



YOLOv8 Analysis for Vehicle Classification Under Various Image Conditions

Eben Panja^{1*}, Hendry², Christine Dewi³

^{1,2,3}Department of Information Technology, Satya Wacana Christian University, Salatiga, Indonesia

Abstract.

Purpose: The purpose of this research is to detect vehicle types in various image conditions using YOLOv8n, YOLOv8s, and YOLOv8m with augmentation.

Methods: This research utilizes the YOLOv8 method on the DAWN dataset. The method involves using pre-trained Convolutional Neural Networks (CNN) to process the images and output the bounding boxes and classes of the detected objects. Additionally, data augmentation applied to improve the model's ability to recognize vehicles from different directions and viewpoints.

Result: The mAP values for the test results are as follows: Without data augmentation, YOLOv8n achieved approximately 58%, YOLOv8s scored around 68.5%, and YOLOv8m achieved roughly 68.9%. However, after applying horizontal flip data augmentation, YOLOv8n's mAP increased to about 60.9%, YOLOv8s improved to about 62%, and YOLOv8m excelled with a mAP of about 71.2%. Using horizontal flip data augmentation improves the performance of all three YOLOv8 models. The YOLOv8m model achieves the highest mAP value of 71.2%, indicating its high effectiveness in detecting objects after applying horizontal flip augmentation.

Novelty: This research introduces novelty by employing the latest version of YOLO, YOLOv8, and comparing its performance with YOLOv8n, YOLOv8s, and YOLOv8m. The use of data augmentation techniques, such as horizontal flip, to increase data variation is also novel in expanding the dataset and improving the model's ability to recognize objects.

Keywords: CNN, Data augmentation, Dawn, Object detection, YOLOv8

Received November 2023 / **Revised** February 2024 / **Accepted** February 2024

This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).



INTRODUCTION

Object detection is a computer vision technology that can recognize specific objects or objects in an image or video and provide output in the form of the object's location along with a label or description of the type of object detected. Object detection enables a computer to identify objects within an image [1]. Traditional object detection methods such as HOG, HARR, and SIFT have been widely used. While these traditional methods are effective in object detection, they still rely heavily on specific features and need more flexibility in handling data variations in object appearance [2],[3]. In recent years, deep learning-based approaches, such as Convolutional Neural Networks (CNN), have shown superior performance in object detection.

One of the deep learning algorithms or methods in object detection is You Only Look Once (YOLO) [4],[5]. Over the years, the development of YOLO has continued to grow from YOLOv2 to YOLOv7 [6],[7]. There is a new version with a more accurate model, YOLOv8. YOLO is an object detection method that has good performance. YOLO uses a pre-trained Convolutional Neural Network (CNN) to process images in layers and output the bounding boxes and classes of detected objects. CNNs can perform a magnificent object recognition process from an image. CCN processes the input image and extracts increasingly complex features such as angle and texture [6],[5]. This technology is usually used for various applications such as

*Corresponding author.

Email addresses: 972022007@student.uksw.edu (Panja), hendry@uksw.edu (Hendry), christine.dewi@uksw.edu (Dewi)

DOI: [10.15294/sji.v11i1.49038](https://doi.org/10.15294/sji.v11i1.49038)

security surveillance, product quality analysis, facial recognition, and others. YOLO is also utilized for detecting vehicle types using object detection [8].

Vehicle type detection is one of the essential things in the world of transportation and road safety. A sound transportation system is one of the critical role in the urban traffic environment [6]. Detecting vehicle types such as cars, motorbikes, buses, and trucks from captured images refers to fitting the target object across consecutive frames. This can be a challenge if the object is in different conditions, such as variations in viewing angle, vehicle direction, lighting conditions, and variations in vehicle image details. This can result in incorrect feature extraction. Additionally, high computation prolongs the data training process, impacting the performance and speed of object detection. To address these challenges, an algorithm or method is required to accurately and swiftly analyze or detect objects [9].

Many studies have used the You Only Look Once (YOLO) method in object detection. A previous study [10], used YOLOv5 and Flip-Mosaic to improve the network's ability to detect vehicle types. The dataset used covered various scenarios on the motorway, monitoring different parts of the road and different viewpoints, thus providing a highly relevant dataset for vehicle object detection in high-speed scenarios, and the Flip-Mosaic method significantly improved the target recognition rate, which was more accurate. Another study used the YOLOv5 architecture to classify hand symbols in real-time, and the model's performance results for classifying hand symbols achieved 80% accuracy, 95% precision, 84% recall, and 89% F1 score [11]. Further research on developing the YOLOv4 approach for live object detection and tracking on CCTV recordings of vehicle activity on Medan city roads at Kratau Bilal, Glugur, and Merdeka locations. The YOLOv4 method produces a detection accuracy rate of 87.98% based on mean average precision (mAP) [12].

This study uses the newer YOLOv8 to improve the detection accuracy of vehicle types such as cars, motorcycles, buses, and trucks. It evaluates the performance of the YOLOv8 method in detecting objects under different weather conditions. By testing the YOLOv8n, YOLOv8s, and YOLOv8m versions on the DAWN dataset with six object classes, this study highlights the performance differences between the models and the application of data augmentation techniques such as horizontal flip. The horizontal flip data augmentation technique flips the vehicle image horizontally to provide a variety of viewpoints and vehicle directions in the dataset. It helps the YOLO model to learn vehicle detection capabilities from different directions.

METHODS

Figure 1 illustrates the research flow using YOLOv8 with the DAWN dataset. YOLOv8 is one of the most well-known models in object detection and classification. The first step in data processing is to collect and prepare a dataset of vehicle images used in model training. This dataset should be large and representative enough so that the model can learn enough variations of the objects to be detected and classified. Once the image dataset is prepared, the next step is to perform Data Pre-Processing. This process involves several stages, such as resizing and creating augmentation data, such as horizontal flips. In this research, scenario 1 involves using the original dataset. Then, scenario 2 adds data augmentation using several versions of YOLOv8, namely YOLOv8n, YOLOv8s, and YOLOv8m. The next stage is to train the YOLOv8 model using image datasets that have been processed and object detection. After completing the training process, the next step is to classify the detected objects. The results of the YOLOv8 model in detection and classification are analyzed by examining precision, recall, and Mean Average Precision (mAP).

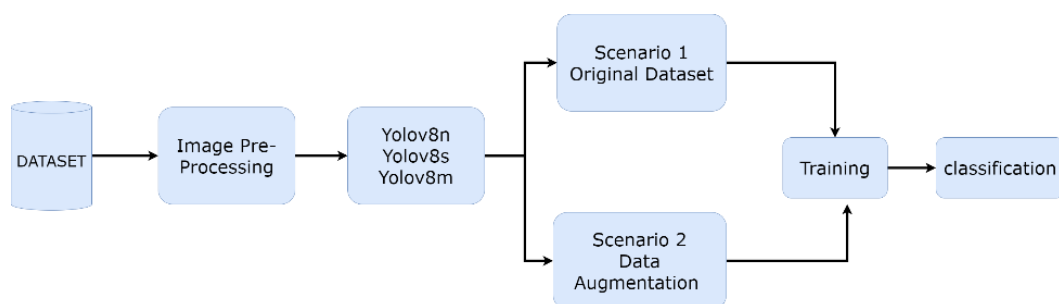


Figure 1. Research Flow

Dataset

The data collection for vehicle detection is taken from the DAWN dataset. The dataset used is open-sourced and widely accessible. The DAWN dataset was developed in 2020, with a total of 1000 images from a collection of real traffic environments [13]. The main objective of the DAWN dataset is to assess the performance of vehicle detection and classification techniques. It includes a range of natural images depicting traffic scenes in various adverse weather conditions, categorized into four groups: fog, snow, rain, and sand. The DAWN dataset shows significant diversity in vehicle type, size, direction, pose, lighting conditions, position, and occlusion. In addition, this dataset greatly emphasizes traffic scenarios that occur in challenging weather, such as heavy snowfall, hail, rain, and extreme weather, such as sand and dust storms [13]. The DAWN dataset has 4 categories of images in various weather conditions in Figure 2, and there are six classes: person, bicycle, car, motorbike, bus, and truck. The weather category statistics of the DAWN dataset can be seen in Table 1.



Figure 2. Example image of DAWN dataset with different conditions (a) sand, (b) fog, (c) snow, and (d) rain [13].

Table 1. DAWN dataset categories

Weather	Images
Sand	316
Fog	279
Snow	203
Rain	196

Image Pre-Processing

Data Augmentation: Data augmentation is a strategy often used to increase dataset diversity and variety and improve model performance [14]. In this study, data augmentation was performed on the dataset to test the difference in results on the original dataset and when adding data augmentation.

In Scenario 1, testing was performed on the original DAWN dataset without adding data augmentation. In Scenario 2, horizontal flip data augmentation was added. Horizontal flip is used to create various viewpoints and directions in the dataset, helping the YOLOv8 model to detect objects from different perspectives. Horizontal flip is applied to the training dataset images as a data augmentation step. The number of images according to the scenario can be seen in Table 2.

Table 2. Number of images per scenario

Scenario	Train	Valid	Test
Scenario 1	688	197	98
Scenario 2	1376	197	98

In the first scenario, without augmentation, 70% of the total data (983 images) used for training, which is equivalent to 688 images. The amount of data to be used for validation is 20%, 197 images, and the amount of data to be used for testing is 10%, 98 images. In Scenario 2, the amount of data used for training remains at 70% of the total data, which is 688 images. However, each original image augmented by flipping, effectively doubling the number of training images to 1376. Therefore, the total data to be used for training is 1376 images. The number of images for validation and testing remains the same as in scenario 1. Figures 3 and 4 show the images from Scenarios 1 and 2, respectively.

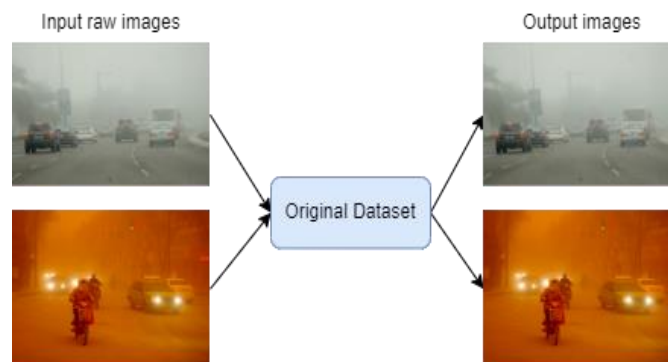


Figure 3. Sample data original dataset scenario 1

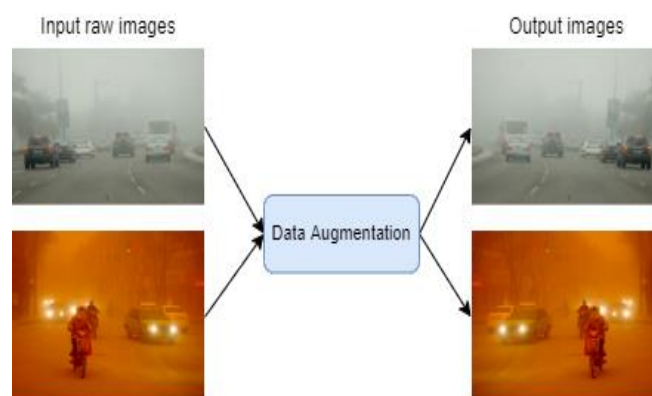


Figure 4. Sample data augmentation scenario 2

Horizontal flip augmentation involves randomly flipping the input image along the vertical axis (from left to right) with a certain probability [15]. The formula for performing a horizontal flip is described in Equation (1) and Equation (2).

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{1}$$

$$x' = -x, y' = y \quad (2)$$

YOLOv8

Object detection algorithms can be categorized into two primary types: classification-based, also known as 2-stage detectors, and regression-based, referred to as 1-stage detectors. Classification-based algorithms follow a two-stage process. In the first stage, they identify a set of regions that are deemed likely to contain objects. In the second stage, Convolution Neural Networks (CNN) are applied to search for objects within the previously marked regions [16],[17]. In contrast, single-stage detectors or regression-based algorithms such as You Only Look Once (YOLO) detect objects in a single detection. YOLO architecture does not select a specific region as the area of interest. Instead, it predicts the class with bounding boxes, which makes detection faster. Therefore, YOLO has become famous for fast and accurate object detection [5],[18].

YOLO continues to evolve, and YOLOv8 was released by the Ultralytics team, which is the latest version of YOLO in January 2023. YOLOv8 introduced five versions, namely YOLOv8n (nano), YOLOv8s (small), YOLOv8m (medium), YOLOv8l (large), and YOLOv8x (extra-large). These models support various computer vision tasks, including object detection, segmentation, pose estimation, tracking, and classification. YOLOv8 can be executed via the command line interface (CLI) or installed as a Python package via pip. In addition, YOLOv8 offers seamless integration options for labelling, training, and deployment. Notably, the developers have introduced several enhancements to improve model accuracy by intensifying data addition during training. One notable feature of YOLOv8 is its ability to incorporate additional images during online training, contributing to its strong performance [19]. YOLOv8 is more efficient than previous versions because it uses a larger feature mAP.

RESULTS AND DISCUSSIONS

Training Results

The DAWN dataset training process uses Google Collaboratory with T4 GPU, System RAM 12.7GB, GPU RAM 15 GB, and 26.8/166.8 GB of storage. In the training process, the 983 images from the DAWN dataset are divided into 70% for training and 30% for validation data. The training was performed using YOLOv8n (nano), YOLOv8s (small), and YOLOv8m (medium) with 100 epochs. This study uses the mean Average Precision (mAP) to represent the average precision across all classes. 'N' denotes the total number of discovered object classes. P denotes the precision value, and R denotes the recall value, and mAP is calculated using Equation (3) as follows [20].

$$mAP = \frac{1}{N} \sum_{i=1}^N \int_0^1 P dR \quad (3)$$

Batch 0 and 1 consist of several data samples randomly selected from the DAWN dataset. The model will process the data in sets and calculate the gradient used to optimize the model parameters, and this is done through a process of "forward pass" and "backward pass," where the model predicts the output and then the gradient indicating how much error will be calculated. Then comes the parameter update stage, where the model will be updated using the computed angle. The main goal is to reduce the model error so that the model can better predict the given data. After completing batches 0 and 1, the model will be tested using validation data to check how much the model has improved during one training iteration. The evaluation results may include metrics such as mAP, confusion matrix, or other appropriate metrics. At this stage, we will explain the training results for batch 0 in Figure 5 and batch 1 in Figure 6.

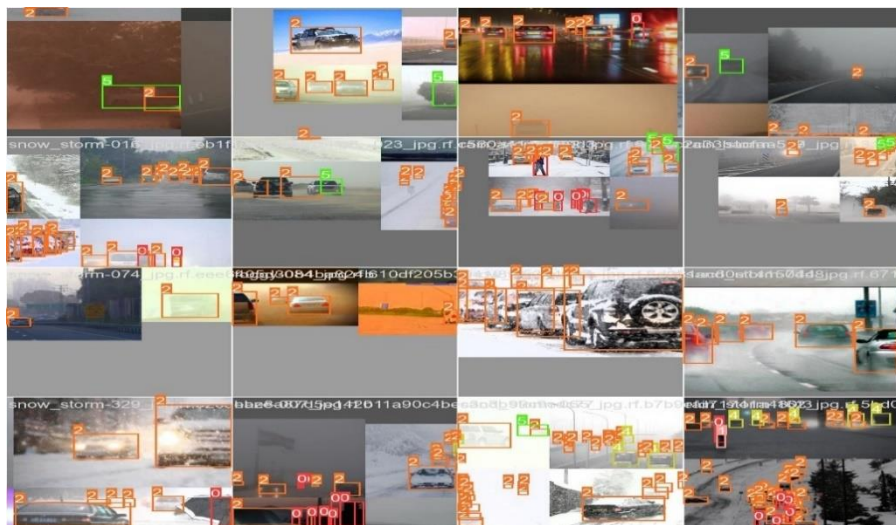


Figure 5. Training process batch 0 labels with DAWN datasets

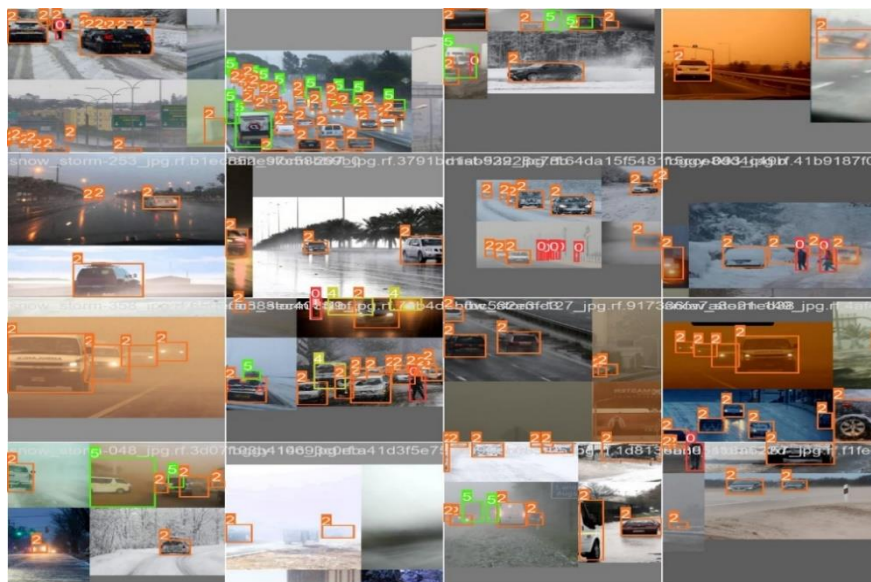


Figure 6. Training process batch 1 labels with DAWN datasets

Table 3. DAWN training results original dataset

Class	YOLOv8n			YOLOv8s			YOLOv8m		
	Precesion	Recall	mAP	Precesion	Recall	mAP	Precesion	Recall	mAP
All	0.735	0.519	0.585	0.744	0.614	0.678	0.844	0.602	0.685
Person	0.808	0.491	0.569	0.76	0.532	0.617	0.798	0.506	0.612
Bicycle	0.992	0.333	0.529	1	0.628	0.732	0.972	0.667	0.834
Car	0.883	0.737	0.819	0.850	0.770	0.816	0.907	0.731	0.834
Motorcycle	0.585	0.545	0.591	0.569	0.638	0.732	0.7985	0.727	0.740
Bus	0.49	0.455	0.412	0.581	0.591	0.594	0.680	0.387	0.508
Truck	0.655	0.556	0.592	0.703	0.526	0.575	0.719	0.594	0.583

Table 3 shows the results of the mean Average Precision (mAP) metric, which measures the performance of the YOLOv8 model's object detector on the validation dataset. A high mAP value indicates better performance in detecting objects. YOLOv8n achieved a mAP value of 58.5%, while YOLOv8s performed better, with a mAP value of approximately 67.8%. The YOLOv8m model has a mAP rate of about 68.5%,

which shows a better mAP rate than YOLOv8n and is almost comparable to YOLOv8s. This shows that YOLOv8m has an excellent ability to detect objects in the validation dataset. Based on the training results, it can be concluded that the YOLOv8m model has the best value with a precision of 84.4%, recall of 60.2%, and mAP value of 68.5%. Furthermore, the bicycle and car classes have the highest mAP value of 83.4%.

Table 4. Horizontal flip data augmentation training result

Class	YOLOv8n			YOLOv8s			YOLOv8m		
	Precesion	Recall	mAP	Precesion	Recall	mAP	Precesion	Recall	mAP
All	0.764	0.524	0.611	0.726	0.584	0.677	0.812	0.635	0.714
Person	0.751	0.448	0.570	0.783	0.574	0.701	0.701	0.596	0.653
Bicycle	0.874	0.500	0.772	1	0.500	0.750	1	0.595	0.877
Car	0.891	0.713	0.809	0.840	0.790	0.842	0.849	0.774	0.840
Motorcycle	0.802	0.545	0.584	0.636	0.636	0.732	0.720	0.727	0.718
Bus	0.541	0.409	0.379	0.429	0.409	0.449	0.913	0.480	0.576
Truck	0.726	0.509	0.555	0.670	0.595	0.589	0.691	0.636	0.622

Table 4 shows the results of the DAWN dataset training process with the addition of horizontal flip data augmentation. The YOLOv8n model, after training with horizontal flip data augmentation, achieved an average mAP level of approximately 61.1%, then YOLOv8s has an improvement in performance with a mAP value of about 67.7% after going through the training process with horizontal flip augmentation. This result shows that such augmentation successfully improves the object detection capability, and the YOLOv8m model significantly improves performance after going through the training process with horizontal flip augmentation, with a mAP value of about 71.4%. This indicates that YOLOv8m has an excellent ability to detect objects after the augmentation. Based on the training results, it can be concluded that the YOLOv8m model with horizontal flip data augmentation has the best value with a precision of 81.2%, recall of 63.5%, and mAP value of 71.4%. In addition to the mAP training results, Precision-Recall curves were obtained from each scenario, namely the original DAWN dataset training scenario in each version of YOLOv8. Figure 7 explains the Precision-Recall curve of the original DAWN dataset and Figure 8 Precision and Recall of Horizontal Flip Data Augmentations.

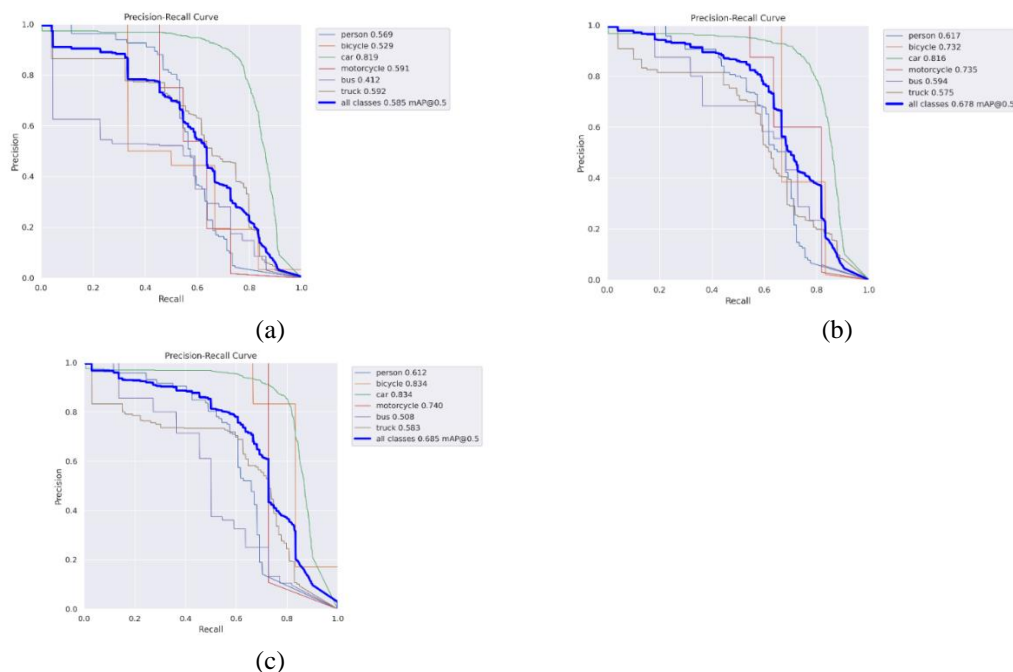


Figure 7. Precision and Recall DAWN original dataset (a) YOLOv8n, (b) YOLOv8s, and (c) YOLOv8m

Figure 7 illustrates the training process followed by the results of the precision and recall curves of the model. The YOLOv8n precision and recall curve has an average mAP value of 58.5%, with the car class having the highest mAP value of 81.9%. Meanwhile, the truck class has the second-highest mAP value at

59.2%. In the YOLOv8s model, the overall mAP value of all classes is 67.8%, with the car class having the highest mAP value of 81.6%, followed by the motorcycle class with the second highest mAP value of 73.5%. In the YOLOv8m model, the overall mAP value is 68.5%, where the car and bicycle classes have the highest mAP value of 83.4%, followed by the motorcycle class with the second highest mAP value of 74%. Using mAP can evaluate the model's overall performance, with the highest mAP usually indicating a better model for detecting objects. The results also identify the object class with the best performance in each model. In all three precision and recall curves, the "car" class consistently has the highest mAP in all three models, indicating that it is better at detecting cars than other object classes.

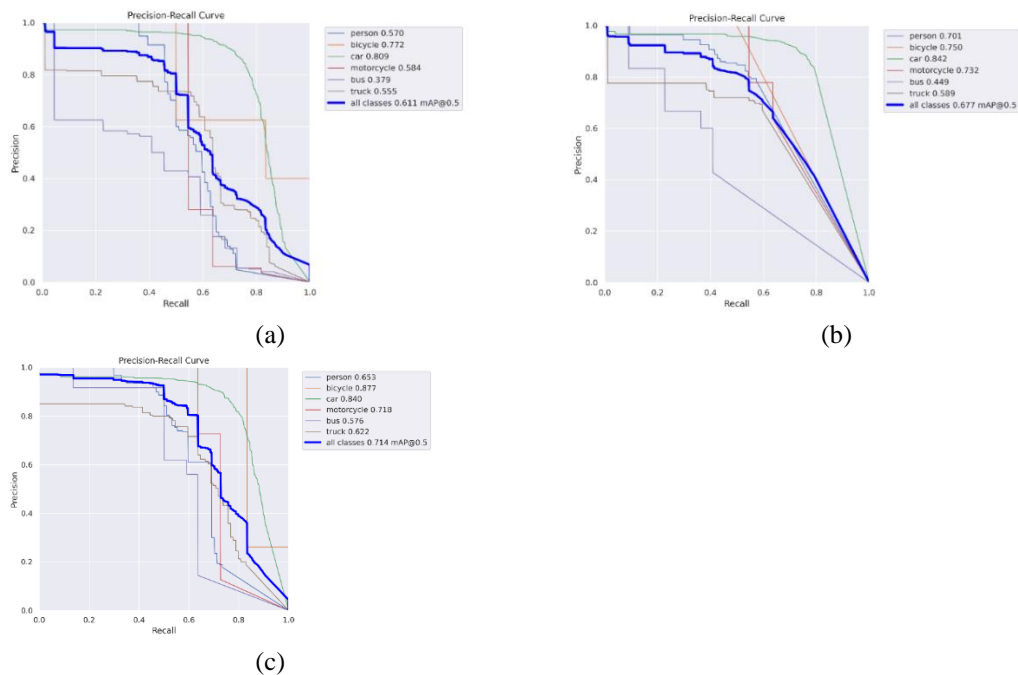


Figure 8. Precision and Recall of horizontal flip data augmentations (a) YOLOv8n, (b) YOLOv8s, and (c) YOLOv8m

Figure 8 illustrates the training process followed by the Precision and Recall curve results of Horizontal Flip Data Augmentation on the model. The YOLOv8n precision and recall curve has an average mAP value of 61.1%, with the car class having the highest mAP value of 0.809%. Meanwhile, the bicycle class has the second-highest mAP value of 77.2%. In the YOLOv8s model, the overall mAP value of all classes is 67.7%, with the car class having the highest mAP value of 0.842%, followed by the bicycle class with the second highest mAP value of 75%. In the YOLOv8m model, the overall mAP value is 71.4%, where the bicycle class has the highest mAP value of 87.7%, followed by the car class with the second highest mAP value of 84%. These results show the performance of the object detection model with horizontal flip data augmentation. The average mAP and the highest-class mAP provide information on how well the model can detect objects in the image. Higher results indicate that the model performs better in the object detection task, and the "car" and "bicycle" classes consistently have the highest mAP among the classes in the model.

The results can be described using four categories: True Positive (TP), where the model predicts a label that matches the ground truth; False Positive (FP), where the model predicts a label that does not correspond to the ground truth; True Negative (TN), where the model correctly does not predict a label; and False Negative (FN), where the model incorrectly predicts a negative evaluation despite the positive situation. Equations (4)-(5) describe precision and recall [21],[22] while equation (6) describes F1 to evaluate the model [23],[24].

$$Precision (P) = \frac{TP}{TP+FP} \tag{4}$$

$$Recall (R) = \frac{TP}{TP+FN} \tag{5}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision+Recall} \tag{6}$$

Equations (7) were applied to compute the YOLO loss function [25].

$$\begin{aligned}
 \text{YOLO Loss Function} = & \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{obj} [(x_i - \hat{x}_i)^2 + (y - \hat{y}_i)^2] \\
 & + \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{obj} \left[(\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2 \right] + \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{obj} (C_i - \hat{C}_i)^2 + \\
 & \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{noobj} (C_i - \hat{C}_i)^2 + \sum_{i=0}^{S^2} \mathbb{1}_i^{obj} \sum_{c \in \text{classes}} (p_i^{\odot} - \hat{p}_i(c))^2
 \end{aligned} \tag{7}$$

S is an abbreviation that refers to the formula used to compute the number of cells in the image. B represents the number of predicted bounding boxes in each grid cell, while c represents the predicted class for each grid cell. Furthermore, the symbol $p_i(c)$ refers to the confidence probability of the class. The center coordinates of the anchor box j in cell i are given by x_{ij} and y_{ij} , while the height and width dimensions of the box are described by h_{ij} and w_{ij} . Moreover, the confidence value for the box is characterized as C_{ij} . Two weights, λ_{coord} and λ_{noobj} , are used to determine the relative significance of localization and recognition during training.

Testing Result

Tables IV and V are the evaluation results of 6 classes on the original DAWN dataset and the addition of horizontal flip data augmentation in the YOLOv8n, YOLOv8, and YOLOv8m versions. The application of horizontal flip data augmentation is used to generate a variety of viewpoints and vehicle orientations in the dataset by rotating the vehicle image horizontally. This helps the YOLOv8 model acquire the ability to detect vehicles from different viewpoints and directions. The YOLOv8m model has the highest mAP of 71.2%.

Table 5. DAWN testing results original dataset

Class	YOLOv8n			YOLOv8s			YOLOv8m		
	Precision	Recall	mAP	Precision	Recall	mAP	Precision	Recall	mAP
All	0.602	0.574	0.580	0.743	0.615	0.685	0.743	0.667	0.689
Person	0.771	0.532	0.569	0.751	0.523	0.620	0.773	0.580	0.616
Bicycle	0.425	0.375	0.488	1	0.636	0.778	0.824	0.788	0.848
Car	0.826	0.780	0.819	0.848	0.766	0.817	0.863	0.793	0.837
Motorcycle	0.480	0.59	0.570	0.568	0.636	0.735	0.717	0.727	0.740
Bus	0.503	0.545	0.454	0.581	0.591	0.535	0.654	0.455	0.508
Truck	0.605	0.620	0.579	0.706	0.535	0.586	0.624	0.657	0.583

Table 5 shows the mAP results of the YOLOv8n model on the original DAWN dataset, with an average mAP rate of about 58%. YOLOv8s shows improved performance over YOLOv8n, with a mAP value of about 68.5%. This shows that the YOLOv8s model better detects objects on the DAWN dataset. The YOLOv8m model achieved the highest mAP value of about 68.9% in the test on the DAWN dataset. This indicates that YOLOv8m performs best in detecting objects on the dataset compared to the other two models. This result shows that YOLOv8m is the best model for detecting objects on the original DAWN dataset, with the highest mAP among the three models. YOLOv8s also shows improved performance compared to YOLOv8n, although not as good as YOLOv8m.

Table 6. Horizontal flip data augmentation testing result

Class	YOLOv8n			YOLOv8s			YOLOv8m		
	Precesion	Recall	mAP	Precesion	Recall	mAP	Precesion	Recall	mAP
All	0.758	0.526	0.609	0.764	0.564	0.620	0.807	0.628	0.712
Person	0.743	0.468	0.569	0.849	0.521	0.608	0.697	0.585	0.657
Bicycle	0.862	0.500	0.743	0.956	0.500	0.625	1	0.597	0.879
Car	0.891	0.721	0.808	0.762	0.762	0.830	0.848	0.778	0.857
Motorcycle	0.799	0.545	0.591	0.636	0.646	0.679	0.719	0.727	0.719
Bus	0.532	0.409	0.380	0.409	0.409	0.416	0.907	0.563	0.563
Truck	0.722	0.515	0.562	0.557	0.557	0.560	0.674	0.615	0.615

In table 6 of the DAWN dataset testing process with Horizontal Flip Data Augmentation, the results of the YOLOv8n model data achieved a MAP value of about 60.9%, YOLOv8s also experienced an improvement in performance with a MAP value of about 62%, YOLOv8m achieved the highest MAP value of about 71.2% after using horizontal flip augmentation. These results show that YOLOv8m is the most effective model in object detection on the DAWN dataset, especially after using horizontal flip augmentation. Such augmentation can improve the model's ability to identify objects in various situations and orientations, which makes it a good choice for more complex object detection tasks. Thus, these results provide important insights into the model performance and data augmentation effects in the context of object detection on the DAWN dataset.

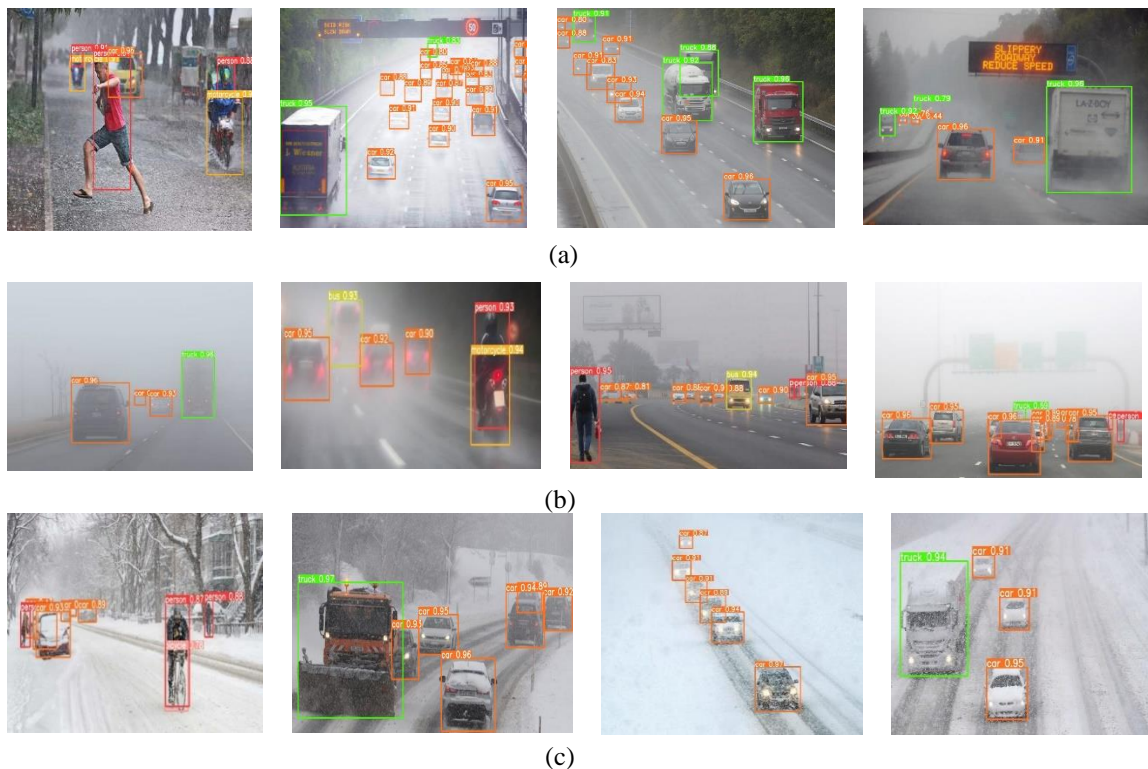
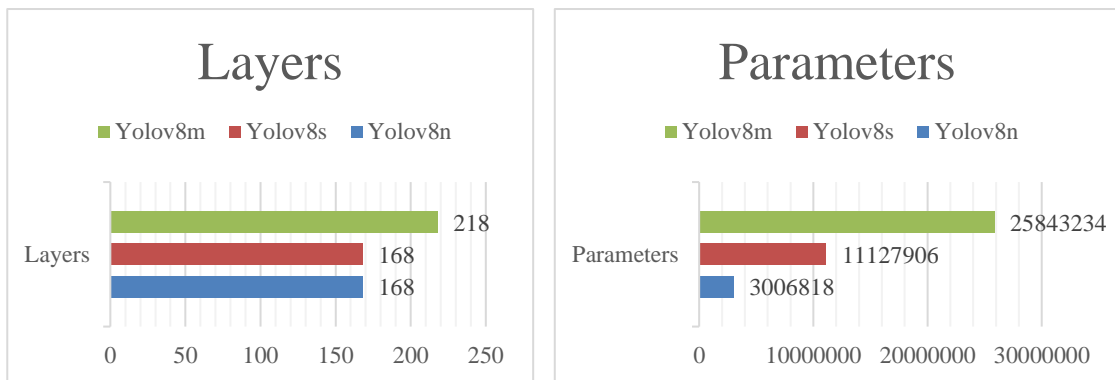


Figure 9. Recognition result of DAWN dataset with YOLOv8 in different scenario (a) fog, (b) rain, (c) sand, and (d) snow

The detection results of the DAWN dataset applied to the YOLOv8 model can be seen in Figures 9(a)-(d). Each object on the image can be detected well, even under different weather conditions, such as fog, rain, sand, and snow, with different accuracies. Under different weather conditions in Figures 9(a)-(d), YOLOv8 can detect objects in the image, such as cars, people, buses, trucks, motorcycles, and bicycles. These results show that the YOLOv8 model has successfully detected objects in various image conditions



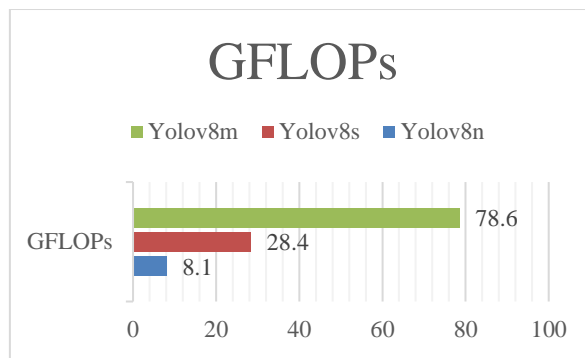


Figure 10. Layers, Parameters, and GFLOPs

Figure 10 contains information on various models' detection performance and complexity, including layers, parameters, and GFLOPs. The YOLOv8n model, with 168 layers, 3006818 parameters, and 8.1 GFLOPs, has the lowest number of layers among the models. This allows the training process to be faster but is limited in handling more complex and diverse data variations. In addition, the number of parameters in this model is 3006818, and the GFLOPs rate of 8.1 is relatively small, which means this model is lighter and requires fewer computational resources. YOLOv8s with 168 layers, 11127906 parameters, and 28.4 GFLOPs, this model is heavier than YOLOv8n. However, the YOLOv8s model has the same 168 layers as YOLOv8n but a higher number of parameters and GFLOPs, and this model may require more computing resources and longer inference time compared to YOLOv8n. The YOLOv8m model has 218 layers, 25843234 parameters, and 78.6 GFLOPs. The YOLOv8m model, with 218 layers and 25843234 parameters, is the largest and has the highest GFLOPs at 78.6, indicating its deeper depth and higher computational requirements compared to the previous models.

Table 7. Comparison of methods on the DAWN datasets

Reference	Method	Mean Average Precision (mAP)
Farid et al. [26]	YOLOv5	0.342
Chaudhuri [27]	Faster R-CNN	0.368
Our Method	YOLOv8m	0.712

In table 7 provides an explanation of the comparison with previous studies. The YOLOv8m technique outperforms previous studies in the mAP evaluation metric, achieving a value of 71.2% on the DAWN dataset. In vehicle classification under various image conditions there is an improvement in the method used. Farid et al. [26] proposed the YOLOv5 method which only showed 34.2% mAP. Another study [27] implemented Faster R-CNN and obtained an mAP value of 36.8%. Comparing the results of these methods, YOLOv8m has a much better performance than previous studies using YOLOv5 and Faster R-CNN. The high mAP value of 71.2% indicates that the model used has a better ability to detect objects in images than the model used in the previous study.

CONCLUSION

This research provides a brief overview of the YOLOv8 method in detecting objects in various weather conditions by testing and analyzing the results obtained. The research process was carried out using the YOLOv8n, YOLOv8s, and YOLOv8m versions of the DAWN dataset with six classes of person, bicycle, car, motorcycle, bus, and truck. The mAP values for the test results are as follows: Without data augmentation, YOLOv8n achieved approximately 58%, YOLOv8s scored around 68.5%, and YOLOv8m achieved roughly 68.9%. However, after applying horizontal flip data augmentation, YOLOv8n's mAP increased to about 60.9%, YOLOv8s improved to about 62%, and YOLOv8m excelled with a mAP of about 71.2%. The application of horizontal flip data augmentation improves the performance of the three YOLOv8 models. The YOLOv8m model achieves the highest mAP value of 71.2%, indicating that it is highly effective in detecting objects after adding horizontal flip augmentation.

REFERENCES

- [1] J. Lee and K. il Hwang, "YOLO with adaptive frame control for real-time object detection applications," *Multimed. Tools Appl.*, vol. 81, no. 25, pp. 36375–36396, 2022, doi: 10.1007/s11042-021-11480-0.
- [2] H. Bahri, D. Krcmarik, and J. Koci, "Accurate Object Detection System on HoloLens Using YOLO Algorithm," *Proc. - 2019 3rd Int. Conf. Control. Artif. Intell. Robot. Optim. ICCAIRO 2019*, pp. 219–224, 2019, doi:

- 10.1109/ICCAIRO47923.2019.00042.
- [3] B. S. Rekha, A. Marium, G. N. Srinivasan, and S. A. Shetty, "Literature Survey on Object Detection using YOLO," *Int. Res. J. Eng. Technol.*, vol. 07, no. 06, pp. 3082–3088, 2020, [Online]. Available: <https://www.irjet.net/archives/V7/i6/IRJET-V7I6576.pdf>
 - [4] D. S. Gothane, "A Practice for Object Detection Using YOLO Algorithm," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, vol. 7, no. 2, pp. 268–272, 2021, doi: 10.32628/cseit217249.
 - [5] N. M. A. A. Dazlee, S. A. Khalil, S. Abdul-Rahman, and S. Mutalib, "Object Detection for Autonomous Vehicles with Sensor-based Technology Using YOLO," *Int. J. Intell. Syst. Appl. Eng.*, vol. 10, no. 1, pp. 129–134, 2022, doi: 10.1039/b000000x.
 - [6] N. Zarei, P. Moallem, and M. Shams, "Fast-Yolo-Rec: Incorporating Yolo-Base Detection and Recurrent-Base Prediction Networks for Fast Vehicle Detection in Consecutive Images," *IEEE Access*, vol. 10, no. November, pp. 120592–120605, 2022, doi: 10.1109/ACCESS.2022.3221942.
 - [7] C. Dewi, A. P. S. Chen, and H. J. Christanto, "Recognizing Similar Musical Instruments with YOLO Models," *Big Data Cogn. Comput.*, vol. 7, no. 2, 2023, doi: 10.3390/bdcc7020094.
 - [8] A. L. Rishika, C. Aishwarya, A. Sahithi, and M. Premchender, "Real-time Vehicle Detection and Tracking using YOLO-based Deep Sort Model: A Computer Vision Application for Traffic Surveillance," *Turkish J. Comput. Math. Educ.*, vol. 14, no. 01, pp. 255–264, 2023.
 - [9] A. Bathija and G. Sharma, "Visual Object Detection and Tracking using YOLO and SORT," *Int. J. Eng. Res. Technol.*, vol. 8, no. 11, pp. 705–708, 2019, [Online]. Available: <https://www.ijert.org>
 - [10] Y. Zhang, Z. Guo, J. Wu, Y. Tian, H. Tang, and X. Guo, "Real-Time Vehicle Detection Based on Improved YOLO v5," *Sustain.*, vol. 14, no. 19, 2022, doi: 10.3390/su141912274.
 - [11] S. Wibowo and I. Sugiarto, "Hand Symbol Classification for Human-Computer Interaction Using the Fifth Version of YOLO Object Detection," *CommIT J.*, vol. 17, no. 1, pp. 43–50, 2023.
 - [12] N. N. Hasibuan, M. Zarlis, and S. Efendi, "Detection and tracking different type of cars with YOLO model combination and deep sort algorithm based on computer vision of traffic controlling," *J. dan Penelit. Tek. Inform.*, vol. 6, no. 1, pp. 210–220, 2021, [Online]. Available: <https://doi.org/10.33395/sinkron.v6i1.11231>
 - [13] M. A. Kenk and M. Hassaballah, "DAWN: Vehicle Detection in Adverse Weather Nature Dataset," pp. 1–6, 2020, doi: 10.17632/766ygrbt8y.3.
 - [14] S. E. Ryu and K. Y. Chung, "Detection model of occluded object based on yolo using hard-example mining and augmentation policy optimization," *Appl. Sci.*, vol. 11, no. 15, 2021, doi: 10.3390/app11157093.
 - [15] R. Hao, K. Namdar, L. Liu, M. A. Haider, and F. Khalvati, "A Comprehensive Study of Data Augmentation Strategies for Prostate Cancer Detection in Diffusion-Weighted MRI Using Convolutional Neural Networks," *J. Digit. Imaging*, vol. 34, no. 4, pp. 862–876, 2021, doi: 10.1007/s10278-021-00478-7.
 - [16] M. H. Putra, Z. M. Yussof, K. C. Lim, and S. I. Salim, "Convolutional neural network for person and car detection using YOLO framework," *J. Telecommun. Electron. Comput. Eng.*, vol. 10, no. 1–7, pp. 67–71, 2018.
 - [17] F. He and D. Tao, "Recent advances in deep learning theory," *Int. J. Mach. Learn. Cybern.*, vol. 11, no. 4, pp. 747–750, 2021, doi: 10.1007/s13042-020-01096-5.
 - [18] I. P. Sary, S. Andromeda, and E. U. Armin, "Performance Comparison of YOLOv5 and YOLOv8 Architectures in Human Detection using Aerial Images," *Ultim. Comput. J. Sist. Komput.*, vol. 15, no. 1, pp. 8–13, 2023, doi: 10.31937/sk.v15i1.3204.
 - [19] S. Tamang, B. Sen, A. Pradhan, K. Sharma, and V. K. Singh, "Enhancing COVID-19 Safety: Exploring YOLOv8 Object Detection for Accurate Face Mask Classification," *Orig. Res. Pap. Int. J. Intell. Syst. Appl. Eng. IJISAE*, vol. 2023, no. 2, pp. 892–897, 2023, [Online]. Available: www.ijisae.org
 - [20] S. Zhao, J. Zheng, S. Sun, and L. Zhang, "SS symmetry An Improved YOLO Algorithm for Fast and Accurate," *Symmetry MDPI*, vol. 14, p. 1669, 2022.
 - [21] Z. Ning, X. Wu, J. Yang, and Y. Yang, "MT-YOLOv5: Mobile terminal table detection model based on YOLOv5," *J. Phys. Conf. Ser.*, vol. 1978, no. 1, 2021, doi: 10.1088/1742-6596/1978/1/012010.
 - [22] J. Hu, X. Zhi, T. Shi, W. Zhang, Y. Cui, and S. Zhao, "Pag-yolo: A portable attention-guided yolo network for small ship detection," *Remote Sens.*, vol. 13, no. 16, 2021, doi: 10.3390/rs13163059.
 - [23] H. Yang *et al.*, "Deep learning for automated detection of cyst and tumors of the jaw in panoramic radiographs," *J. Clin. Med.*, vol. 9, no. 6, pp. 1–14, 2020, doi: 10.3390/jcm9061839.
 - [24] P. K. Sekharamanthy, F. Melgani, and J. Malacarne, "Deep Learning-Based Apple Detection with Attention Module and Improved Loss Function in YOLO," *Remote Sens.*, vol. 15, no. 6, 2023, doi: 10.3390/rs15061516.
 - [25] C. Dewi, A. P. Shun Chen, and H. Juli Christanto, "YOLOv7 for Face Mask Identification Based on Deep Learning," *2023 15th Int. Conf. Comput. Autom. Eng. ICCAE 2023*, no. June, pp. 193–197, 2023, doi: 10.1109/ICCAE56788.2023.10111427.
 - [26] A. Farid, F. Hussain, K. Khan, M. Shahzad, U. Khan, and Z. Mahmood, "A Fast and Accurate Real-Time Vehicle Detection Method Using Deep Learning for Unconstrained Environments," *Appl. Sci.*, vol. 13, no. 5, 2023, doi: 10.3390/app13053059.
 - [27] A. Chaudhuri, "Smart Traffic Management of Vehicles using Faster R-CNN based Deep Learning Method," *arXiv Prepr. arXiv2311.10099*, 2023.