An approach for traffic sign recognition

Dat Tien Nguyen[1], Quan Minh Vu[1], Thanh Hoang Nguyen[1], Khai Quang Ho[1],
Phuong Quang Luu[1], Thanh Huu Duong[1*]

[1]Ho Chi Minh City Open University, Ho Chi Minh City, Vietnam
[*]Corresponding author: thanh.dh@ou.edu.vn

ARTICLE INFO

DOI:10.46223/HCMCOUJS.tech.en.15.1.3350.2025

Received: April 06th, 2024

Revised: June 12th, 2024

Accepted: July 10th, 2024

Keywords:

FPS: Frame Per Second; mAP: mean Average Precision; NMS: Non-Maximum Suppression; object detection; YOLO: You Only Look One

ABSTRACT

This article presents a model for detecting and recognizing traffic signs based on the YOLO (You Only Look Once) algorithm. Our system can detect traffic signs in real-world scenarios, including prohibitory, stop, no entry, speed limit, regulatory, and hazardous signs. However, there are still some cases where successful recognition is not achieved. Experiments were conducted on a dataset of 29,632 images, yielding % recognition accuracy of 86.8%. The system performs well in practical environments with relatively high accuracy, yet some errors persist during detection.

1. Introduction

The application of information technology in daily life is gradually becoming more common, with people tending to adopt technology and innovative city models to address social issues, and traffic issues are among the hot topics. The application of traffic sign recognition helps to address problems related to misinterpretation of traffic signs, leading to traffic congestion and preventable accidents. Detecting and recognizing traffic signs is an intelligent tool supporting safer traffic participation. Such intelligent systems are strongly developed worldwide, with works like Yu et al. (2022) "Traffic Sign Detection and Recognition in Multi Images Using a Fusion Model With YOLO and VGG Network"; etc. In Vietnam, some research works by Truong et al. (2015), "Detection and Recognition of Road Traffic Signs Using HOG Features and Artificial Neural Networks," etc.

This article introduces a technique for detecting and recognizing traffic signs utilizing the YOLO model. YOLO leverages state-of-the-art developments in deep learning and computer vision, delivering unmatched performance in speed and accuracy. Its efficient architecture renders it applicable across various scenarios and readily deployable on diverse hardware platforms, from edge devices to cloud APIs. To develop an application for traffic sign detection and recognition that meets everyday life needs, we collected and trained the model on a dataset of 29,632 images, achieving an overall accuracy of 86.8%.

Throughout our research process, we have recognized that there are still numerous challenges in attempting to improve accuracy and apply the project to real-life scenarios. Firstly, we must search for and gather images to enhance accuracy and create a highly reliable dataset. Additionally, we must consider related conditions such as blurriness, occlusion, and weather conditions affecting the photos. Once sufficient scenes have been collected, we label each object, also known as preprocessing. Images containing traffic signs captured from real-world scenarios often lack proper annotations, making it challenging to tag them with the names of specific signs they contain. This limitation hinders the use of traditional methods relying on pre-tagged images.

Additionally, benchmark datasets must include images without traffic signs to simulate real-world scenarios effectively. This enables a detector to differentiate between genuine traffic signs and similar-looking objects. After curating a diverse dataset, we train the model and assess its reliability. We compare and enhance dataset reliability through multiple training iterations, identifying and rectifying deficiencies to achieve optimal results. Ultimately, we select the dataset with the highest accuracy for utilization.

2. Related studies

In recent years, traffic sign detection and recognition have witnessed significant advancements, reflecting the increasing demand for intelligent transportation systems and road safety measures. Notable studies have explored and proposed methods and techniques to enhance the accuracy and effectiveness of traffic sign recognition systems.

In India, a YOLOv4 research group in traffic sign recognition and detection (Saxena et al., 2023). This work introduces a modified YOLOv4-based deep learning model for robust traffic sign detection and recognition in challenging environments. The model demonstrates superior performance by incorporating CSPDarknet53 as the backbone and employing innovative techniques such as anchor box calculation using GIoU and nighttime image enhancement. Experimental results on diverse datasets, including MTSD, TT-100K, and an Indian traffic sign dataset, showcase remarkable accuracy improvements over existing methods. However, there remain challenges in extending the model's performance to different datasets, as evidenced by variations in accuracy on cross-data experiments.

A notable study by Truong et al. (2015) proposed a method for detecting and recognizing traffic signs using Histogram of Oriented Gradients (HOG) features and artificial neural networks. This work highlights the effectiveness of combining feature extraction techniques with machine learning algorithms for accurate traffic sign recognition.

Another study by Zhu et al. (2016) focused on detecting and classifying traffic signs in real-world environments. This approach addresses challenges in detecting and classifying traffic signs in real-world settings, promoting the development of autonomous driving systems and intelligent transportation.

Another proposed model combines YOLO and VGG networks for multi-image traffic sign detection and recognition (Yu et al., 2022). This approach aims to enhance the robustness and accuracy of the traffic sign recognition system in diverse real-world scenarios.

These studies all contribute positively to improving the performance and accuracy of traffic sign recognition systems, thereby enhancing traffic safety and driving forward the development of related technologies.

3. The research method

3.1. Dataset

We have collected common traffic signs in Vietnam and categorized them into 07 classes, as shown in Table 1. The total number of images in this dataset, including 07 classes, is approximately 29,632, evenly distributed across each class. We have labeled each class for every traffic sign appearing in the images. To prepare the dataset for model training, we preprocessed the data to generate additional exceptional cases. We adjusted the image sizes to enhance the model's ability to recognize the traffic signs during training.

Table 1

A Dataset for Traffic Sign Recognition

Example	Training dataset	Test dataset	Validation dataset	Total	Class	Label
	634	676	621	1,931	0	Prohibition Traffic Sign
	885	737	747	2,369	1	No parking, stopping traffic sign
	341	695	633	1,669	2	No entry traffic sign
	792	349	358	1,499	3	No turn traffic sign
	612	496	434	1,542	4	Limit speed traffic sign
	464	532	522	1,518	5	Mandatory traffic sign
	547	1402	1469	3,418	6	Danger traffic sign

Source. Google Images (https://images.google.com/)

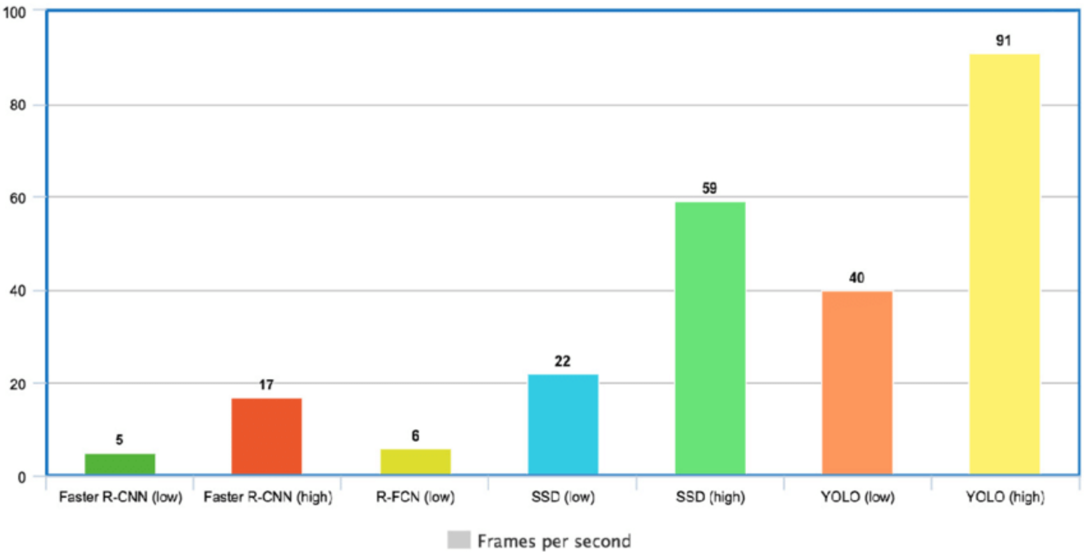
3.2. YOLO

Numerous methods are available for object detection, such as using HOG and artificial neural networks (Truong et al., 2015), employing Deep Learning, and CNN (Amjoud & Amrouch, 2023). Here, we utilize the YOLO algorithm. We choose YOLO for tackling the task of detecting and recognizing traffic signs for the following reasons:

- **Speed:** Looking at Figure 1, we can see exactly the difference between YOLO and others. YOLO achieves exceptional speed due to its streamlined approach, eschewing complex pipelines. It can rapidly process images at 45 Frames Per Second (FPS). Moreover, YOLO outperforms other real-time systems by doubling the mean Average Precision (mAP), making it highly suitable for real-time processing tasks. Observing the image below, it's evident that YOLO's performance far surpasses that of other object detectors, reaching an impressive 91 FPS.

Figure 1

YOLO Speed Compared to other State-of-The-Art Object Detectors



Source. The researcher's data analysis

- **Detection accuracy:** YOLO significantly surpasses other state-of-the-art models in detection accuracy, exhibiting minimal background errors.

- **Good generalization:** With its latest version, YOLO has taken a step forward by offering improved generalization to new domains. This enhancement further solidifies its suitability for applications that depend on fast and robust object detection.

- **Open-source:** The decision to make YOLO open-source has spurred continuous improvements by the community. This collaborative effort has played a pivotal role in the rapid advancements of YOLO within a relatively short timeframe.

3.2.1. YOLOv8

YOLO object detection system emphasizes adapting to real-time surface defect detection in industrial settings. As YOLO versions progress, particularly with YOLO-v5's focus on edge deployment, research interest remains high due to its open architecture and real-time capabilities.

YOLOv6 outperforms YOLOv5, YOLOX, and PPE-YOLOE in achieving higher mean Average Precision (mAP) scores on the COCO dataset across different Frames Per Second (FPS) rates. However, it's crucial to recognize that while the COCO dataset provides valuable insights, it may not perfectly reflect the performance of these models on your specific dataset.

YOLOv7 comprises several versions, with YOLOv7 being the primary model. YOLOv7-tiny is a compact variant tailored for efficient inference on edge devices. YOLOv7-W6, commonly employed in cloud computing, is another variant available.

YOLOv8 introduces a semantic segmentation model known as YOLOv8-Seg, which utilizes a CSPDarknet53 feature extractor as its backbone, followed by a C2f module instead of the traditional YOLO neck architecture. This module is succeeded by two segmentation heads responsible for predicting semantic segmentation masks based on the input image. Like YOLOv8, the model incorporates five detection modules and a prediction layer in its detection heads. YOLOv8-Seg has demonstrated state-of-the-art performance across object detection and semantic segmentation benchmarks while maintaining high speed and efficiency.

YOLOv8 can be executed via the Command Line Interface (CLI) or installed as a PIP package. Additionally, it offers multiple integrations for labeling, training, and deployment purposes.

With the improvements introduced in various versions of YOLO, it is evident that YOLOv8 stands out in object detection performance compared to the others. Therefore, our team utilized YOLOv8 for traffic sign detection and recognition.

3.2.2. YOLO architecture

The You Only Look Once (YOLO) architecture revolutionizes image object detection with real-time efficiency and high accuracy. By dividing the input image into a grid and making predictions directly, YOLO eliminates the need to detect multiple sliding windows. Its architecture comprises several crucial components working in tandem for effective object detection.

The YOLO architecture divides the input image into an S × S grid, with each grid cell responsible for detecting objects. The model predicts bounding boxes, associated confidence scores, and conditional class probabilities within each cell. This results in a tensor of predictions that encodes object localization and class probabilities (Redmon et al., 2016).

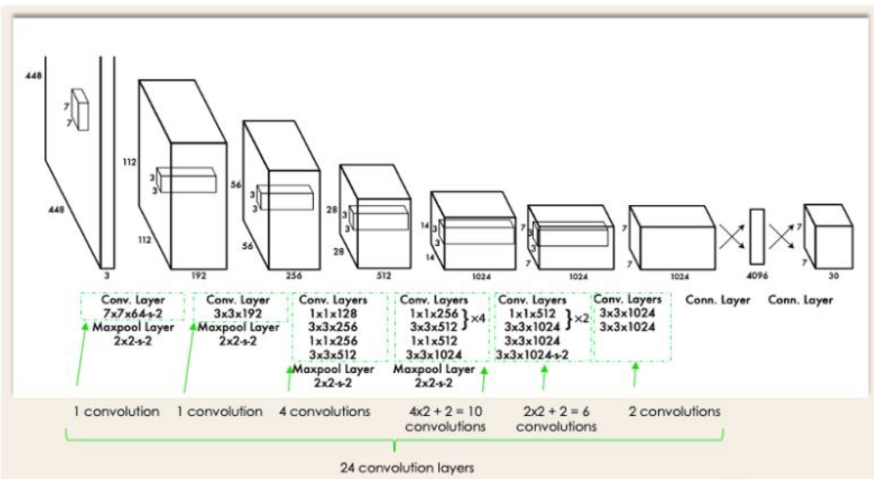
At the heart of the YOLO architecture lie the backbone convolutional layers, forming the bedrock for feature extraction. These layers, often based on architectures like Darknet or ResNet, extract hierarchical features from the input image, capturing low-level and high-level features crucial for accurate object detection (Terven et al., 2023).

Following feature extraction, the detection head of YOLO generates bounding boxes and associated confidence scores for detected objects through a series of convolutional layers and detection layers.

In Figure 2, the detection network architecture comprises 24 convolutional layers, succeeded by 02 fully connected layers. To condense the feature space from the preceding layers, alternating 01 × 01 convolutional layers are employed. Initially trained on the ImageNet classification task with an input image resolution of 224 × 224, these convolutional layers adapt to increased resolution twofold for the detection phase.

Figure 2

The Architecture of YOLO



Source. The data are from “You only look once: Unified, real-time object detection” by J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, 2016, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 779-788)

The network's output manifests as a tensor of predictions with dimensions 07×07×30. This tensor encompasses the predicted bounding boxes, confidence scores, and class probabilities for the detected objects.

The bounding box prediction process entails predicting bounding boxes directly from the feature maps generated by the backbone network. Each grid cell in the feature map predicts multiple bounding boxes, accompanied by corresponding confidence scores and class probabilities, facilitating precise localization and classification of objects. Subsequently, YOLO applies Non-Maximum Suppression (NMS) to eliminate redundant or overlapping boxes, ensuring each object is detected only once and enhancing overall precision.

During the training process, YOLO undergoes end-to-end training using labeled training data containing images annotated with bounding boxes around traffic signs. The model undergoes optimization using techniques like Stochastic Gradient Descent (SGD) or adaptive optimization algorithms to minimize a predefined loss function, measuring the disparity between predicted and ground-truth bounding boxes. In this research, the YOLOv8 architecture was chosen for its superior performance and efficiency in traffic sign recognition tasks, building upon the advancements of previous YOLO versions by incorporating improvements in feature extraction, bounding box prediction, and model optimization.

3.2.3. YOLO object detection

In the previous section, we presented the structure of YOLO, and in this section, we will discuss how the YOLO algorithm works for detecting and recognizing traffic signs.

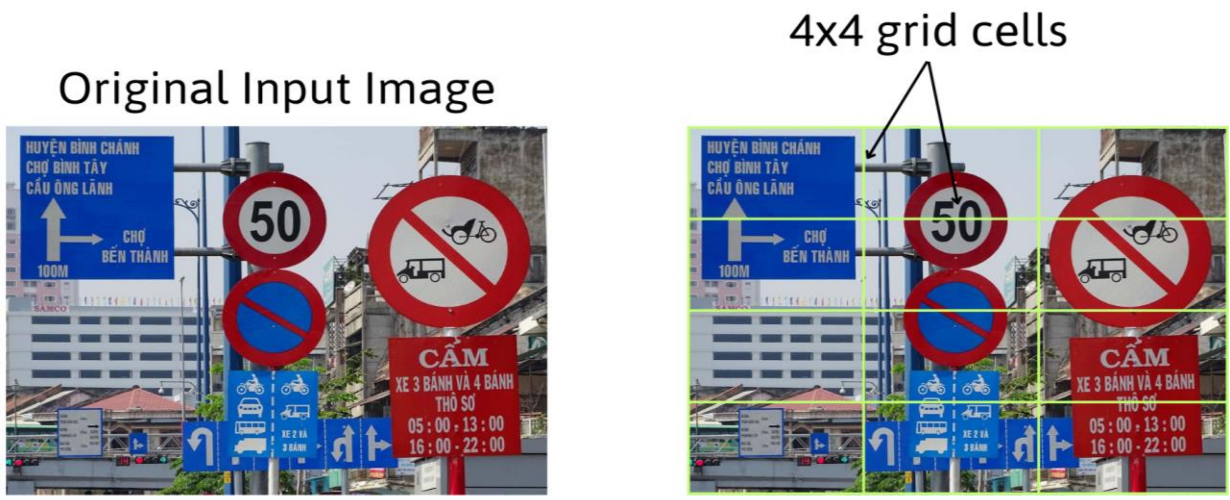
The YOLO algorithm works based on the following steps:

3.2.3.1. Residual blocks

To detect and recognize traffic signs easily, the YOLO algorithm divides the input image into an NxN grid with equal-sized cells, where N, in our case, is 04, as shown in the picture on the right. Each cell in the grid represents localizing and predicting the class of the object lying within the cell, along with the probability of each expected object.

Figure 3

YOLO Dividing Input Image into 4x4 Grid Cells



Source. The researcher's data analysis

3.2.3.2. Bounding box regression

The subsequent stage involves identifying bounding boxes that outline all objects within the image. The number of bounding boxes can match the number of objects in the picture.

YOLO employs a single regression module to determine the attributes of these bounding boxes, represented by Y in the following format: Y = [pc, bx, by, bh, bw, c1, c2].

This process is particularly crucial during the model's training phase.

The element pc denotes the probability score of the grid containing an object. For example, all grids highlighted in red will possess a probability score greater than zero (significant). The image on the right is a simplified version, as the probability of each yellow cell is zero (insignificant).

Figure 4

YOLO Determine the Bounding Box of Object based on the Probability

Source. The researcher's data analysis

3.2.3.3. Intersection over unions or IOU for short

Object detection algorithms can be divided into two main categories: single-shot detectors and two-stage detectors. YOLO is a single-shot detector that uses a fully convolutional neural network to process an image. The goal of the IOU (a value between 0 and 1) is to discard such grid boxes only to keep those relevant. Here is the logic behind it:

- The user defines its IOU selection threshold, which can be, for instance, 0.5.

- Then YOLO computes the IOU of each grid cell, which is the Intersection area divided by the Union Area.

- Finally, it ignores the prediction of the grid cells having an IOU ≤ threshold and considers those with an IOU > threshold.

3.2.3.4. Non-maximum suppression

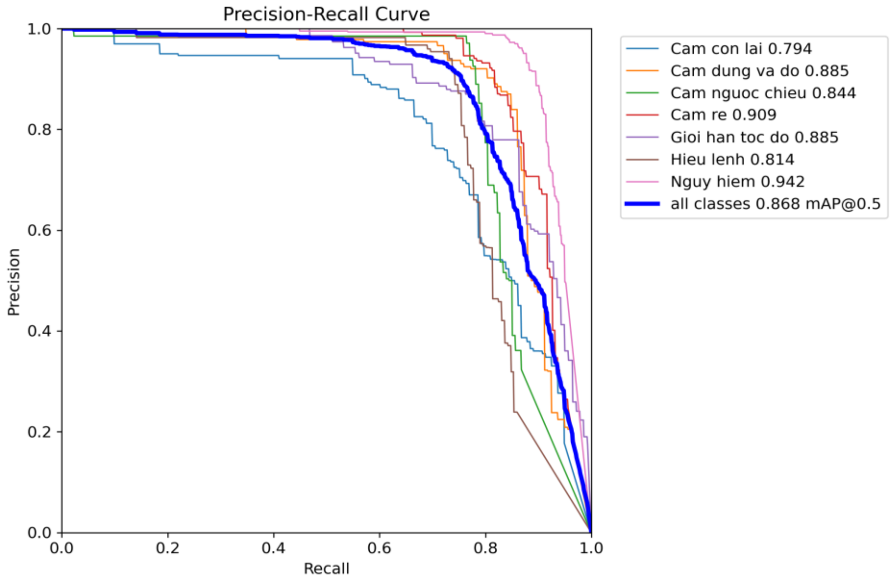
Merely establishing a threshold for the Intersection over Union (IOU) isn't always sufficient, as an object might generate multiple boxes with IOU surpassing the threshold. Retaining all such boxes could lead to noise. This is where Non-Maximum Suppression (NMS) comes into play, allowing us to keep only the boxes with the highest probability detection score.

4. Results and discussions

4.1. Training results

Figure 5

Precision-Recall (mAP) Curve



Source. Data analysis result of the research

In the task of traffic sign recognition and detection, Precision-Recall metrics are crucial. Figure 5 provides an overview of the model's accuracy for each class and all classes combined. This curve aids in selecting the optimal threshold to maximize both metrics.

Table 2

Map of Each Class in Traffic Sign Recognition

Class Name	True Prediction	False Prediction
Another traffic sign	79.4%	20.6%
No parking, stopping traffic sign	88.5%	11.5%
No entry traffic sign	84.4%	15.6%
No turn traffic sign	90.9%	9.1%
Limit speed traffic sign	88.5%	11.5%
Mandatory traffic sign	81.4%	18.6%
Danger traffic sign	94.2%	5.8%

Source. Data analysis result of the research

To be applicable in practical situations, achieving a higher level of accuracy is perhaps necessary, including data about adverse weather conditions and environmental factors. However, this is still quite an impressive outcome for the initial steps.

4.2. Experiments

With an accuracy rate of up to 86.8% in detecting and recognizing traffic signs, we have conducted some experimental runs by predicting signs on images and videos to evaluate the model more objectively. Here are some trial results:

Figure 6

Some Experimental Images of Traffic Sign Recognition

Source. Data analysis result of the research

The trial results above show that the model's accuracy in recognizing and detecting traffic signs is very high, at 80% or above. However, there are still cases where images are too small or blurry, leading to incorrect identification and lower accuracy. With such trial results in mind, we can consider applying the traffic sign detection and recognition model in real-world scenarios in the future. However, adjustments will still be needed better to adapt to the traffic conditions in Vietnam.

5. Conclusions and developments

5.1. Conclusions

In our research, we applied the YOLO algorithm to detect and recognize traffic signs automatically. Adopting YOLO has made the research project more effective in modeling traffic sign detection and recognition while improving processing speed and accuracy compared to manual methods used in previous studies.

During the research process, the YOLOv8 version has demonstrated its performance by significantly improving upon previous versions of YOLO. Utilizing YOLOv8, our team conducted the process of traffic sign recognition more quickly and accurately, enhancing traffic safety and promoting the development of automation applications in the transportation field.

We believe this research will contribute to improving traffic safety and advancing the development of automation applications in transportation, thereby enhancing overall road safety and accelerating the growth of automation applications in the transportation sector.

5.2. Developments

A potential direction for developing this research is to focus on optimizing the performance of the YOLOv8 model. This includes tuning the model's hyperparameters, expanding training data, and experimenting with new techniques to improve the accuracy and processing speed of the model.

In addition to traffic sign recognition, the model should also be applied to detect and classify other objects on the road, such as pedestrians, vehicles, or street scenery. This could significantly enhance street surveillance and security systems.

Furthermore, research should integrate the YOLOv8 model with other methods, such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), or Reinforcement Learning (RL) techniques, to improve the performance and accuracy of the system.

Finally, to ensure the stability and effectiveness of the model in real-world scenarios, deployment and testing on actual systems such as autonomous vehicles and traffic surveillance cameras are necessary.

From optimizing the model, expanding its application to enhancing training data, and integrating with other methods, we believe developing and improving the YOLOv8 model will enhance traffic safety and advance automation applications in this field.

References

Amjoud, A. B., & Amrouch, M. (2023). Object detection using deep learning, CNNs, and vision transformers: A review. *IEEE Access*, *11*, 35479-35516.

Flores-Calero, M., Astudillo, C. A., Guevara, D., Maza, J., Lita, B. S., Defaz, B., Ante, J. S., Zabala-Blanco, D., & Moreno, J. M. A. (2024). Traffic sign detection and recognition using YOLO object detection algorithm: A systematic review. *Mathematics, 12*(2), Article 297.

Liang, L., Bao, H., Pan, W., & Pan, F. (2022). Traffic sign detection via improved sparse R-CNN for autonomous vehicles. *Journal of Advanced Transportation, 2022,* Article 3825532.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 779-788).

Saxena, S., Dey, S., Shah, M., & Gupta, S. (2023). Traffic sign detection in unconstrained environment using improved YOLOv4. *Expert Systems with Applications, 238,* Article 121836.

Sugiharto, A., & Harjoko, A. (2016). Traffic sign detection based on HOG and PHOG using binary SVM and k-NN. In *2016 3rd International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)* (pp. 1-6).

Terven, J., Córdova-Esparza, D. M., & Romero-González, J. A. (2023). A comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS. *Machine Learning and Knowledge Extraction, 5*(4), 1680-1716.

Truong, B. Q., Truong, C. H., & Truong, D. Q. (2015). Phát hiện và nhận dạng biển báo giao thông đường bộ sử dụng đặc trưng hog và mạng nơron nhân tạo [Detection and recognition of road traffic signs using hog features and artificial neural networks]. *Tạp chí Khoa học Đại học Cần Thơ*, 47-54.

Yu, J., Ye, X., & Tu, Q. (2022). Traffic sign detection and recognition in multi images using a fusion model with YOLO and VGG network. *IEEE Transactions on Intelligent Transportation Systems, 23*(9), 16632-16642.

Zhu, Z., Liang, D., Zhang, S., Huang, X., Li, B., & Hu, S. (2016). Traffic-sign detection and classification in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2110-2118).