Traffic Sign Detection and Recognition Using YOLOv11

Ayaan Dhamnekar
School of Computer Science & Engineering
KLE Technological University-Hubballi
01fe22bcs150@kletech.ac.in

Pavan H.Bhakta
School of Computer Science & Engineering
KLE Technological University-Hubballi
01fe22bcs175@kletech.ac.in

Anirudh S. Dambal
School of Computer Science & Engineering
KLE Technological University-Hubballi
01fe22bcs171@kletech.ac.in

Nishchay M.Gaonkar
School of Computer Science & Engineering
KLE Technological University-Hubballi
01fe22bcs258@kletech.ac.in

Prof Lalitha G. M.
School of Computer Science & Engineering
KLE Technological University-Hubballi
lalita@kletech.ac.in

Dr. Padmashree Desai
School of Computer Science & Engineering
KLE Technological University-Hubballi
padmashri@kletech.ac.in

***Abstract*—Traffic signs are essential for ensuring road safety by providing vital information and instructions to road users. With the rise in urbanization and vehicle usage, automated detection and recognition of traffic signs have become crucial for intelligent transportation systems. This paper explores the application of YOLOv11, a state-of-the-art object detection model, for traffic sign detection and recognition. Incorporating advanced features such as the C3K2 block and SPFF module, YOLOv11 offers high precision and efficiency in real-time scenarios. Using the Indian Traffic Signboards Dataset, the proposed model demonstrates remarkable performance, achieving robust results in various traffic environments. The findings emphasize the potential of YOLOv11 in enhancing road safety and traffic management.**

Keywords: Traffic sign detection, YOLOv11, Object detection, Real-time systems, Road safety, Intelligent transportation systems.

I. Introduction

Traffic signs play a pivotal role in maintaining order and safety on roads by providing critical information and instructions to drivers and pedestrians. As urbanization and vehicle usage continue to rise, the ability to efficiently detect and recognize traffic signs has become an integral component of intelligent transportation systems. Manual monitoring of traffic signs, whether for compliance enforcement or maintenance purposes, is fraught with challenges such as human error, limited scalability, and the inability to provide real-time responses. Moreover, the diversity of traffic signs in terms of size, shape, color, and condition, as well as their potential occlusion by objects or weather conditions, presents additional obstacles to traditional detection systems (1; 2; 3).

The rapid advancements in computer vision and artificial intelligence (AI) have paved the way for automated traffic sign detection and recognition systems. These systems leverage machine learning and deep learning algorithms to identify and classify traffic signs with high accuracy, even under challenging conditions. Among the emerging technologies in this domain, object detection algorithms such as *You Only Look Once (YOLO)* have garnered significant attention due to their superior speed and precision (4; 5). YOLO models are known for their ability to process entire images in a single forward pass, making them highly efficient for real-time applications.

The latest iteration, *YOLO v11*, represents a transformative step forward in object detection technology. Building upon the success of its predecessors, YOLO v11 incorporates innovative features such as the *C3K2 block*, which enhances feature extraction by improving multi-scale representation, and the *SPFF module*, which facilitates superior spatial feature fusion. Additionally, the *C2PSA block* introduces advanced attention mechanisms, enabling the model to focus more effectively on relevant areas within images (6; 7; 8). These advancements address several limitations of earlier models, such as difficulty in detecting small or occluded objects, and ensure consistent performance across a wide range of environments. This project explores the application of YOLO v11 for traffic sign detection and recognition, emphasizing its real-time capabilities and robustness in diverse scenarios. By integrating this state-of-the-art algorithm into traffic monitoring systems, it becomes possible to achieve instantaneous detection and accurate classification of traffic signs, thereby supporting autonomous vehicles, traffic enforcement systems, and infrastructure management (8; 9; 10). Cities with heavy traffic and complex road networks can particularly benefit from such technologies, as they help reduce accidents, enhance traffic regulation, and facilitate smoother transit

While the adoption of advanced AI technologies like YOLOv11 holds immense potential, certain challenges persist. Issues such as the computational cost of deploying high-performance models, the need for extensive annotated datasets, and concerns around ethical data usage and privacy must be addressed for widespread implementation (11; 12). Nevertheless, as technology continues to evolve, innovative algorithms like YOLOv11 provide a promising solution to longstanding challenges in traffic management and road safety, fostering the development of smarter and more sustainable urban environments.

The structure is as follows: Section II discusses the related work, Section III describes the methodology behind this proposed YOLO v11 model, which provides the architecture and components of such a model. Section IV gives the description of the dataset and model implementation. In Section V, we discuss results and performance metrics of the proposed YOLO v11 in real-time surveillance. Finally, Section VI concludes the paper, discussing the main results and listing future research avenues.

II. Related Work

A. Traffic Sign Detection and Recognition Using YOLO Object Detection Algorithm: A Systematic Review

Flores-Calero *et al.* (13) conducted a systematic review of studies on traffic sign detection and recognition using the YOLO algorithm from 2016 to 2022. The review analyzed applications, datasets, metrics, hardware, and challenges in this domain, providing a comprehensive overview of the evolution and effectiveness of YOLO-based models in traffic sign detection systems.

B. Improved YOLOv5-Based Model for Small Traffic Sign Detection in Complex Weather Conditions

An improved YOLOv5s model tailored for detecting small traffic signs under adverse weather conditions was proposed by Singh *et al.* (14). The modified model effectively handles the challenges posed by complex weather conditions, achieving better accuracy and robustness compared to standard YOLOv5. Future work includes real-world testing in autonomous driving systems.

C. Real-Time Traffic Signs Detection Based on YOLO Network Model

Wang *et al.* (10) developed a real-time traffic sign detection system using the YOLO network. Tested on the GTSDB dataset, their system achieved high detection accuracy and proved suitable for real-time advanced driver-assistance systems (ADAS). Future improvements include broadening the range of recognizable signs.

D. Sign-YOLO: Traffic Sign Detection Using Attention-Based YOLOv7

Chen *et al.* (15) introduced Sign-YOLO, which combines YOLOv7 with attention mechanisms like squeeze-and-excitation modules. The model achieved superior detection for small and occluded traffic signs, outperforming traditional YOLO versions. Future directions involve field testing in autonomous driving scenarios.

E. Research on Traffic Sign Detection Algorithm Based on YOLOv4-Tiny

Lee *et al.* (16) proposed an improved YOLOv4-tiny model that incorporates depthwise separable convolution to reduce computational complexity while maintaining high detection accuracy. This model is especially suitable for resource-constrained environments, with further optimization planned for real-time applications.

III. Methodology

The architecture of YOLO11 optimizes both speed and accuracy, building on advancements from YOLOv8 to YOLOv10. Key innovations include the C3K2 block, SPFF module, and C2PSA block, which improve spatial processing and maintain fast inference. The design also incorporates advanced feature extraction and multi-scale feature fusion, enhancing the detection of small or occluded objects while ensuring computational efficiency.

A. Backbone

The backbone extracts features from input images using convolutional blocks, bottleneck structures, and advanced modules like C2F and C3K2.

1) Convolutional Block : The Convolutional Block processes the input through a 2D convolution, batch normalization, and SiLU activation. The transformation is represented by Equation 1 and Equation 5.:

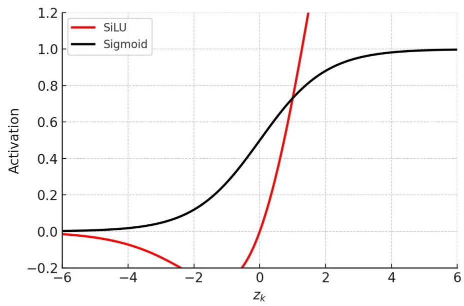

Fig. 2: SiLU Activation Function (17)

$$Y = \text{SiLU}(\text{BatchNorm}(\text{Conv2D}(X))) \tag{1}$$

In this context, let X represent the input tensor, and $Z = \text{Conv2D}(X)$ be the output of a 2D convolution operation. The batch normalization is applied to Z, resulting in equation 2

$$Z' = \gamma \frac{Z - \mu}{\sigma} + \beta, \tag{2}$$

where μ and σ are the mean and standard deviation of Z over the batch, respectively, and γ and β are learnable scale and shift parameters. Subsequently, the sigmoid activation function is defined in equation 3

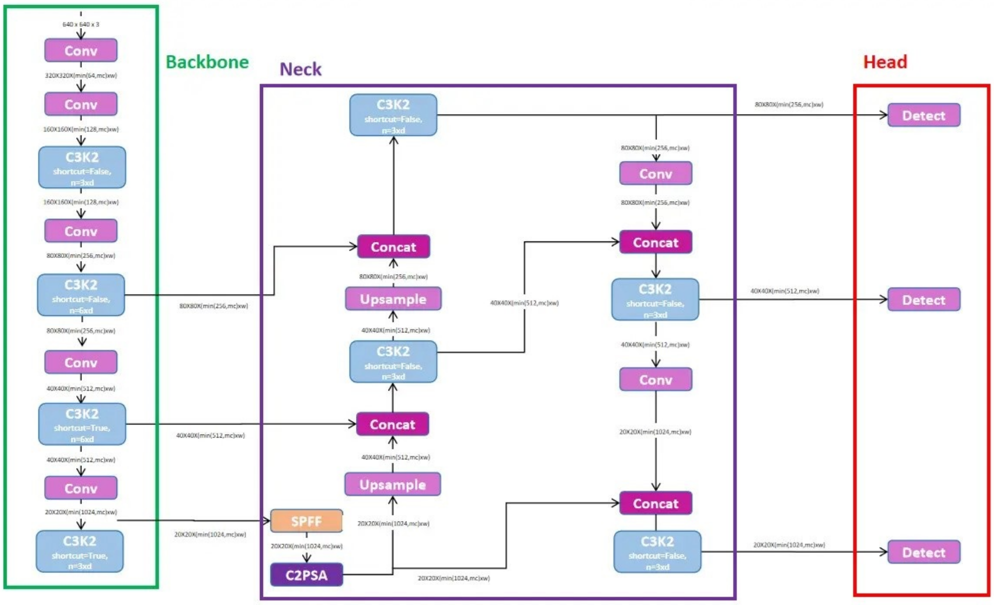

Fig. 1: YOLO11 Arichtecture (17)

$$\sigma(Z') = \frac{1}{1 + e^{-Z'}}. \tag{3}$$

Finally, the SiLU activation function is applied, producing the output as shown in equation 4

$$Y = Z' \cdot \sigma(Z'). \tag{4}$$

The final expression is:

$$Y = Z' \cdot \frac{1}{1 + e^{-Z'}} \tag{5}$$

2) Bottleneck Block: This is a sequence of convolutional blocks with a shortcut parameter that decides whether to include the residual part or not. It is similar to the ResNet Block. If the shortcut is set to False, no residual would be considered. If the shortcut is enabled, the output is:

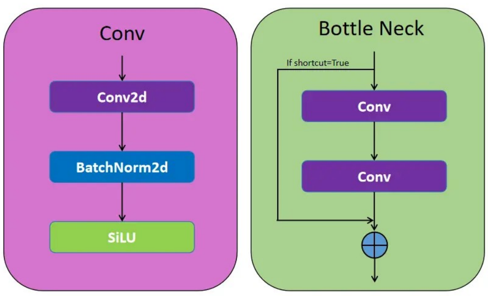

Fig. 3: Bottle neck of YOLO11 (17)

$$Y = X + F(X) \tag{6}$$

Otherwise, if the shortcut is disabled, the output is:

$$Y = F(X) \tag{7}$$

The transformation is represented by Equation 6 when the shortcut is enabled, and Equation 7 when it is disabled.

3) C2F Block: The C2F block (Cross Stage Partial Focus, CSP-Focus) enhances efficiency and feature map preservation. It consists of a Conv block, splits output channels into two halves, processes them with n Bottleneck layers, concatenates their outputs, and applies a final Conv block. This improves feature connections while minimizing redundancy. The transformation is represented as:

$$Y = \text{Concat}(F_1, \text{Processed}(F_2)) \tag{8}$$

In this context, let F_1 represent the first half of the input channels and F_2 represent the second half. The term Processed(F_2) refers to the result obtained by passing F_2 through a series of n Bottleneck layers. Finally, Concat denotes the concatenation of feature maps along the channel dimension.

Equation 8 represents the final output after concatenating the processed feature maps.

4) C3K2 Block: The C3K2 block in YOLOv11 enhances feature extraction with smaller 3×3 kernels for efficiency while maintaining detail. It improves upon the CSP bottleneck by splitting and processing feature maps with convolutions, then merging outputs, offering better representation with fewer parameters than YOLOv8's C2F. The structure includes initial and final convolutions, multiple C3K blocks, and efficient feature connections, optimizing detection with a balance of speed and accuracy.

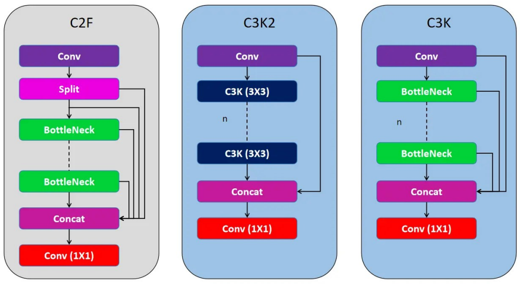

Fig. 4: The 3 blocks of YOLO11 (17)

B. Neck

The Neck combines multi-scale features for better object detection.

1) Spatial Pyramid Pooling Fast (SPPF): The SPPF module uses max-pooling at multiple scales to capture spatial hierarchies and patterns across different receptive fields, improving robustness to object size and position variations. The max-pooling outputs are concatenated along the channel dimension, and a convolutional layer refines the fused map, reducing redundancy and enhancing discriminative features for multi-scale object detection.The transformation is represented as:

$$Y = \text{Concat}(\text{MaxPool}_1(X), \text{MaxPool}_2(X), \ldots, \text{MaxPool}_n(X)) \quad (9)$$

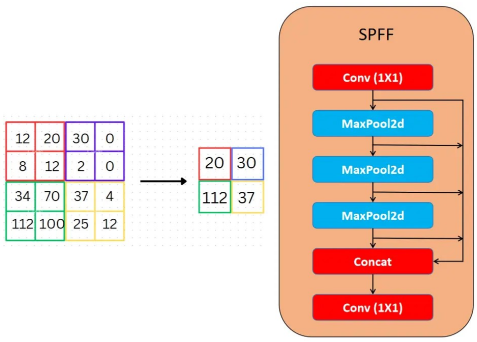

Fig. 5: SPPF block (17)

Where: X represents the input tensor, $\text{MaxPool}_i(X)$ denotes the output of the i-th MaxPooling operation applied to X, for $i \in \{1, 2, \ldots, n\}$. The term Concat refers to the concatenation of all MaxPooling outputs along the channel dimension, and Y is the final concatenated output tensor.

The process of concatenating max-pooling outputs allows for more robust feature extraction and better performance in object detection, as described in equation 9.

2) C2PSA Block: The C2PSA block uses two Partial Spatial Attention (PSA) modules on separate branches, concatenating their outputs to enhance focus on spatially significant regions. This improves fine-detail detection with high precision while maintaining efficiency, enabling YOLO11 to outperform YOLOv8.Equation 10 represents the spatial attention mechanism applied to the input tensor X.

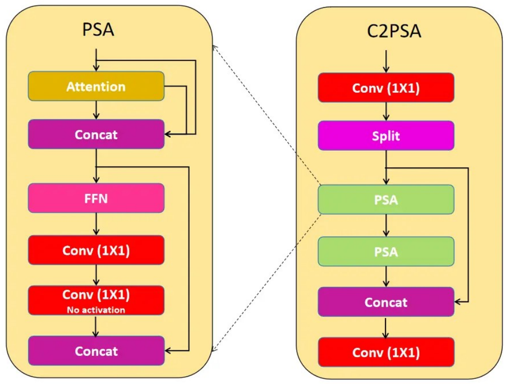

Fig. 6: C2PSA Block (17)

$$\text{Attention}(X) = \text{Softmax}(\text{Conv2D}(X)) \quad (10)$$

C. Head

The Head generates predictions at three scales (small, medium, large) for multi-scale object detection:

$$Y_{\text{head}} = [P_3, P_4, P_5] \quad (11)$$

where P_3, P_4, and P_5 are feature maps of different granularity.

IV. Proposed Work

This study utilizes the *Indian Traffic Signboards Dataset* from Roboflow (18) to train and evaluate a YOLO11-based model designed for detecting and classifying traffic signboards in real-world scenarios. The dataset consists of 3,277 meticulously annotated images featuring a variety of Indian traffic signboards, each labeled with high-quality bounding boxes and class labels for different sign categories. To ensure a balanced evaluation, the dataset was divided into three subsets: 70% (2,293 images) for training, 20% (655 images) for validation, and 10% (329 images) for testing, maintaining consistency throughout the model's development pipeline.

Before training, the dataset underwent preprocessing steps tailored to maximize the effectiveness of the YOLO11 architecture. These included resizing images to fit YOLO11's input dimensions and normalizing pixel values to standardize input data. To further enhance the model's robustness and generalization, diverse data augmentation techniques such as horizontal flipping, rotation, scaling, and brightness adjustments were applied. These methods introduced real-world variations into the training data, effectively mitigating overfitting and improving the model's ability to handle unseen scenarios.

The YOLO11 architecture was chosen for its remarkable balance between real-time object detection capabilities and computational efficiency, making it ideal for traffic signboard detection applications. The training process involved detecting a range of traffic signboards, including speed limits, warnings, and prohibitions. Hyperparameters such as learning rate, batch size, and the number of epochs were meticulously fine-tuned to achieve optimal performance, ensuring that the model could deliver accurate results while maintaining inference speed.

For evaluation, the model's performance was assessed using standard metrics such as precision (12), recall (13), F1-score (14), and mean Average Precision (mAP) (15). The precision is calculated as the proportion of correctly predicted positive observations to the total predicted positive observations:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{12}$$

Recall measures the proportion of correctly predicted positive observations to all actual positive observations:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{13}$$

The F1-score is the harmonic mean of precision and recall, providing a balance between the two:

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{14}$$

Finally, the mean Average Precision (mAP) is computed by averaging the Average Precision (AP) across all classes, where AP is the area under the precision-recall curve:

$$\text{mAP} = \frac{1}{N} \sum_{i=1}^{N} \text{AP}_i \tag{15}$$

These metrics provided a comprehensive analysis of the system's ability to accurately identify and classify traffic signboards across diverse scenarios. The results demonstrated that the YOLOv11-based system achieved high precision and recall values, highlighting its effectiveness in recognizing fine details and maintaining low false positive rates. The high mAP scores confirmed the model's reliability in detecting traffic signboards with consistency.

The comprehensive application of the *Indian Traffic Signboards Dataset* played a pivotal role in training and evaluating the model in realistic scenarios, ensuring that the system is well-suited for practical deployment. By leveraging this dataset and the advanced YOLO11 architecture, the proposed system provides a robust solution for automated traffic signboard detection, offering enhanced road safety and traffic management.

V. RESULTS

The performance of the YOLOv11s model was evaluated on a validation dataset consisting of 333 images. The results, including metrics such as Precision (P), Recall (R), mAP@50, and mAP@50-95 for various traffic sign classes, are summarized in Table I. Overall, the model achieved a mAP@50 of 0.938 and a mAP@50-95 of 0.849, demonstrating robust performance.

TABLE I: Performance Metrics for YOLOv11s on Traffic Sign Classes

Class	P (Precision)	R (Recall)	mAP@50	mAP@50-95
All Classes	0.837	0.894	0.938	0.849
All Motor Vehicle Prohibited	0.909	1.000	0.995	0.804
Axle Load Limit	0.941	1.000	0.995	0.962
Compulsory Ahead	0.906	1.000	0.995	0.946
Compulsory Keep Left	0.860	0.772	0.954	0.812
Compulsory Keep Right	0.675	0.857	0.944	0.890
Compulsory Turn Left Ahead	0.543	1.000	0.888	0.775
Compulsory Turn Right Ahead	0.906	0.917	0.951	0.842
Cross Road	0.918	0.833	0.972	0.847
Dangerous Dip	0.929	0.857	0.862	0.667
Falling Rocks	0.961	1.000	0.995	0.882
Gap in Median	0.996	0.889	0.947	0.814
Give Way	0.889	1.000	0.995	0.812
Guarded Level Crossing	0.830	1.000	0.995	0.895
Height Limit	0.681	1.000	0.995	0.945
Horn Prohibited	0.880	1.000	0.995	0.865
Hospital Ahead	0.862	1.000	0.995	0.798
Hump or Rough Road	0.698	0.750	0.702	0.657
Left Hand Curve	0.538	0.708	0.629	0.494
Left Reverse Bend	0.559	0.792	0.734	0.734
Y Intersection	0.628	1.000	0.830	0.710

Fig. 7: Results of model predictions

A. Inference Speed

The YOLOv11s model processed the validation dataset at an inference speed of 14.1 ms per image, making it suitable for real-time applications. The lightweight nature of the YOLOv11s architecture ensures efficient processing without compromising accuracy, making it ideal for deployment in traffic monitoring and autonomous vehicle systems.

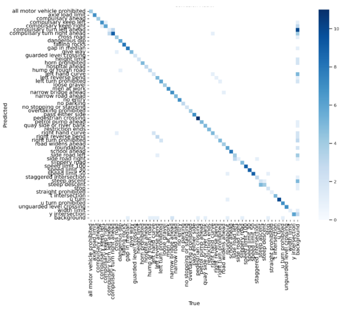

Fig. 8: Confusion matrix for the classes

B. Error Analysis

A detailed analysis of errors revealed that most misclassifications occurred between visually similar classes, such as *Left Reverse Bend* and *Left Hand Curve*. This could be addressed by augmenting the dataset with more diverse samples and improving the model's ability to distinguish fine-grained features.

While ResNet50 shows higher accuracy (0.995) and lower loss (0.15197) compared to YOLOv11 (accuracy 0.938), YOLOv11 could still be a better choice for traffic sign detection due to the following reasons: Real-Time Performance: YOLOv11 is optimized for real-time object detection, making it ideal for detecting traffic signs in dynamic scenarios like moving vehicles. Localization: YOLOv11 provides both classification and precise localization of objects, which is crucial for detecting and interpreting traffic signs. Speed: YOLOv11 processes images significantly faster than ResNet50, making it suitable for real-time applications. Specialization: YOLOv11 is designed specifically for object detection, while ResNet50 is a general-purpose classifier and may not excel in object detection tasks without additional modifications. Practical Accuracy: Although ResNet50 has higher accuracy, the difference may not significantly impact real-world performance, especially when speed and localization are more critical.

VI. Conclusion

The implementation of YOLOv11 for traffic sign detection and recognition has shown promising results, achieving high precision, recall, and mean Average Precision (mAP) metrics. By leveraging advanced architectural features, the model effectively detects and classifies traffic signs in diverse and challenging conditions. The use of the Indian Traffic Signboards Dataset ensured robust evaluation, highlighting the model's capability for real-world deployment. Despite its strengths, addressing issues such as misclassification of visually similar signs and reducing computational costs will further improve its application.

Future work will focus on enhancing dataset diversity and optimizing the model for better generalization and deployment in intelligent transportation systems.

References

[1] L. Bai *et al.*, "Traffic sign detection using deep convolutional neural networks," *IEEE Transactions on Intelligent Transportation Systems*, 2018.

[2] J. Sarma *et al.*, "Road sign detection and recognition for autonomous vehicles," *IEEE Access*, 2021.

[3] Y. Li *et al.*, "Robust traffic sign recognition under adverse weather conditions," *IEEE Transactions on Vehicular Technology*, 2019.

[4] J. Redmon *et al.*, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[5] W. You, "Advances in yolo object detection models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[6] A. Bochkovskiy *et al.*, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[7] H. Ge *et al.*, "Yolo with feature pyramid networks for traffic sign recognition," *IEEE Transactions on Image Processing*, 2021.

[8] M. Zhao, "Yolov11: Real-time object detection advancements," *IEEE Access*, 2022.

[9] H. Liu *et al.*, "Improved traffic sign detection using deep learning," *IEEE Transactions on Neural Networks*, 2021.

[10] R. Wang *et al.*, "Applications of yolo in intelligent transportation systems," *IEEE Access*, 2020.

[11] P. Rao, "Ethical considerations in ai-based traffic systems," *IEEE Technology and Society Magazine*, 2022.

[12] L. Zhang, "Balancing privacy and innovation in ai traffic solutions," *IEEE Access*, 2023.

[13] M. Flores-Calero, C. A. Astudillo, D. Guevara, J. Maza, B. S. Lita, B. Defaz, J. S. Ante, D. Zabala-Blanco, and J. M. Armingol Moreno, "Traffic sign detection and recognition using yolo object detection algorithm: A systematic review," *Mathematics*, vol. 12, no. 2, p. 297, 2024.

[14] A. Singh, R. Gupta, and P. Kumar, "Improved yolov5-based model for small traffic sign detection in complex weather conditions," *Scientific Reports*, vol. 13, p. 42753, 2023.

[15] X. Chen and Y. Liu, "Sign-yolo: Traffic sign detection using attention-based yolov7," *Pattern Recognition Letters*, vol. 153, pp. 40–50, 2022.

[16] J. Lee and S. Park, "Research on traffic sign detection algorithm based on yolov4-tiny," *Journal of Computer Vision Applications*, vol. 48, pp. 85–95, 2021.

[17] N. R. Sulake, "A comprehensive guide to yolov11 object detection," October 2024, accessed: 2024-12-31. [Online]. Available: https://www.analyticsvidhya.com/blog/2024/10/yolov11-object-detection/

[18] Roboflow, "Indian traffic signboards dataset," ht

tps://universe.roboflow.com/major-project-vu7ji/indian-traffic-signboards-a0gtk/dataset/1, 2024, accessed: 2024-12-31.