Journal of Engineering Science and Technology
Vol. 19, No. 1 (2024) 133 - 145

DETECTION OF VIETNAMESE TRAFFIC DANGER AND WARNING SIGNS VIA DEEP LEARNING

THI THUC ANH CU[1], XUAN CAN VUONG[1,*],
CHI TRUNG NGUYEN[1], TRONG THUAT VU[2], THI AN NGUYEN[3]

[1]Faculty of Transport Safety and Environment, University of Transport and Communications, No.3 Cau Giay Street, Lang Thuong Ward, Dong Da District, Hanoi, Vietnam
[2]Faculty of Electrical-Electronic Engineering, University of Transport and Communications, No.3 Cau Giay Street, Lang Thuong Ward, Dong Da District, Hanoi, Vietnam
[3]Faculty of Economical Mathematics, National Economics University, No. 207 Giai Phong Road, Hai Ba Trung District, Hanoi, Vietnam
*Corresponding Author: vuongcan@utc.edu.vn

Abstract

Traffic signs are among the most crucial components of the road transport infrastructure that guide and require road users to adapt their behaviour to traffic regulations and ensure traffic safety. Traffic sign detection is a crucial part of cutting-edge driver assistance systems and intelligent transportation systems that improve the safety of road users. Despite all the previous studies of traffic sign detection that have achieved some progress, it is still a difficult task due to the range of issues, such as the change in environmental conditions, traffic conditions, different countries, real-time processing, and so on. This paper introduces the implementation of YOLOv5 (You Only Look Once) architecture for the automatic detection of Vietnamese danger and warning signs (VDWSs). Firstly, this paper proposes a dataset of VDWSs with more than 11138 real traffic scene images of ten classes of VDWSs. The dataset is partitioned into three parts, including the training set, the validation set, and the testing set. And then, a training model is conducted by YOLOv5. Finally, a comparison is performed to evaluate the proposed method. The results of the experiment show that the proposed method has achieved over 98% accuracy. Comparison results highlighted that the proposed method reached higher performance than other methods. The proposed method can aid in the development of more accurate, robust traffic sign identification algorithms in Vietnam

Keywords: Deep learning, Traffic sign detection, Vietnamese danger and warning signs, YOLO.

1. Introduction

Traffic danger and warning signs indicate the potential dangers ahead on the road. Road users need to notice these signs so that they can be more proactive in recognizing and handling situations on the road, reducing the risk of traffic accidents. Recently, automatic traffic sign detection (TSD) in general and warning sign detection, in particular, have received an increasing interest to increase driving safety and reduce road accidents, especially for sign inventory and maintenance, advanced driver assistance systems, and intelligent autonomous vehicles.

The TSD is for predicting whether a given image contains a traffic sign and for performing classification and locating traffic signs. In general, there can be many traffic signs in an image and each traffic sign can appear in different places with different sizes. The main tasks of the TSD consist of localization in the bounding box and classification correctly. The TSD has been developed for decades and has some results, but it has still faced big challenges due to environmental changes during sunny, rainy, shadows, and so on. Besides, complex traffic conditions and object occlusions affect the TSD performance as well. Some datasets for TSD, such as GTSDB (German Traffic Sign Detection Benchmark) [1],

BTSD (Belgian Traffic Signs Dataset) [2], STSD (Swedish Traffic Signs Dataset), CTSD (Chinese Traffic Sign Dataset) [3], etc. have limited volume, insufficient annotation data, single-image styles, and so on. They cannot effectively cover all possible scenarios in different countries [4]. Therefore, it is essential to create detection algorithms and methods suitable to the traffic environment and types of traffic signs due to the semantic changes of traffic signs in different countries.

The methods of the TSD may be categorized into three main classes, colour-based, shape-based, and learning-based methods (including deep learning) [5]. The detection principle of the first two classes relies on the colour and shape of signs. In recent years, the learning-based methods, especially deep learning methods are very interested in research and development and have also obtained certain achievements due to the robustness of algorithms by self-learning. The typical deep learning methods applied to TSD consist of the R-CNN (Region-based Convolutional Neural Networks) series [6], the YOLO series [7], and SSD (Single Shot Multibox Detector) [8]. However, the algorithm performances vary for different image attributes. Moreover, the indicators of the performance in these algorithms are not uniform [4].

Therefore, the main purpose of this study is to detect traffic danger and warning signs in Vietnam via YOLOv5. The main reasons for choosing the YOLOv5 for this study include as follows: 1) it is currently the most advanced algorithm in the field of fast object detection with a lightweight architecture; 2) It allows model training to be performed using small computational resources and low cost; and 3) it is small size, so it can allow applications on mobile devices. Hence, the primary goal of this study is to provide the TSD for Vietnamese Danger and Warning Signs (VDWSs). The main features of the method include:

- Establishment of a dataset of VDWSs.
- Utilization of YOLOv5 to detect VDWSs
- Evaluation of the method's effectiveness

2. Background

2.1. Vietnamese danger and warning signs (VDWSs)

According to Article 31 of the National Technical Regulation on Traffic Signs and Signals (Regulation No. QCVN 41: 2019/BGTVT) issued by the Ministry of Transport of Vietnam, Vietnamese danger and warning signs (VDWSs) are used to warn road users of the nature of danger or precautions to be taken on the road. When encountering the VDWSs, road users must slow down to the necessary level, pay attention to observation, and be prepared to handle possible situations to prevent accidents [9]. The VDWSs are in a yellow background, red border, and black pattern. The shape of the main VDWSs is an equilateral triangle with an upward top angle (except for signs of yield to the main road (W.208), right-angle railroad crossing (W.242a, b), and non-right-angle railroad crossing (W.243a, b, c)). The VDWSs contain 84 different signs from W.201a to W.247, and some of the VDWSs are shown in Fig. 1.

W.201a
Bend to left

W.201b
Bend to right

W.202a
Double bend
first to left

W.202b
Double bend
first to right

Fig. 1. Some of the VDWSs.

2.2. Detection of traffic signs using deep learning

Detection approaches for traffic signs using deep learning can be split into two-stage methods and one-stage methods. The former methods create a series of candidate regions that may include features, classify each region based on its features, and then perform bounding box regression on each candidate region using convolutional neural networks (CNNs). Therefore, they are also called region-based CNNs, such as R-CNN, Fast R-CNN [10], and Faster R-CNN [6]. Whereas the one-stage methods instead of creating candidate regions, use a CNN to regress the location and categorization of all the objects in the entire image. Typical one-stage methods consist of the YOLO algorithms [11] and the SSD algorithm (Single Shot multi-box Detector) [8].

Qian et al. [12] conducted a detection of traffic signs on a custom dataset using Fast R-CNN and a hybrid region proposal. Their method achieved a generally average precision of 85.58%. Boujemaa et al. [13] detected German traffic signs through fast R-CNN and the CNN with the colour segmentation technique (C-CNN) and they concluded that the Fast R-CNN was much faster than the C-CNN. Zhang et al. [14] detected small traffic signs via an improved faster R-CNN. The performance of their method was better than SSD with VGG16 (Visual Geometry Group). Saleh et al. [15] recognized traffic signs in Germany using YOLOv2. Their method achieved a recognition accuracy of 97% in real-time under different real-world scenarios. Sánchez [16] used YOLOv3-tiny to detect speed traffic signs on a simulator, which include signs of 30 km/h, 60 km/h, and 90 km/h. The detection procedure of the model was obtained satisfactorily with a precision value of 0.92. Qin and Yan [17] used Faster R-CNN and YOLOv5 to recognize New Zealand traffic signs. Naimi [18] used SSD with MobileNet-v2 to detect traffic signs and lights, including speed limit, stop, crosswalk, and traffic light. His method has higher accuracy and faster recognition than the original SSD with VGG16. Zhang et al. [4] proposed a new Chinese TSD benchmark and conducted a comprehensive evaluation of nine algorithms of the TSD on their benchmark, for instance, the R-CNN series, YOLO series, SSD, and RetinaNet. In general, the advantages of the two-stage method are detection accuracies. However, the procedure requires a lot of time and resources and has a low computation efficiency. Despite a reduction in process accuracy, the one-stage methods are greatly expedited by a unified network structure. In addition, these methods are affected by the amount of data.

In Vietnam, current studies related to TSD are still quite modest, most of them using traditional methods with limited datasets or using datasets available from abroad. Moreover, it has not yet focused on a specific group of signs, such as danger and warning signs.

From the results achieved abroad and the current state of research in the country, in this study, this paper initially builds a deep learning network using YOLOv5 for offline identification of some dangerous and warning signs in Vietnam.

3.Materials and methods

3.1. Brief of YOLOv5 and production process

YOLO as one of the faster algorithms predicts all bounding boxes for an image by using a single neural network to collect features from each part of an image. YOLO creates a $S \times S$ grid from an image input. A grid cell will be utilized to detect an object when its centre overlaps with that cell. The number of bounding boxes and the confidence stores associated with each box should be predicted for each grid cell. The score of confidence is defined as 0 if no object falls within a grid cell [19, 20].

Recently, the YOLO network has been released in many versions, such as YOLOv2 [11], YOLOv3 [21], YOLOv4 [22], YOLOv5 [23], and other extended versions. The easier implementation and other enhancements seen in YOLOv5, which was built using PyTorch rather than Darknet like earlier versions, include anchor boxes that automatically learn and enrich mosaic data. As a result, YOLOv5 improves accessibility for real-time object recognition and prediction performance in terms of training speed or accuracy.

The architecture of YOLOv5 has three main components, namely the model backbone, neck model, and head model (see Fig. 2 [24]). The backbone model using Cross Stage Partial (CSP) networks [25] is mainly responsible for the extraction of important features from images. Enhancing image features, processing and improving the shallow features extracted from the backbone, merging the shallow features with the deep features to improve the robustness of the network, and obtaining more useful features are all accomplished using the neck model using Path Aggregation Network (PANet) [26]. The head model based on YOLOv3 and YOLOv4 is used for classification and regresses the features achieved by the backbone and neck models.

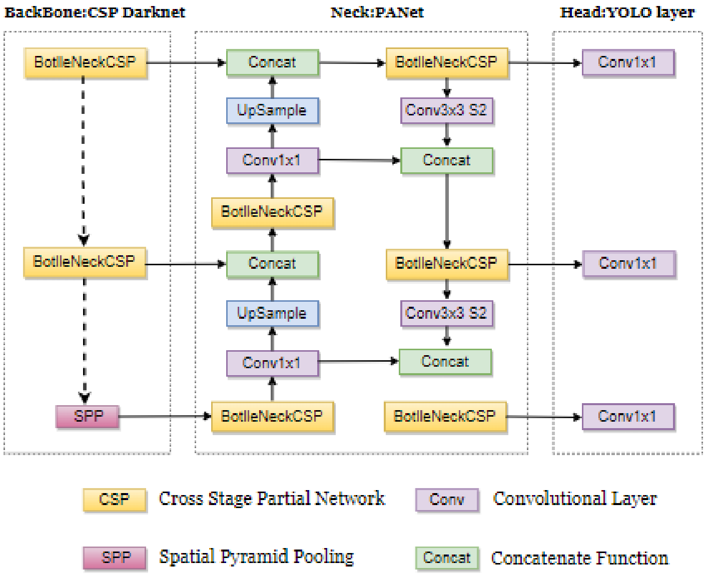

Fig. 2. The architecture overview of YOLOv5.

Four distinct size models are offered by YOLOv5, including YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. Due to the relatively limited size of the dataset for this investigation, a pre-trained model of YOLOv5m (21.8M params) is used to perform this study. Figure 3 depicts the study's production procedure.

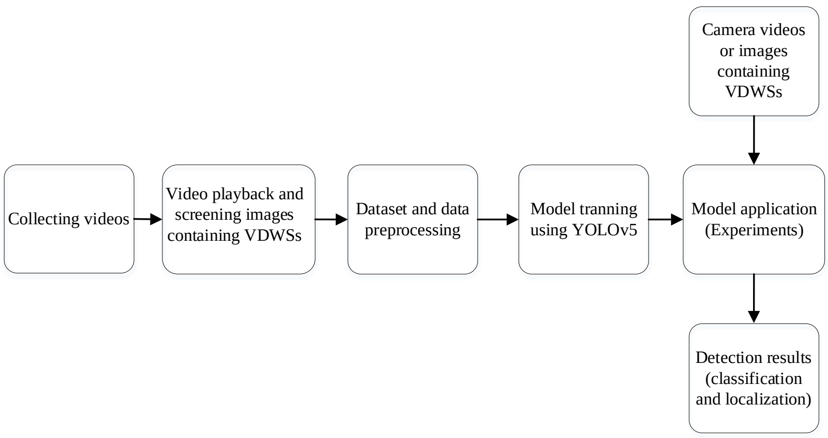

Fig. 3. The production process of this study.

The steps of the production process are as follows:

- Step 1: Collecting videos. Videos containing VDWSs on the roads of Vietnam are collected to establish a dataset
- Step 2: Video playback and screening images containing VDWSs. After completing the first step, the videos are played back. When detecting an image containing a VDWS, it will proceed to separate that image.

- Step 3: Dataset and data pre-processing. The data on the signs containing the images in the previous step is used to perform the labelling and determine the position of signs in the image. Next, the data is divided into 03 sets, including the training set, the validation set, and the testing set.
- Step 4: Training model using YOLOv5. The parameters for YOLOv5 are configured and the first two sets (i.e., the training set and the validation set) are used to train the model. The model after successfully training the weights is saved in the format "last.pt"
- Step 5: Model application. The trained model is used for the execution application with input from images, videos, and cameras. The result of the identification process will directly display the type of sign and the position of the sign on the image and save the identification image or video. Details are shown in the following sections.

3.2. Dataset and data preprocessing

In this study, a camera is used to capture realistic images of VDWSs in Vietnam with different environmental conditions, such as in sunlight, cool weather, cloudy weather, obscured by shade, and partially damaged. The pixels of the cropped images are 900×900 and stored in *.PNG format. The dataset consists of 11138 images with an average of one instance per image in a total of ten classes (categories) as follows: W.201a-Bend to left (1649 images); W.201b-Bend to right (1070 images); W.202a-Double bend first to left (1619 images); W.202b-Double bend first to right (1739 images); 207a- Road junction with priority (785 images); W.207b-Road junction with priority (1369 images); W.207c-Road junction with priority (697 images); W.207d-Road junction with priority (808 images); W.225-School zone ahead (651 images); W.245a-Slow (751 images). Traffic sign classes in the dataset are described in Table 1 as follows.

Table. 1. Traffic sign classes in the dataset.

No.	Name of sign	Shape of sign	Images	Instances
1	W.201a Bend to left		1649	1658
2	W.201b Bend to right		1070	1073
3	W.202a Double bend first to left		1619	1624
4	W.202b Double bend first to right		1739	1739
5	W207aRoad intersection with priority		785	793
6	W.207b Road junction with priority		1369	1438

7	W.207c Road junction with priority		697	771
8	W.207d Road junction with priority		808	882
9	W.225 School zone ahead		651	664
10	W.245a Slow		751	790

3.3. Training model

To train the YOLOv5 model, firstly each image contains an annotation file with the bounding box coordinates and the sign's class ID based on the YOLO format. The annotation file is generated by the LabelImg tool [27] and is saved as a *.txt file. After that, the dataset with both image and annotation files is divided into three sets, 70% for training, 20% for validation, and 10% for testing. The training set, the validation set, and the testing set consist of 7788, 2237, and 1113 images, respectively. The first two sets are used to train the model, and the testing set is used to evaluate the performance of the model.

To avoid training the YOLOv5 on the CPU, the training process used the free Jupyter Notebook from Google called Google Colab [28]. This service enables the execution and processing of some code in the cloud, giving users free access to extremely powerful resources.

Before performing model training, the hyper-parameters are adjusted to help determine more optimal parameters for the training dataset. The hyper-parameters of this study are set as follows. The training model has an initial learning rate of 0.01, a momentum size of 0.937, and a weight decay parameter of 0.005. The maximum value of epochs is 300 and training time takes 5.843 hours. The results of the training model are shown in Figs. 4 and 5.

As demonstrated in Fig. 5, the trained model's performance is evaluated after each iteration. The trained model performs better, and the loss function value decreases as a result. There are three types of loss as shown in Fig. 6, including bounding loss (box_loss), object loss (obj_loss), and classification loss (cls_loss). The loss of the bounding box depicts how well the algorithm can locate the centre of an object and predict the bounding box of an object.

The likelihood that an object will be found in a potential area of interest is measured by object loss. The classification loss gives insight into how well the algorithm can predict the class of a given object. In this study, the losses are very small (within 1.5%), so the model is trained with high reliability. Besides, after training the model, the best precision and recall are 0.999 for both.

Fig. 4. The training batch of YOLOv5.

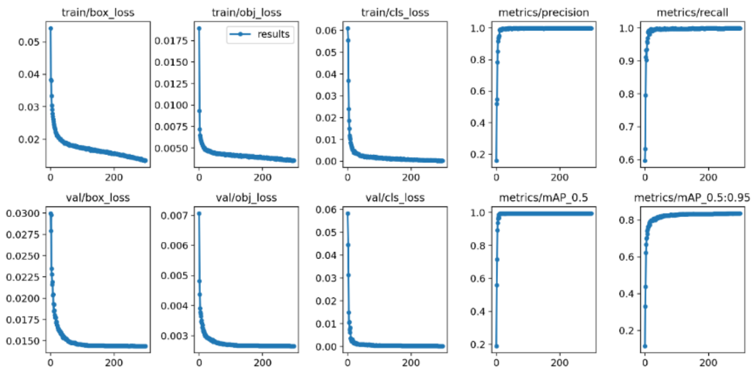

Fig. 5. The results of model training.

The mean average precision (mAP) with IoU (intersection over union) thresholds of 0.5 is 0.995. The mAP with IoU thresholds from 0.5 to 0.95 is 0.836. These promising results show great potential in using YOLOv5 deep learning architecture to train the model in TSD in particular, as well as other recognition requirements in general in the future. The larger the dataset, the better, faster, and more accurate the object detection will be.

4. Experiments

This paper designs 1113 test images with 1144 instances from the testing set to evaluate the model performance. The testing environment is the free Jupyter Notebook from Google Colab [28] with a GPU Tesla T4. The input data is the testing image, and the detection result is the saved image with the position of the sign and the accuracy of the prediction of each sign. Image is detected based on the trained model with a confidence of 0.25 and an IOU threshold of 0.45. Figures 6 and 7 show some results of TSD on the testing set. There are several images with non-detected signs in Fig. 7 because the size of signs in the images is smaller than compared to the identification capacity of the algorithm.

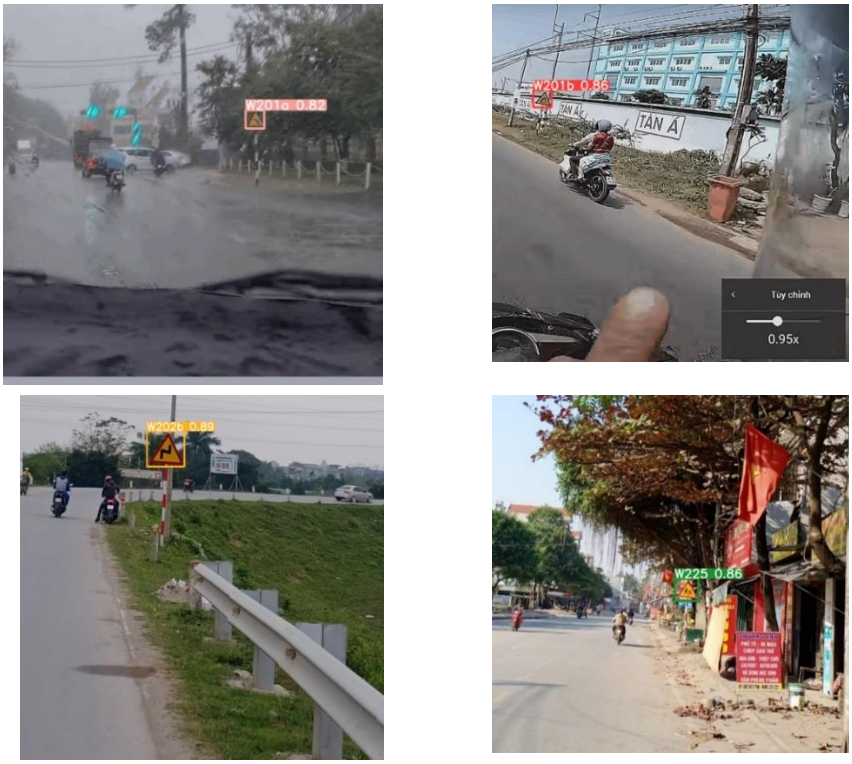

Fig. 6. Some detection examples of the method in the testing set.

Fig. 7. Two images with non-detected signs.

In this paper, two indicators, including accuracy and speed of detection (FPS) are used to assess the performance of the model. The accuracy of the model is determined by two parameters, including precision (P) and recall (R). Of which, precision represents accuracy in predicting sign classes (i.e., names of objects) and locations. The recall represents the ability to detect signs in the input data of traffic signs. The higher the precision and recall, the better, and they are calculated as follows:

$$P = \frac{TP}{TP+FP} \tag{1}$$

$$R = \frac{TP}{TP+FN} \tag{2}$$

where TP - True Positive; FP - False Positive; FN - False Negative.

In addition, this paper has compared the results with the published method (YOLOv4 and SSD with Mobilenet). Comparisons are made under the same conditions, for example using the same dataset and executing on the same hardware platform. The evaluation results are shown in Table 2.

Table 2. The evaluation results of the model performance.

Method	*TP*	*FP*	*FN*	*P* (%)	*R* (%)	FPS
Mobilenet-SSD	863	0	281	100	75.44	21.6
YOLOv4	1125	5	19	99.56	98.34	2.7
YOLOv5	1125	0	19	100	98.34	78.7

From the experimental results, the proposed method using YOLOv5 gives better performance than the methods using YOLOv4 and Mobilenet-SSD in terms of accuracy and speed of detection. Detection results of YOLOv5 with accuracy up to 98%. Table 2 shows that the proposed method using YOLOv5 gives a much faster processing speed than other methods. With a fast detection rate of 12.7ms/frame (equivalent to 78.7 FPS), enabling real-time detection implementation. The accuracy and processing speed of YOLOv5 than YOLOv4 is since it has added some new features and improvements based on a more complex architecture. Compared to Mobilenet-SSD, YOLO has a superior ability to identify signs that are small compared to the size of the image. It is explained that YOLO and Mobilenet-SSD have certain differences in how to detect objects as well as how to build boxes to locate and determine the boundaries of objects in the image.

In addition, the exact results of precision and recall depend on many factors, such as background lighting conditions and the size of objects in the images. Hence, the method works better with large sizes of signs.

Traffic signs play an important role in traffic management and ensuring traffic safety on the road, especially in a mixed and complex traffic environment like Vietnam [29, 30]. Detecting and identifying traffic signs contributes significantly to promoting the effective use of signs and traffic safety. This research result is an important basis for promoting policies in the development of intelligent systems on vehicles as well as autonomous vehicles in Vietnam.

5. Conclusions

In this paper, the method of TSD based on YOLOv5 is proposed to detect a set of 10 classes of danger and warning signs in Vietnam. YOLOv5 is a single neural network that in a single assessment, directly produces bounding boxes and probabilities from an entire image. It achieves a greater recall rate with fewer false positives and offers excellent precision with quick processing while detecting traffic signs.

The proposed method builds a dataset and then performs model training and testing. The collection includes 11138 images of actual traffic scenarios along with their related in-depth annotations. Compared with other methods (YOLOv4 and Mobilenet-SSD), the proposed method using YOLOv5 can be more efficient and accurate. Besides, the method using YOLOv5 has been successfully tested by Google Colab, which may pave the way for a fresh method of real-time TSD execution on mobile devices with reduced errors as an increase in traffic sign classes and samples.

The model also needs to be improved in some respects, such as increasing the sign classes and number of samples in different environments and developing a real-time system on smartphones and autonomous vehicles. Besides, the model will be compared with many other deep learning methods to see its performance.

Acknowledgments

This research is partly funded by the University of Transport and Communications under grant number T2023-MT-002.

References

1. Houben, S.; Stallkamp, J.; Salmen, J.; Schlipsing, M.; and Igel, C. (2013). Detection of traffic signs in real-world images: The German traffic sign detection benchmark. *Proceedings of the* 2013 *International Joint Conference on Neural Networks* (*IJCNN*), Dallas, TX, USA,1-8.
2. Timofte, R.; Zimmermann, K.; and Van, G.L. (2014). Multi-view traffic sign detection, recognition, and 3D localisation. *Machine Vision and Applications*, 25(3), 633-647.
3. Yang, Y.; Luo, H.; Xu, H.; and Wu, F. (2015). Towards real-time traffic sign detection and classification. *IEEE Transactions on Intelligent transportation systems*, 17(7), 2022-2031.
4. Zhang, J.; Zou, X.; Kuang, L-D.; Wang, J.; Sherratt, R.S.; and Yu, X. (2022). CCTSDB 2021: A more comprehensive traffic sign detection benchmark. *Human-centric Computing and Information Sciences*,12(23), 1-18.
5. Saadna, Y.; and Behloul, A. (2017). An overview of traffic sign detection and classification methods. *International Journal of Multimedia Information Retrieval*, 6(3),193-210.
6. Ren, S.; He, K.; Girshick, R.; and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems* 28, Montreal, Quebec, Canada, 91-99.
7. Redmon, J.; and Farhadi, A. (2017). YOLO9000: better, faster, stronger. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, USA, 7263-7271.

8. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C-Y; and Berg A.C. (2016). Ssd: Single shot multibox detector. *Proceedings of the European Conference on Computer Vision*, Amsterdam, The Netherlands, 21-37.
9. Ministry of Transport of Vietnam. (2019). National technical regulation on traffic signs and signals. *QCVN* 41*:* 2019*/BGTVT*.
10. Girshick, R. (2015). Fast r-cnn. *Proceedings of the IEEE international Conference on Computer Vision*, Santiago, Chile, 1440-1448.
11. Redmon, J.; Divvala, S.; Girshick, R.; and Farhadi, A. (2016).You only look once: Unified, real-time object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, USA 779-788.
12. Qian, R.; Liu, Q.; Yue, Y.; Coenen, F.; and Zhang, B. (2016). Road surface traffic sign detection with hybrid region proposal and fast R-CNN. *Proceedings of the* 2016 12*th International Conference on Natural Computation*, *Fuzzy Systems and Knowledge Discovery* (*ICNC-FSKD*), Changsha, China, 555-559.
13. Boujemaa, K.S.; Berrada, I.; Bouhoute, A.; Boubouh, K. (2017). Traffic sign recognition using convolutional neural networks. *Proceedings of the* 2017 *International Conference on Wireless Networks and Mobile Communications* (*WINCOM*), Rabat, Morocco,1-6.
14. Zhang, Z.; Zhou, X.; Chan, S.; Chen, S.; and Liu, H. (2017). Faster R-CNN for small traffic sign detection. *Proceedings of the CCF Chinese Conference on Computer Vision*, Tianjin, China, 155-65.
15. Saleh, S.; Khwandah, S.A.; Heller, A.; Mumtaz, A.; and Hardt, W. (2019).Traffic signs recognition and distance estimation using a monocular camera. *Proceedings of the* 6*th* *International Conference Actual Problems of System and Software Engineering[online]*, Moscow, Russia, 407-418.
16. Sánchez, J.M. (2019). *Speed traffic sign detection on the CARLA simulator using YOLO*. MSc dissertation, Universitat Pompeu Fabra.
17. Qin, Z.; and Yan, W.Q. (2021). Traffic-sign recognition using deep learning. *Geometry and Vision: Proceedings of the First International Symposium, ISGV* 2021, Auckland, New Zealand,13-25.
18. Naimi, H. (2021). *Traffic sign and light detection using deep learning for automotive applications*. MSc dissertation, Department of Electrical and Computer Engineering,University of Windsor.
19. Vuong, X.C.; Phan, X.V.; Mou, R.F.; Vu, T.T.; Vu V.D.; and Nguyen D.N. (2021). Vehicle Detection and Counting Under Mixed Traffic Conditions in Vietnam Using Yolov4. *International Journal Advanced Research Engineering a Technology*,12(2), 722-730.
20. Vuong, X.C.; Mou, R.F.; and Vu, T.T. (2022). Vehicle Tracking and Speed Estimation under Mixed Traffic Conditions using YOLOv4 and SORT: A Case Study of Hanoi. *Transport Problems: an International Scientific Journal*, 17(4),17-26.
21. Redmon, J.; Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv*, arXiv:1804.02767,1-6
22. Bochkovskiy, A.; Wang C-Y.; and Liao H-YM. (2020). Yolov4: Optimal speed and accuracy of object detection. *arXiv*, arXiv:2004.10934,1-17.

23. Glenn, J. (2021). ultralytics/yolov5. Retrieved November 11, 2021, from https://github.com/ultralytics/yolov5.
24. ultralytics/yolov5. (2020). Overview of model structure about YOLOv5 #280. Retrieved November 11, 2021, from January 19, 2023, from https://github.com/ ultralytics/yolov5/issues/280.
25. Wang, C-Y.; Liao, H-Y.M.; Wu, Y-H.; Chen, P-Y. Hsieh, J-W.; and Yeh, I-H. (2020). CSPNet: A new backbone that can enhance learning capability of CNN. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, Seattle, USA, 390-391.
26. Liu, S.; Qi, L.; Qin, H.; Shi, J.; and Jia, J. (2018). Path aggregation network for instance segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition*, Salt Lake City, USA, 8759-8768.
27. Tzutalin. (2020). labelImg. Retrieved December 25, 2020, from https://github.com/tzutalin/labelImg.
28. Google. (2022). Welcome to Colaboratory. Retrieved October 9, 2022, from https://colab.research.google.com/.
29. Vuong, X-C.; Mou, R-F.; Nguyen, H-S.; Vu, T-T. (2018). Signal timing optimization of isolated intersection for mixed traffic flow in Hanoi City of Vietnam using VISSIM. *Proceedings of the International Conference on Smart Vehicular Technology*, *Transportation*, *Communication and Applications*, Mount Emei, China, 133-139.
30. Vuong, X-C.; Mou, R-F.; and Vu, T-T. (2019). Safety impact of timing optimization at mixed-traffic intersections based on simulated conflicts: a case study of Hanoi, Vietnam. *Proceedings of the* 2019 4*th* *International Conference on Intelligent Transportation Engineering* (*ICITE*), Singapore, 247-251.