

Vehicle Identification and Classification Using YOLO Algorithm

CHAU LY THI HUYEN, CHI PHAM QUYET, QUYEN THI NGUYEN*

*Faculty of Information Technology, Van Lang School of Technology, Van Lang University,
Ho Chi Minh City, Vietnam*

**Corresponding author: quyen.nt@vlu.edu.vn*

(Received: 12 April 2025; Accepted: 24 December 2025; Published online: 12 January 2026)

ABSTRACT: Vehicle identification and classification are among the challenging activities for the management and control of a large number of different vehicles moving in the inner city. Among many identification and classification systems, the YOLO algorithm stands out for its ability to analyze at high speed and with high accuracy. The algorithm is continually evolving, with notable versions including YOLOv8. This research presents a method for identifying and classifying vehicles using the YOLOv8 algorithm. The assessment of the proposed method's effectiveness was conducted using two COCO datasets (328,000 images) and a real-world dataset from Ho Chi Minh City (HCMC) with more than 1,000 images. The findings indicate that the proposed method can be applied to identify and classify vehicles with an accuracy of 93%-98%. Comparative results with prior studies also demonstrate the superiority of the YOLOv8 algorithm.

ABSTRAK: Pengecaman dan pengelasan kenderaan adalah salah satu aktiviti mencabar dalam pengurusan dan kawalan sejumlah besar kenderaan berbeza yang bergerak di bandar. Di antara kebanyakan sistem pengecaman dan pengelasan, algoritma YOLO menonjol kerana keupayaannya menganalisa pada kelajuan berketepatan tinggi. Algoritma ini terus dibangunkan dengan penambahbaikan yang banyak dan versi yang ketara ialah YOLOv8. Penyelidikan ini membentangkan kaedah mengenal pasti dan mengelaskan kenderaan menggunakan algoritma YOLOv8. Penemuan penilaian keberkesanan kaedah yang dicadangkan telah dijalankan menggunakan dua set data COCO dengan 328,000 imej dan set data sebenar di Ho Chi Minh City (HCMC) lebih daripada 1,000 imej. Dapatan kajian menunjukkan kaedah ini diaplikasikan dalam mengenal pasti dan pengelasan kenderaan berketepatan 93% hingga 98%. Hasil perbandingan dengan kajian lepas juga menunjukkan keunggulan algoritma YOLOv8.

KEYWORDS: YOLOv8, Vehicle identification, Vehicle classification, COCO datasets

1. INTRODUCTION

The number of vehicles in large cities has become a challenge for the management and monitoring of vehicular traffic. Vehicle detection and classification are among the key activities of intelligent traffic observation systems [1]. In addition, as the economy grows, the number of automobiles and the incidence of traffic accidents are increasing, making it crucial to implement measures to reduce these accidents and protect life and property. Vehicle identification and classification are essential to traffic analysis, particularly in developed countries [2]. By the end of 2023, Ho Chi Minh City (HCMC), Vietnam, will have approximately 10 million vehicles. This indicates that the identification and management of vehicles on the streets are urgent requirements to ensure traffic safety. Meanwhile, public transportation studies are the primary resource for analyzing traffic accidents, congestion, and traffic flow, enabling a better understanding of traffic dynamics to reduce fatal accidents that

mainly occur due to driver carelessness or due to poor visibility in severe weather conditions, or improper lighting [3,4].

In recent years, the ongoing enhancement of computing hardware has facilitated the swift advancement of convolutional neural networks, leading to significant successes in the field of computer vision [5]. Currently, the most widely used methods for object detection are R-CNN chains [6,7], SSD chains [8,9], and YOLO chains [10,11]. In the research [12], YOLO, as one of the most representative single-stage object detection frameworks, has been extensively applied to traffic target detection tasks. It employs one of the most effective neural network architectures to achieve high accuracy and overall processing speed. The images are fed into the YOLO algorithm to detect and classify vehicles in each image, which is a major reason for its popularity. Redmon introduced the first iteration of YOLO in their research [10]. In particular, the YOLOv8 version (released in 2018) introduces new and optimized features [13], making this algorithm an excellent option for diverse object detection tasks across a broad spectrum of applications. Researchers have continually proposed various enhanced techniques, including object segmentation [14], silhouette extraction [15], and feature extraction and classification [16], to improve the object detection performance of the system [17,18]. Many fields are applying YOLO for object detection, such as in the creation of the LVD-YOLO model to detect vehicles based on YOLOv5 [19], detecting fruit ripeness in a natural orchard environment with YOLOv7 [20], classification of blood cell image [21], human detection in infrared imagery [22], rice grain detection [23], identification of fault position in power grids [24], safety tracking [25], or applying YOLOv8 to detect wildlife targets for better conservation work [26].

In this study, the YOLOv8 algorithm is employed to detect and classify vehicles in images and videos. The suggested approach is applied to the COCO dataset and to a traffic image dataset in HCMC, Vietnam. Using a large dataset of images and videos, the research produces highly accurate identification and classification results, improving the applicability of the YOLOv8 algorithm for developing a traffic monitoring system and generating traffic statistics, thereby helping to reduce traffic congestion and accidents.

The primary contributions of our research are as follows:

- Successfully applying the YOLOv8 algorithm to identify and classify vehicles.
- The method of identifying and classifying vehicles based on YOLOv8 is successfully applied to two datasets with high accuracy.
- Building a real dataset and successfully applying the proposed method to identify and classify.

The following are the remaining parts of this paper. The problem of identifying and classifying vehicles using the YOLOv8 algorithm, along with the implementation steps, is presented in Sections 2 and 3. The results of applying the proposed method to two large datasets are shown in Section 4, and the conclusion is provided in Section 5.

2. METHOD

2.1. Yolov8 Algorithm

The YOLO architecture operates on the principle of conducting object detection in a single pass through the network, which makes YOLO high-speed and suitable for real-time applications. The input image is segmented into a grid, and bounding boxes along with class probabilities are predicted for each grid cell. The YOLOv8 architecture is built on a

convolutional neural network and comprises three primary components: the backbone, neck, and head, each responsible for distinct processing tasks, as illustrated in Figure 1.

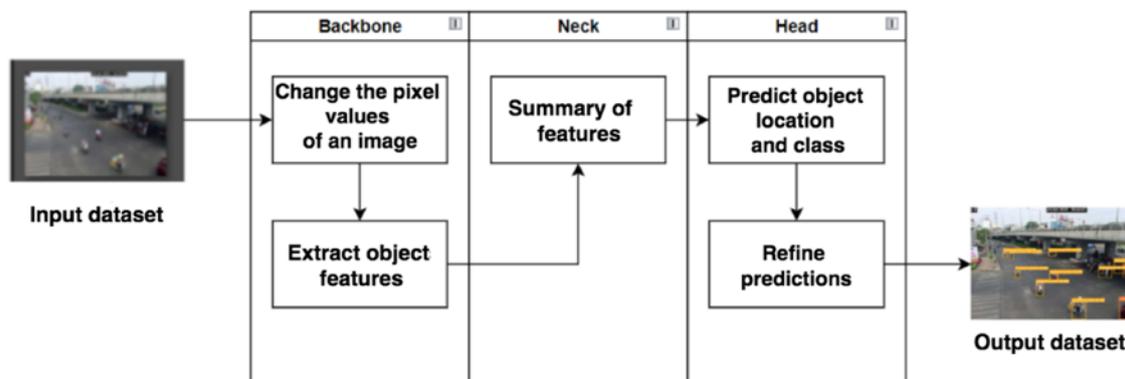


Figure 1. Operation model of the YOLOv8 algorithm in vehicle image recognition

YOLOv8 works as follows:

- **Input Processing:** YOLOv8 takes an input image and processes it through its backbone network to extract object features. This process includes resizing the image to a fixed dimension of 640x640 pixels and normalizing the pixel values.
- **Feature Fusion:** The extracted object features are then processed through the Neck, where they are aggregated to capture information at different scales efficiently.
- **Detection:** The Head uses the processed features to predict bounding boxes, confidence scores, and class probabilities for every detected object. After obtaining the object bounding box and the center point location, YOLO identifies objects throughout the entire image in an end-to-end manner [5].
- **Post-processing:** Finally, techniques like NMS are used to enhance the predictions by removing overlapping boxes and ensuring that each object is detected only once with a high level of confidence.

2.2. Steps to apply the YOLO algorithm in traffic identification and classification

The research will be conducted on the YOLOv8 variant and shown through the flowchart of the vehicle identification and classification algorithm presented in Figure 2.

The identification and classification algorithm based on the YOLOv8 algorithm is summarized in the following 5 main steps:

- *Step 1:* Data collection and processing

First, it is essential to gather a substantial amount of image and video data of vehicles. The objects identified in this research are cars, trucks, buses, motorbikes, and bicycles. If it is an image, label it as a training and validation dataset. If it is a video, split it into multiple frames and label them similarly to the images. Then label these images with the specific location and vehicle to create an accurate training dataset.

- *Step 2:* Model training

Using YOLOv8, detected vehicles are assigned an identifier for identification and classification. The number of vehicles will be determined for each frame. YOLOv8 is trained on a labeled dataset to enable the model to recognize vehicle characteristics in input images.

The training process will enhance the model's ability to accurately and quickly recognize and classify vehicles. The model training process is evaluated using the Precision metric to indicate the model's prediction accuracy. The value of this index is determined by Eq. (1).

$$Precision = \frac{TP}{TP+FP} \times 100\% \quad (1)$$

where TP (True Positive) is the actual number of vehicles, and FP (False Positive) is the number of vehicles overcounted or missed.

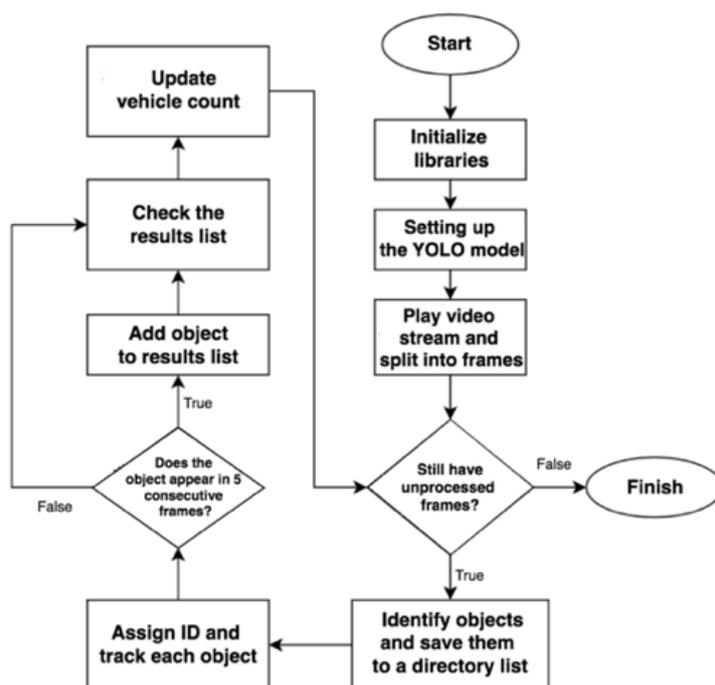


Figure 2. Flowchart of vehicle identification and classification algorithm

- *Step 3: Vehicle recognition phase*

YOLOv8 draws bounding boxes around detected vehicles and uses different colors to distinguish among objects. Display information about the vehicle within the bounding box (e.g., vehicle type and optional attributes, such as direction of movement), then save it to the object list.

- *Step 4: Feature extraction*

YOLOv8 uses the network's feature layers to extract the necessary information from the detected vehicles.

- *Step 5: Vehicle classification and counting*

Finally, vehicles detected by YOLO with the previously assigned identifier will be continuously monitored over five consecutive frames. If the check result is correct, the object will be saved to the item list. Each time a vehicle passes through the detection area, the counter increments the total count of detected vehicles in the item list, and the new count replaces the previous count. The result is displayed to facilitate tracking and understanding of the detected vehicles.

3. RESULTS AND DISCUSSION

3.1. COCO dataset

COCO is an image and caption dataset comprising over 328,000 images, each with a caption describing the objects in the image. COCO is used to train deep learning models for tasks such as object detection, image segmentation, and image captioning. It is trusted by the majority of users, as shown in Figure 3. The dataset is divided into three groups: Train, Validate, and Test, each serving its intended purpose.

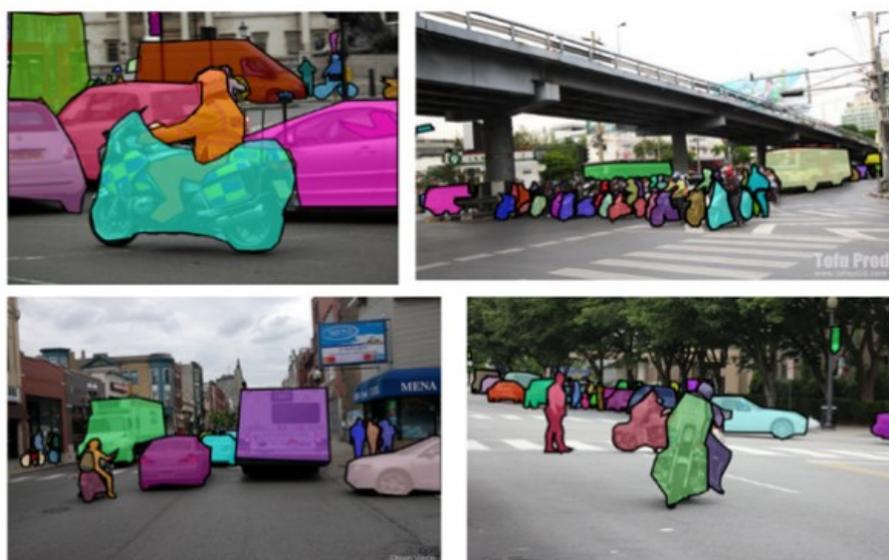


Figure 3. Some labeled images in the COCO dataset

3.2. HCMC dataset

The HCMC dataset includes 05 types of vehicles: cars, motorbikes, buses, trucks, and bicycles, with more than 1000 images and videos collected in the inner city of HCMC to form a dataset for vehicle recognition and classification. These images and videos are presented similarly in Figure 4 and were taken under good lighting conditions.

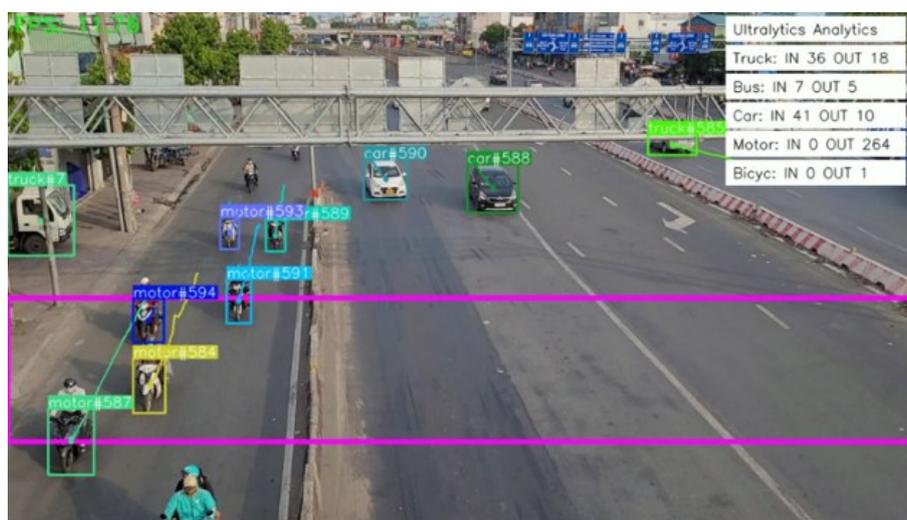


Figure 4. Sample video recording of traffic at Ho Chi Minh City Pedestrian Bridge

3.3. Implementation Results

3.3.1. COCO dataset

The chart illustrating the model's classification performance on the dataset is presented in Figure 5. The curve indicates the relationship between the model's accuracy and confidence in brand recognition. The results show that on the COCO dataset, the model's parameters achieve perfect accuracy at a confidence level of 1.00. This indicates that the model is highly accurate when it is confident about its predictions at a high level. Recognition results for some images under varying lighting conditions are shown in Figures 6 and 7.

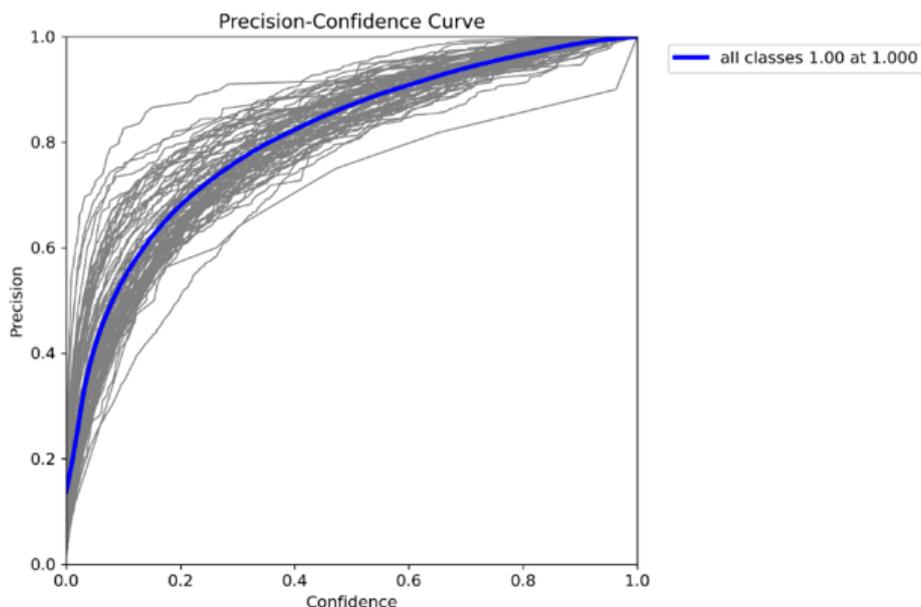


Figure 5. Precision-Recall Curve for the COCO dataset



Figure 6. COCO Some images recognized using the COCO dataset

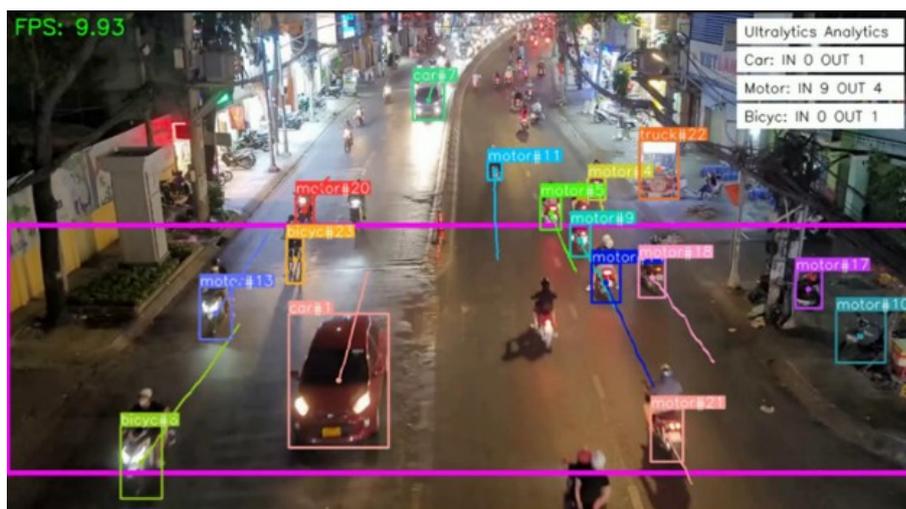


Figure 7. YOLOv8 running on NightStreet.mp4 video

Table 1. Comparison table of Average Precision (AP) values alongside other state-of-the-art techniques

Techniques	AP	AP50	AP75
YOLOv2 [27]	21.6	44.0	19.2
SSD [27]	31.2	50.4	33.3
YOLOv2_Vehicle [27]	39.1	59.1	42.3
R-CNN [27]	33.2	53.3	35.2
Proposed [27]	40.3	60.5	43.5
YOLOv8	98.1	99.8	99.3

```
Running per image evaluation...
Evaluate annotation type *bbox*
DONE (t=126.43s).
Accumulating evaluation results...
DONE (t=28.63s).
Average Precision (AP) @[ IoU=0.50:0.95 | area= all | maxDets=100 ] = 0.981
Average Precision (AP) @[ IoU=0.50 | area= all | maxDets=100 ] = 0.998
Average Precision (AP) @[ IoU=0.75 | area= all | maxDets=100 ] = 0.993
Average Precision (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.963
Average Precision (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.997
Average Precision (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.999
Average Recall (AR) @[ IoU=0.50:0.95 | area= all | maxDets= 1 ] = 0.559
Average Recall (AR) @[ IoU=0.50:0.95 | area= all | maxDets= 10 ] = 0.965
Average Recall (AR) @[ IoU=0.50:0.95 | area= all | maxDets=100 ] = 0.989
Average Recall (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.977
Average Recall (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.998
Average Recall (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.999
```

Figure 8. Overall index obtained when evaluating the dataset

The comparison results between YOLOv8 and other vehicle detection techniques, evaluated on the COCO dataset, are presented in Table 1. These methods have been successfully applied to vehicle detection and trained and evaluated on the COCO dataset. This ensures both relevance to the problem domain and fairness in comparison. The experiment demonstrates that YOLOv8 is the most effective method compared with other vehicle detection techniques in terms of Average Precision (AP) at IOU thresholds of 50 and 75, with substantial differences relative to the proposed method and prior methods. The fact that YOLOv8 is five times larger than its predecessor, YOLOv2, shows how far the YOLO method has advanced,

and the accuracy is more than twice as high as that of the proposed method [27], showing that the YOLO model has surpassed other advanced detection techniques. The YOLOv8 index obtained from the evaluation process is shown in Figure 8.

The YOLOv8 variant can identify vehicle types more effectively than other leading vehicle recognition methods, as indicated in Table 2. The identification methods listed in the table achieve high accuracy, with the lowest recognition rate being 94.2% for SSD-based car recognition, which is considered very high. The advanced R-CNN method and the YOLOv2 variants both achieve recognition rates above 95% and differ by only 2%. The proposed method in [27] is a recognition model based on the YOLOv2 variant, but has been modified to give significantly improved performance when all indicators are superior to the original version. The YOLOv8 model trained on the COCO dataset outperforms other object detection methods in all aspects, with the accuracy of vehicle types almost asymptotically close to 100%. The YOLOv8 index obtained through the evaluation process is shown in Figure 9.

Table 2. Comparison table of experimental outcomes for vehicle classification

Techniques	Car (AP)	Truck (AP)	Bus (AP)	mAP (%)
R-CNN [27]	95.04	97.17	95.02	95.74
YOLOv2 [27]	95.52	95.41	96.34	95.75
YOLOv2_Vehicle [27]	96.32	95.55	97.73	96.56
SSD [27]	94.2	97.75	97.97	96.64
Proposed [27]	96.54	99.29	97.74	97.85
YOLOv8	99.5	99.5	99.5	99.7

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95
all	5000	36335	1	0.999	0.995	0.995
person	5000	10777	1	0.999	0.995	0.995
bicycle	5000	314	1	1	0.995	0.995
car	5000	1918	1	0.998	0.995	0.995
motorcycle	5000	367	1	0.995	0.995	0.995
airplane	5000	143	1	1	0.995	0.995
bus	5000	283	1	1	0.995	0.995
train	5000	190	1	1	0.995	0.995
truck	5000	414	1	1	0.995	0.995
boat	5000	424	1	1	0.995	0.995

Figure 9. Vehicle Rating Index of the COCO Dataset

3.3.2. HCMC dataset

The following graph was obtained during the training of the recognition model using the HCMC dataset. Figure 10 illustrates the correlation between precision and confidence, with the x-axis representing the confidence threshold and the y-axis indicating accuracy. The figure shows that on the HCMC dataset, the model achieves nearly 1.0 as confidence approaches 0.999, indicating high accuracy.

The results indicate that the model accurately detects and predicts most objects when the input data is of good quality, and the directions do not obscure one another, as shown in Table 3. Examples of recognized images in the traffic dataset for specific routes are shown in Figure 11.

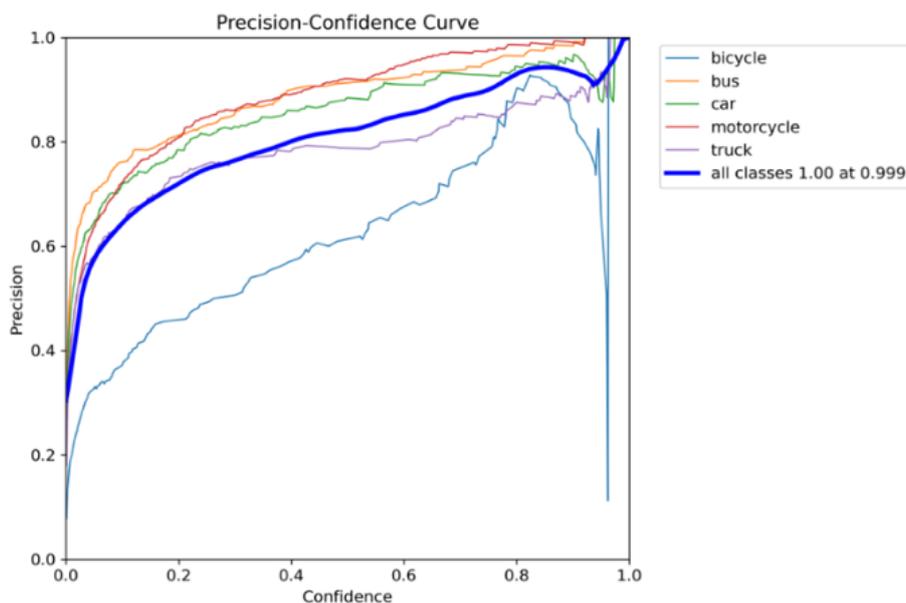


Figure 10. Precision-Confidence Curve for the HCMC dataset

Table 3. Test results on images with low vehicle count

Vehicle	Highway13.	HighwayNam	Dist1Bicycl.	PublicBicycle	Public Bicycle	Exact ratio (%)
	.jpg	.jpg	.jpg	.jpg	Station.jpg	
	Counted/Reality	Counted/Reality	Counted/Reality	Counted/Reality	Counted/Reality	
Car	4/4	9/10	0/0	4/4	3/3	95.23
Motorbike	5/6	0/0	5/5	4/4	0/0	93.33
Truck	1/1	9/9	0/0	0/0	1/1	100
Bus	1/1	7/7	2/2	0/0	1/1	100
Bicycle	0/0	0/0	3/3	5/6	8/8	94.12



Figure 11. Labeled images on some routes

The counts of each vehicle type relative to the total number of vehicles in the test videos are shown in Table 4. The table shows that YOLOv8 performs well with video as an input data type when accurately detecting most of the objects appearing in the video with an accuracy of at least 98.3%, and can reach 100% under ideal conditions. The results of video recognition and classification are presented in Figure 12.

Table 4. YOLOv8 test results on video data

Vehicle	BT.mp4	BicycleRacing.mp4	Parade.mp4	Footbridge1.mp4	Footbridge2.mp4	Exact ratio (%)
	Counted/Reality	Counted/Reality	Counted/Reality	Counted/Reality	Counted/Reality	
Car	35/37	12/12	4/4	47/47	51/51	98.67
Motorbike	6/6	77/77	13/13	263/270	264/264	98.89
Truck	3/1	36/36	12/12	28/28	54/54	98.47
Bus	0/0	7/7	0/0	7/7	12/12	100
Bicycle	0/0	54/54	0/0	2/2	1/2	98.27

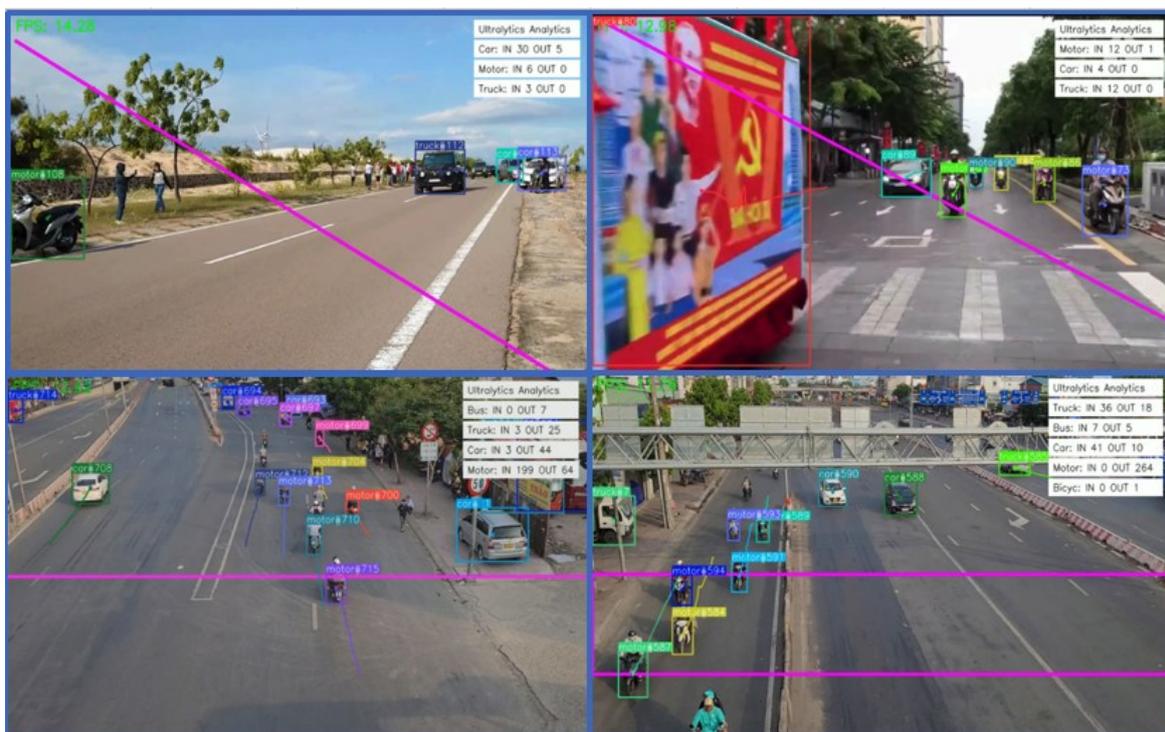


Figure 12. Some Videos Tagged in Footbridge Activity

4. CONCLUSION

In this research, the YOLOv8 algorithm has been successfully applied to identify and classify vehicles in HCMC and the COCO dataset. The model can identify and classify five common types of vehicles: motorbikes, cars, trucks, buses, and bicycles. The findings from the two datasets indicate that the proposed method is effective in identifying and classifying vehicles with high accuracy. Specifically, precision on the two datasets, HCMC and COCO, is {93.33%, 100%} and {98.27%, 100%}. The research has been tested on numerous real videos recorded along inner-city routes in HCMC and has achieved good results. At the same time, in addition, the comparison results with previous studies show that YOLOv8 is the best method compared to vehicle detection techniques when possessing Average Accuracy AP with IOU thresholds of 50 and IOU 75, both of which show a significant difference when compared to the proposed method and earlier methods. In addition, the comparison results indicate that the YOLOv8 variant outperforms other advanced vehicle identification methods in identifying vehicle types, with accuracy approaching 100% across all categories. In short, the research results have contributed to the experiment and demonstrated that the YOLOv8 algorithm is

highly effective in identifying and classifying vehicles; these findings can be widely applied in practice. For future studies, real-world datasets from large cities or areas with high traffic volumes and frequent traffic congestion will continue to be collected to evaluate the effectiveness of YOLOv8.

REFERENCES

- [1] N. Al Mudawi *et al.*, “Vehicle detection and classification via YOLOv8 and deep belief network over aerial image sequences,” *Sustainability*, vol. 15, no. 19, p. 14597, 2023, doi: <https://doi.org/10.3390/su151914597>.
- [2] P. Premaratne, I. Jawad Kadhim, R. Blackledge, and M. Lee, “Comprehensive review on vehicle Detection, classification and counting on highways,” *Neurocomputing*, vol. 556, p. 126627, 2023, doi: <https://doi.org/10.1016/j.neucom.2023.126627>.
- [3] H. Rodríguez-Rangel, L. A. Morales-Rosales, R. Imperial-Rojo, M. A. Roman-Garay, G. E. Peralta-Peñuñuri, and M. Lobato-Báez, “Analysis of Statistical and Artificial Intelligence Algorithms for Real-Time Speed Estimation Based on Vehicle Detection with YOLO,” *Applied Sciences*, vol. 12, no. 6. 2022. doi: 10.3390/app12062907.
- [4] M. Maity, S. Banerjee, and S. S. Chaudhuri, “Faster r-cnn and yolo based vehicle detection: A survey,” in *2021 5th international conference on computing methodologies and communication (ICCMC)*, IEEE, 2021, pp. 1442–1447. doi: 10.1109/ICCMC51019.2021.9418274.
- [5] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 2002, doi: 10.1109/5.726791.
- [6] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation,” in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587. doi: 10.1109/CVPR.2014.81.
- [7] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980–2988. doi: 10.1109/ICCV.2017.322.
- [8] W. Liu *et al.*, “SSD: Single Shot MultiBox Detector BT - Computer Vision – ECCV 2016,” B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., Cham: Springer International Publishing, 2016, pp. 21–37.
- [9] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, “DSSD: Deconvolutional Single Shot Detector,” *ArXiv*, vol. abs/1701.06659, 2017, [Online]. Available: <https://api.semanticscholar.org/CorpusID:7691159>
- [10] J. Redmon, “You only look once: Unified, real-time object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.
- [11] J. Redmon and A. Farhadi, “YOLO9000: Better, Faster, Stronger,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6517–6525. doi: 10.1109/CVPR.2017.690.
- [12] A. Ji and X. Ma, “Vehicle detection and classification for traffic management and autonomous systems using YOLOv10,” *Alexandria Engineering Journal*, vol. 127, no. March, pp. 804–816, 2025, doi: 10.1016/j.aej.2025.06.049.
- [13] M. Safaldin, N. Zaghden, and M. Mejdoub, “An Improved YOLOv8 to Detect Moving Objects,” *IEEE Access*, vol. 12, pp. 59782–59806, 2024, doi: 10.1109/ACCESS.2024.3393835.
- [14] A. Ahmed, A. Jalal, and A. A. Rafique, “Salient Segmentation based Object Detection and Recognition using Hybrid Genetic Transform,” in *2019 International Conference on Applied and Engineering Mathematics (ICAEM)*, 2019, pp. 203–208. doi: 10.1109/ICAEM.2019.8853834.
- [15] A. Farooq, A. Jalal, and S. Kamal, “Dense RGB-D map-based human tracking and activity recognition using skin joints features and self-organizing map,” *KSII Transactions on Internet & Information Systems*, vol. 9, no. 5, 2015.

- [16] J.-W. Hsieh, S.-H. Yu, Y.-S. Chen, and W.-F. Hu, "Automatic traffic surveillance system for vehicle tracking and classification," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 2, pp. 175–187, 2006, doi: 10.1109/TITS.2006.874722.
- [17] X. Bai, M. Huang, M. Xu, and J. Liu, "Reconfiguration Optimization of Relative Motion Between Elliptical Orbits Using Lyapunov-Floquet Transformation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 59, no. 2, pp. 923–936, 2023, doi: 10.1109/TAES.2022.3193089.
- [18] H. Min et al., "A fault diagnosis framework for autonomous vehicles with sensor self-diagnosis," *Expert Systems with Applications*, vol. 224, p. 120002, 2023, doi: <https://doi.org/10.1016/j.eswa.2023.120002>.
- [19] H. Pan, S. Guan, and X. Zhao, "LVD-YOLO: An efficient lightweight vehicle detection model for intelligent transportation systems," *Image and Vision Computing*, vol. 151, p. 105276, 2024, doi: 10.1016/j.imavis.2024.105276.
- [20] X. Zhu, F. Chen, Y. Zheng, C. Chen, and X. Peng, "Detection of Camellia oleifera fruit maturity in orchards based on modified lightweight YOLO," *Computers and Electronics in Agriculture*, vol. 226, p. 109471, 2024, doi: 10.1016/j.compag.2024.109471.
- [21] B. Wu, S. Feng, S. Jiang, S. Luo, X. Zhao, and J. Zhao, "EB-YOLO: An efficient and lightweight blood cell detector based on the YOLO algorithm," *Computers in Biology and Medicine*, vol. 192, no. December 2024, 2025, doi: 10.1016/j.combiomed.2025.110288.
- [22] M. Li and N. Yan, "IPD-YOLO: Person detection in infrared images from UAV perspective based on improved YOLO11," *Digital Signal Processing: A Review Journal*, vol. 168, no. PA, p. 105469, 2026, doi: 10.1016/j.dsp.2025.105469.
- [23] C. Liu et al., "Grain-YOLO: An improved lightweight YOLO v8 and its android deployment for rice grains detection," *Computers and Electronics in Agriculture*, vol. 237, no. October 2024, 2025, doi: 10.1016/j.compag.2025.110757.
- [24] S. F. Stefenon, L. O. Seman, A. C. R. Klaar, R. G. Ovejero, and V. R. Q. Leithardt, "Hypertuned-YOLO for interpretable distribution power grid fault location based on EigenCAM," *Ain Shams Engineering Journal*, vol. 15, no. 6, p. 102722, 2024, doi: 10.1016/j.asej.2024.102722.
- [25] J. Wu, "Design and Exploration of Real Time Monitoring System for Power Grid Operation Safety Based on YOLO Object Detection Algorithm," *Procedia Computer Science*, vol. 228, pp. 889–897, 2023, doi: 10.1016/j.procs.2023.11.117.
- [26] L. Chen, G. Li, S. Zhang, W. Mao, and M. Zhang, "YOLO-SAG: An improved wildlife object detection algorithm based on YOLOv8n," *Ecological Informatics*, vol. 83, p. 102791, 2024, doi: 10.1016/j.ecoinf.2024.102791.
- [27] K. N. and C. D.N., "Optimized YOLOv2 based vehicle classification and tracking for intelligent transportation system," *Results in Control and Optimization*, vol. 2, p. 100008, 2021, doi: <https://doi.org/10.1016/j.rico.2021.100008>.