmentation significantly improves object detection model performance. The YOLOv8n model consistently excelled in balancing accuracy and computational efficiency. Among the augmentation techniques, Blur proved to be the most effective in enhancing detection robustness and accuracy for vehicle detection in this context. A visual comparison of mAP@0.5:0.95 across all models and augmentation configurations is presented in Figure 5.

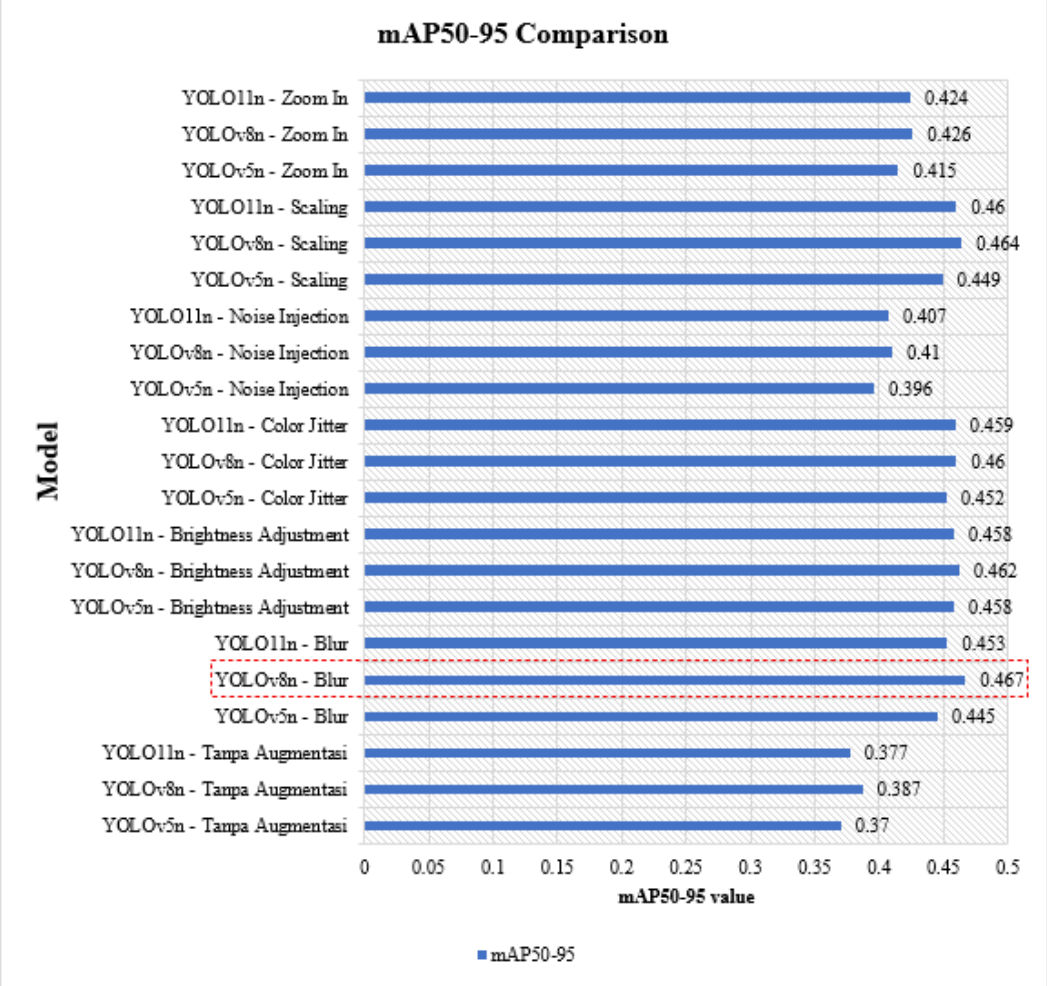

Fig. 5 A Visual Comparison Models of mAP@0.5:0.95

*Imam Ahmad Ashari

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 3, July 2025
DOI : https://doi.org/10.33395/sinkron.v9i3.15071
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Figure 5 presents a visual comparison of mAP@0.5:0.95 scores across all tested model and augmentation configurations. The highest performance was achieved by the YOLOv8n model with Blur augmentation, reaching a mAP@0.5:0.95 value of 0.467, followed by YOLOv8n with Brightness Adjustment (0.462) and Color Jitter (0.460). In contrast, the Noise Injection technique yielded the lowest performance, particularly for YOLOv5n, which scored only 0.396. Models without any augmentation consistently performed worse than those with augmentation applied. This figure clearly highlights that the combination of model architecture and augmentation technique has a significant impact on generalization performance, especially for detecting vehicles in complex and congested traffic conditions.

To analyze detection performance by individual object classes, evaluation metrics were computed separately for each vehicle category: bus, car, motorcycle, and truck. Table 6 presents the values of Precision, Recall, mAP@0.5, and mAP@0.5:0.95 for the YOLOv8n model using the Blur augmentation technique, which was previously identified as the best-performing configuration in this study.

Table 6. Evaluation Results of the Best Model per Class

Class	Images	Precision (P)	Recall (R)	mAP50	mAP50-95
all	144	0.875	0.655	0.756	0.467
bus	44	0.948	0.676	0.778	0.549
car	141	0.872	0.738	0.823	0.485
motorcycle	143	0.820	0.517	0.639	0.301
truck	99	0.861	0.691	0.783	0.533

The results in Table 6 indicate that the bus class achieved the highest mAP@0.5:0.95 score (0.549), followed by truck (0.533), car (0.485), and motorcycle (0.301). Although the car class had the largest number of samples and the highest Recall (0.738), its lower mAP score may be attributed to higher visual complexity and similarity in shape to other objects. In contrast, the relatively low mAP for the motorcycle class suggests that small objects with fine-grained details remain a challenge for detection. These findings support the conclusion that object size and visual clarity significantly influence detection performance in YOLO-based models.

To complement the evaluation of the model's vehicle detection performance, a confusion matrix was used to further examine classification accuracy by class and the occurrence of misclassifications across classes. Figure 6 displays the confusion matrix derived from the YOLOv8n model using the Blur augmentation technique on the test dataset.

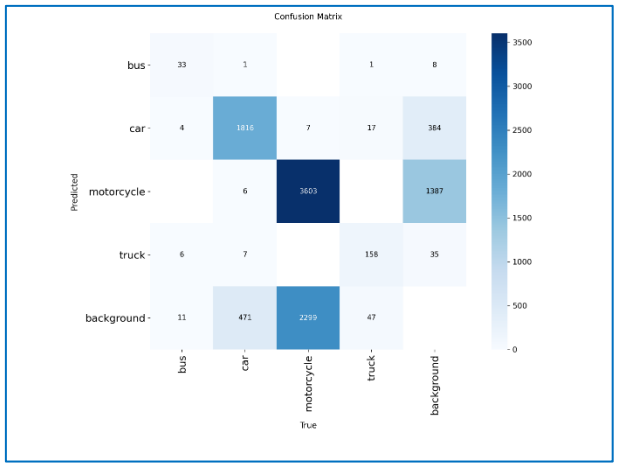

Fig. 6 Confusion Matrix

According to Figure 6, the model successfully recognized most motorcycle objects, with 3,603 correct predictions, although a significant number of false positives were classified as background (2,299 cases). For the car class, the model produced 1,816 correct predictions, but misclassified 471 instances as background and 7 as motorcycles. The bus and truck classes showed relatively low true positives, with 33 and 158 detections respectively. However, their number of misclassifications was lower and more consistent. These results indicate that while overall performance was acceptable, the model still struggled with distinguishing between visually similar vehicle types and separating objects from the background. This was especially apparent for classes with small size or less distinctive features.

*Imam Ahmad Ashari

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 3, July 2025
DOI : https://doi.org/10.33395/sinkron.v9i3.15071
e-ISSN : 2541-2019
p-ISSN : 2541-044X

To monitor the model's training progress, evaluation metrics and loss functions were tracked over the course of training using both the training and validation datasets. Figure 7 presents the loss and performance metric curves for the YOLOv8n model with Blur augmentation over 60 epochs.

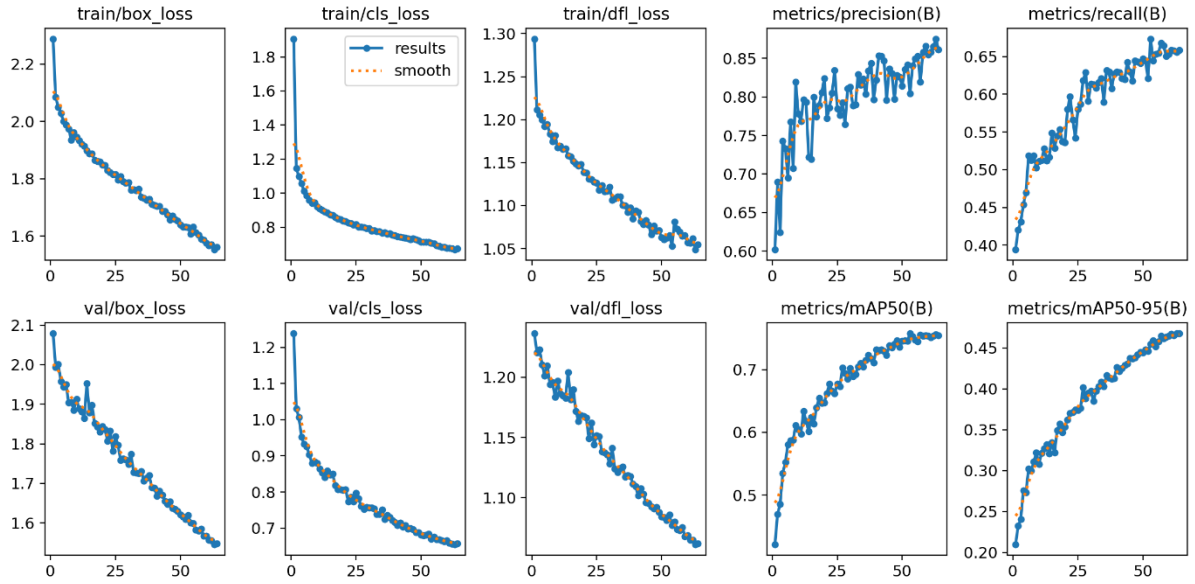

Fig. 7 Loss and Performance Metric Curves

As shown in Figure 7, all loss components, including box loss, classification loss, and distribution focal loss, showed a consistent decrease on both the training and validation sets as the number of epochs increased. This downward trend indicates stable learning and suggests that the model did not exhibit significant overfitting. Meanwhile, performance metrics such as Precision, Recall, mAP@0.5, and mAP@0.5:0.95 demonstrated steady improvement throughout the training process.

The curves for mAP@0.5 and mAP@0.5:0.95 increased sharply during the initial epochs and gradually leveled off toward the end, indicating that the model achieved good convergence. These trends support previous evaluation results and reaffirm that YOLOv8n with Blur augmentation is the most effective configuration identified in this research.

DISCUSSIONS

This study conducted a comparative analysis of three lightweight versions of the YOLO architecture developed by Ultralytics, namely YOLOv5n, YOLOv8n, and YOLO11n, in the context of multi-class vehicle detection under real-world traffic conditions. The evaluation focused on four common vehicle types: motorcycle, car, bus, and truck. These categories were selected because they represent different object sizes and shapes, which reflect the typical challenges found in Indonesian urban traffic environments, such as occlusion, motion blur, and lighting variability (Lin & Zhu, 2024).

To improve model generalization, several augmentation techniques were implemented during training. These included Blur, Brightness Adjustment, Color Jitter, Noise Injection, Scaling, and Zoom In. Among them, the Blur technique produced the best evaluation results during training, particularly for YOLOv8n, which achieved a precision of 0.875, recall of 0.655, mAP@0.5 of 0.756, and mAP@0.5:0.95 of 0.467. This finding is consistent with other, who reported that augmentation methods such as Flip Mosaic enhanced detection of small or blurred objects (Zhang et al., 2022). The good performance of YOLOv8n during training also aligns with prior evaluations of its architecture, which highlight improved detection accuracy across varying object sizes while maintaining computational efficiency (Hong et al., 2024).

However, when tested in scenario-based evaluations involving glare, rain, and low visibility, YOLO11n consistently outperformed YOLOv8n across multiple real-world conditions. This discrepancy between training and testing results may be due to the stronger generalization ability of YOLO11n in noisy or unpredictable environments. Although YOLOv8n showed strong performance on validation datasets, YOLO11n demonstrated better robustness in complex scenes such as rainy weather and harsh lighting, where augmentation alone may not be sufficient. These findings are in line with previous research showing that YOLOv11 performs well in occluded and dense traffic scenarios, with faster inference and improved detection quality (Hong et al., 2024).

*Imam Ahmad Ashari

From a practical perspective, these results support the deployment of YOLO11n for real-time traffic monitoring on edge devices. All three models tested in this study use the "nano" configuration, which is designed to operate efficiently in resource-constrained environments such as embedded systems or mobile platforms. YOLO11n offers a good balance between detection accuracy and computational speed, making it highly suitable for intelligent transportation systems and real-time surveillance applications (Rana et al., 2024) (Wang et al., 2024).

Despite the promising outcomes, this study has several limitations. First, the dataset was collected from a single urban area in Semarang, Indonesia, which may affect the generalizability of the findings to different geographic regions. Second, the study was limited to nano variants of YOLO models and did not include larger or more complex versions such as YOLOv8s or YOLOv11s. Third, although several augmentation techniques were applied, their combined effects with naturally occurring disturbances in real traffic images still require further investigation to understand their impact on detection performance (Rahman et al., 2022).

In conclusion, this study confirms the practical utility of lightweight YOLO architectures for vehicle detection under dense traffic conditions. YOLOv8n achieved the best results during training with Blur augmentation, but YOLO11n consistently outperformed other models during real-world testing scenarios. These results contribute to the growing body of evidence supporting the need to test detection models not only on validation datasets but also in realistic environments that reflect deployment conditions. The insights from this study are valuable for researchers and practitioners seeking to implement efficient and reliable object detection systems in smart city applications.

CONCLUSION

This study concludes that among the three lightweight YOLO architectures evaluated, namely YOLOv5n, YOLOv8n, and YOLO11n, the YOLOv8n model combined with the Blur augmentation technique achieved the best overall performance in detecting vehicle classes such as motorcycle, car, bus, and truck. It produced the highest evaluation scores, with a precision of 0.875, recall of 0.655, mAP@0.5 of 0.756, and mAP@0.5:0.95 of 0.467. Class-wise analysis showed that bus was the easiest object to detect, followed by truck, car, and motorcycle, which remained the most challenging due to its small size, frequent occlusion, and visual similarity to surrounding objects. These results demonstrate that selecting an appropriate model architecture and applying relevant augmentation techniques can significantly enhance detection accuracy, particularly under difficult imaging conditions.

However, the generalizability of these findings is limited by the use of a dataset collected from a single urban location in Semarang, Indonesia, and by focusing exclusively on the nano variants of each model. Broader traffic conditions, weather variations, and model configurations were not included in this study. For future research, it is recommended to explore larger and more complex model variants, integrate attention-based mechanisms, and expand the dataset to cover a wider range of environments. These steps are essential to develop more robust and scalable vehicle detection systems that can be deployed effectively in real-world intelligent transportation scenarios.

ACKNOWLEDGMENT

The authors would like to thank the Internal Grant of Universitas Harapan Bangsa in 2025 for providing financial support, which enabled the successful completion of this research.

REFERENCES

Almukhalfi, H., Noor, A., & Noor, T. H. (2024). Traffic management approaches using machine learning and deep learning techniques: A survey. *Engineering Applications of Artificial Intelligence*, *133*(PB), 108147. https://doi.org/10.1016/j.engappai.2024.108147

Sakhuja, A. (2023). Intelligent Traffic Management System using Computer Vision and Machine Learning. *Innovative Research Thoughts*, *9*(5), 1–10. https://doi.org/10.36676/irt.2023-v9i5-001

Awaluddin, B. A., Chao, C. T., & Chiou, J. S. (2023). Investigating Effective Geometric Transformation for Image Augmentation to Improve Static Hand Gestures with a Pre-Trained Convolutional Neural Network. *Mathematics*, 11(23), 4783.

Elbasha, A. M., & M. Abdellatif, M. (2025). *AIoT-Based Smart Traffic Management System*. 69–77. https://doi.org/10.5121/csit.2024.150204

Eldeen, N., Mohamed, K., & Seyedali, L. (2022). A comprehensive survey of recent trends in deep learning for digital images augmentation. *Artificial Intelligence Review*, *55*(3), 2351–2377. https://doi.org/10.1007/s10462-021-10066-4

Fedorov, A., Nikolskaia, K., Ivanov, S., Shepelev, V., & Minbaleev, A. (2019). Traffic flow estimation with data from a video surveillance camera. *Journal of Big Data*, *6*(1). https://doi.org/10.1186/s40537-019-0234-z

Firdaus, M., & Arief, M. R. (2023). Impact of Data Augmentation Techniques on the Implementation of a

*Imam Ahmad Ashari

Combination Model of Convolutional Neural Network (CNN) and Multilayer Perceptron (MLP) for the Detection of Diseases in Rice Plants. *Journal of Scientific Research, Education, and Technology (JSRET)*, 2(2), 453-465.

Guo, Y., Zhang, Q., Lai, K. K., Zhang, Y., Wang, S., & Zhang, W. (2020). The impact of urban transportation infrastructure on air quality. *Sustainability (Switzerland)*, *12*(14), 1–23. https://doi.org/10.3390/su12145626

Gupta, M., Miglani, H., Deo, P., & Barhatte, A. (2023). Real-time traffic control and monitoring. *E-Prime - Advances in Electrical Engineering, Electronics and Energy*, *5*(March), 100211. https://doi.org/10.1016/j.prime.2023.100211

Hong, X., Huang, J., Zhao, W., Zou, H., Lin, Z., & Chen, Y. (2024). *Object detection for traffic management based on YOLO*. *13018*(Stce 2023), 97. https://doi.org/10.1117/12.3024069

Karim, A., Raza, M. A., Alharthi, Y. Z., Abbas, G., Othmen, S., Hossain, M. S., Nahar, A., & Mercorelli, P. (2024). Visual Detection of Traffic Incident through Automatic Monitoring of Vehicle Activities. *World Electric Vehicle Journal*, *15*(9), 382. https://doi.org/10.3390/wevj15090382

Khan, A. H., Umer, R. M., Dunnhofer, M., Micheloni, C., & Martinel, N. (2023). LBKENet:Lightweight Blur Kernel Estimation Network for Blind Image Super-Resolution. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *14234 LNCS*, 209–222. https://doi.org/10.1007/978-3-031-43153-1_18

Khanam, R., Asghar, T., & Hussain, M. (2025). Comparative Performance Evaluation of YOLOv5, YOLOv8, and YOLOv11 for Solar Panel Defect Detection. *Solar*, *5*(1), 1–25. https://doi.org/10.3390/solar5010006

Lin, X., & Zhu, Z. (2024). Vehicle detection in complex scenarios based on YOLOv5. *Applied and Computational Engineering*, *52*(1), 157–163. https://doi.org/10.54254/2755-2721/52/20241500

Mandal, V. (2019). Artificial Intelligence Enabled Automatic Traffic Monitoring System. *Duke Law Journal*, *1*(1), 1–13. https://doi.org/10.20944/preprints202009.0725.v1

Pei, M., Liu, N., Zhao, B., & Sun, H. (2023). Self-Supervised Learning for Industrial Image Anomaly Detection by Simulating Anomalous Samples. *International Journal of Computational Intelligence Systems*, *16*(1). https://doi.org/10.1007/s44196-023-00328-0

Rahman, R., Bin Azad, Z., & Bakhtiar Hasan, M. (2022). Densely-Populated Traffic Detection Using YOLOv5 and Non-maximum Suppression Ensembling. *Lecture Notes on Data Engineering and Communications Technologies*, *95*, 567–578. https://doi.org/10.1007/978-981-16-6636-0_43

Rana, M. M., Hossain, M. S., Hossain, M. M., & Haque, M. D. (2024). Improved vehicle detection: unveiling the potential of modified YOLOv5. *Discover Applied Sciences*, *6*(7). https://doi.org/10.1007/s42452-024-06029-3

Reche, C., Tobias, A., & Viana, M. (2022). Vehicular Traffic in Urban Areas: Health Burden and Influence of Sustainable Urban Planning and Mobility. *Atmosphere*, *13*(4), 1–21. https://doi.org/10.3390/atmos13040598

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2016-Decem*, 779–788. https://doi.org/10.1109/CVPR.2016.91

Raimondo, D., Raffone, A., Salucci, P., Raimondo, I., Capobianco, G., Galatolo, F. A., ... & Seracchioli, R. (2024). Detection and classification of hysteroscopic images using deep learning. Cancers, 16(7), 1315.

Tchanchou, C., Jiang, X., Wang, Y., & Zheng, R. (2022). YOLO Based Multi-Objective Vehicle Detection and Tracking. *Research Square*, 0–20. https://doi.org/10.21203/rs.3.rs-2006332/v1

Wang, Y., Xu, S., Wang, P., Li, K., Song, Z., Zheng, Q., Li, Y., & He, Q. (2024). Lightweight Vehicle Detection Based on Improved YOLOv5s. *Sensors*, *24*(4). https://doi.org/10.3390/s24041182

Zhang, Y., Guo, Z., Wu, J., Tian, Y., Tang, H., & Guo, X. (2022). Real-Time Vehicle Detection Based on Improved YOLO v5. *Sustainability (Switzerland)*, *14*(19). https://doi.org/10.3390/su141912274

*Imam Ahmad Ashari