iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: XII Month of publication: December 2025

DOI: https://doi.org/10.22214/ijraset.2025.76371

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XII Dec 2025- Available at www.ijraset.com

IntelliSign: A Real-Time YOLOv5 based Traffic Sign Detection and Alert System

Harshitha DR[1], Shalini J[2], Sneha V[3], Varshitha N[4], Mrs. Divya S[5], Dr. Vidyarani H J[6]
[1, 2, 3, 4]*Dept. Of Computer Science and Business System, Dr.AIT, Bengaluru, India*
[5]*Assistant Professor, Dept. of Computer Science and Business System, Dr.AIT, Bengaluru, India*
[6]*Professor & Head, Dept. of Computer Science and Business System, Dr.AIT, Bengaluru, India*

Abstract: Traffic signs play a vital role in ensuring safe and efficient road usage by providing drivers with regulatory, warning, and informational guidance. However, in real driving conditions, signs are often missed due to distractions, complex traffic scenarios, unfamiliar routes, or challenging environmental factors such as glare, shadows, and poor lighting. Missing critical traffic signs can increase the likelihood of accidents and reduce situational awareness. With the rapid advancement of intelligent transportation systems, the need for automated, real-time traffic sign detection has become significant. IntelliSign, a real-time traffic sign detection and driver alert system developed using the YOLOv5 deep learning architecture. The system is capable of processing images, prerecorded videos, and to accurately detect and classify traffic signs with high confidence. OpenCV is used for real-time frame extraction and preprocessing, while pyttsx3 provides offline text-to-speech alerts to immediately notify users when a sign is detected. A Streamlit-based interface enables intuitive visualization and user interaction, supporting seamless real-time operation. Experimental results demonstrate that IntelliSign performs reliably across varied lighting conditions, achieving low-latency inference suitable for practical deployment. The system offers a scalable, software-only approach to improve driver awareness and enhance road safety applications. systems.
Index Terms - Computer vision, deep learning, intelligent transportation systems, traffic sign detection, YOLOv5.

I. INTRODUCTION

Road transportation systems rely heavily on the clarity and visibility of traffic signs. These visual cues guide drivers by conveying essential regulatory information, warnings about hazards, and directions needed for safe and organized mobility. When signs are overlooked- whether due to driver distraction, unfamiliar road layouts, high-speed traffic flow, or unfavourable environmental conditions - the risk of accidents increases significantly. With the number of road vehicles growing each year, the burden on drivers to remain continuously aware of these signs has never been greater. The World Health Organization continues to highlight the lack of timely sign recognition as a contributing factor to a substantial portion of road injuries and fatalities worldwide.

Parallel to this increasing need for roadway safety, deep learning and computer vision have matured into powerful tools capable of interpreting real-time visual data with unprecedented accuracy. Over the past decade, a wide range of object detection frameworks has emerged, beginning with handcrafted features and traditional classifiers and evolving into modern deep neural networks that can process complex scenes with high precision. Architectures such as Faster R-CNN, SSD, and the successive YOLO families have transformed real-time perception by offering a combination of speed, accuracy, and adaptability. YOLOv5, in particular, stands out due to its optimized PyTorch implementation, modular design, and ability to detect small and visually distinct objects-qualifying traffic signs as an ideal use case. In this context, automated traffic sign detection systems have become an integral component of intelligent transportation technologies. They can support advanced driver assistance systems (ADAS), enable autonomous navigation pipelines, and serve as standalone applications to enhance driver awareness. However, many existing systems remain tied to expensive hardware, cloud dependencies, or restricted environments where real-time deployment is not feasible. There is an evident gap between high-performing research models and accessible, practical tools that a wide range of users can benefit from.

IntelliSign, a real-time traffic sign detection and driver alert system designed with both practicality and performance in mind. Combining YOLOv5 for detection,

OpenCV for video processing, and Streamlit for visualization, IntelliSign aims to deliver a smooth and responsive user experience while remaining lightweight enough to run on a standard laptop. The inclusion of offline voice alerts through the pyttsx3 library ensures that the system can operate without internet access, making it suitable for varied deployment environments including rural or low-connectivity areas.

II. RELATED WORK

Research on automated traffic sign detection has progressed significantly over the last decade, driven by increased interest in intelligent transportation systems and the rapid development of deep learning. Early studies primarily relied on traditional image-processing techniques such as color thresholding, edge operators, and handcrafted feature descriptors. Although these approaches performed reasonably well in controlled environments, they struggled under variable illumination, cluttered backgrounds, and diverse sign appearances. As real-world deployment requirements became more demanding, these limitations motivated the transition toward data-driven, learning-based models. One of the earliest works evaluating modern object detectors for traffic environments was conducted by Jensen et al. [1], who compared detection algorithms on complex traffic light datasets. Their study highlighted the difficulty of maintaining reliable performance in dynamic scenes, particularly when facing brightness fluctuations and occlusions. This observation helped establish the need for models capable of robust generalization. Complementing this perspective, Santos et al. [2] proposed an assistive traffic-sign recognition framework aimed at enhancing accessibility for visually impaired users. Their work demonstrated the practical value of real-time sign interpretation in driver-support applications. The shift toward deep neural networks brought substantial improvements. Huang et al. [3] explored the integration of camera systems with fuzzy logic to classify signs, marking one of the early attempts to combine heuristic reasoning with machine-learning-based perception. While effective at the time, such hybrid approaches were soon outperformed by convolutional neural network (CNN) architectures capable of automatically learning discriminative features. Bi et al. [4] enhanced VGG-style CNNs for traffic sign recognition, achieving higher inference speed suitable for 5G-assisted vehicular communication. Meanwhile, Karaduman and Eren [5] applied deep-learning models to directional sign detection, emphasizing that CNN-based solutions outperform conventional methods in both precision and consistency. The evolution of general-purpose object detection frameworks further accelerated progress. Redmon et al. introduced YOLO [9], a single-shot detector that significantly improved inference speed, enabling true real-time processing. Subsequent versions, including YOLOv4 [10] and YOLOv5, refined aspects such as backbone architecture, augmentation strategies, and anchor generation. These improvements helped reduce latency while increasing accuracy for small and densely distributed objects, making traffic signs a well-suited application domain. Datasets such as GTSRB, GTSDB, TT100K, and various open-source regional sign datasets have also contributed to advancing research by providing diverse sign categories and environmental variations. Studies evaluating these datasets consistently report that real-world deployment still faces challenges in lighting inconsistency, motion blur, and partial occlusions—factors that motivate further research into robust, real-time systems.

IntelliSign builds upon the strengths of YOLO-based architectures to provide a practical, laptop-friendly system that integrates real-time perception with instant audio alerts. Unlike many prior approaches, IntelliSign emphasizes accessibility, deployability, and minimal hardware requirements, making it suitable for broader adoption in driver-assistive environments and intelligent mobility applications.

III. METHODOLOGY

The methodology adopted for IntelliSign is structured around a streamlined, end-to-end workflow that supports real-time traffic sign detection using general-purpose computing hardware. The development pipeline comprises dataset preparation, model training, frame acquisition, inference, alert generation, and front-end integration as follows:

1) Dataset Preparation: The dataset used for developing IntelliSign is the *Traffic Sign Localization and Detection (SSD-Annotated)* dataset available through Kaggle. This collection contains images of Indian traffic signs along with predefined bounding-box annotations for detection tasks. The dataset includes a mix of regulatory, warning, and informatory signs captured in varied roadside environments, providing suitable diversity for training a robust detection model. All annotated samples were converted into the YOLOv5-compatible format before training. Basic preprocessing involved standardizing the input images and ensuring annotation consistency across the dataset. Since the dataset already contains structured bounding-box labels, no additional manual labeling was required. The prepared dataset was partitioned into training and evaluation subsets using a commonly adopted ratio to allow the model to learn effectively while preserving sufficient samples for unbiased testing. The use of this Indian traffic sign–focused dataset ensures that IntelliSign remains relevant for deployment in local driving scenarios.
2) YOLOv5 Model Training: YOLOv5 was selected due to its efficient PyTorch implementation and support for small-object detection. Training was performed using the YOLOv5s variant, balancing computational cost with accuracy. Hyperparameters such as learning rate, batch size, confidence threshold, and Intersection-over-Union (IoU) threshold were tuned experimentally. The model was trained for 50–100 epochs based on convergence behavior, using stochastic gradient descent with momentum. Techniques like mosaic augmentation, adaptive anchor computation, and label smoothing further enhanced generalization. Model performance was monitored using mean average precision (mAP), precision, and recall metrics across validation subsets.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XII Dec 2025- Available at www.ijraset.com

3) Real-Time Frame Processing and Inference: Inference was built around OpenCV's real-time frame extraction and manipulation capabilities. For video inputs, frames are captured sequentially, resized to match the YOLOv5 input dimension, and normalized before being passed into the model. After prediction, non-maximum suppression (NMS) filters overlapping detections to retain the most confident bounding boxes. The inference process operates within a continuous loop to maintain real-time responsiveness, ensuring minimal delay between frame capture and final detection.
4) Audio Alert Pipeline: One distinguishing feature of IntelliSign is its offline audio alerting system. Using pyttsx3, detected sign labels are converted into speech without relying on cloud services. The alert pipeline is optimized to avoid repeated announcements, each sign triggers a voice output only when newly detected or when transitions occur in the scene. This minimizes auditory redundancy while preserving timely alerts for drivers.
5) User Interface and Voice Alerts: Streamlit served as the interface, enabling image/video upload and real-time detection visualization and voice alerts.

IV. SYSTEM ARCHITECTURE

A. *Input Acquisition Layer*

The system begins with an input acquisition layer capable of handling two modes: static images, prerecorded video files streams. Streamlit serves as the gateway for user-driven file uploads or camera activation.

B. *Preprocessing and Transformation Layer*

Before inference, incoming frames undergo lightweight preprocessing to align with the YOLOv5 input requirements. The preprocessing is optimized to avoid unnecessary computational overhead, allowing faster processing in continuous video streams.

C. *YOLOv5 Detection Engine*

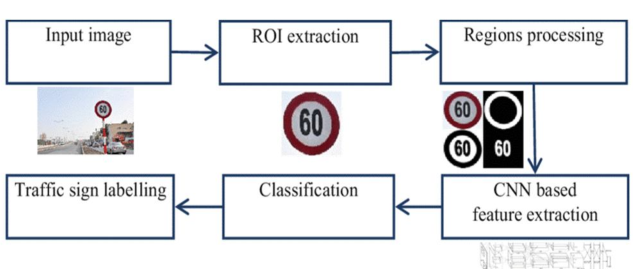
At the core of IntelliSign lies the YOLOv5 detection engine, which performs forward inference on each preprocessed frame. YOLOv5 produces bounding boxes, confidence scores, and predicted sign. The system's accuracy and responsiveness are primarily attributed to the efficiency of this module.

D. *Post-Processing and Decision Layer*

The predictions returned by the detector often include several overlapping suggestions for the same sign. Instead of reporting all of them, the system runs a filtering process that removes redundant bounding boxes. This is achieved through Non-Maximum Suppression (NMS), which looks at overlapping predictions and retains only the one that the model is most confident about. After filtering, the numerical class identifiers are converted into readable sign names. This translation step might appear small, but it is essential because the downstream audio module and on-screen display rely on these human-interpretable labels.

E. *Visualization and Audio Notification*

A Streamlit-based interface displays the output frames with bounding boxes and labels drawn onto them. The interface updates as quickly as new frames arrive, giving users a continuous view of what the detector sees. The design of this alert engine is intentionally simple. It has only one goal: deliver timely warnings without introducing unnecessary noise.



Source: Dung, Hoang Van & Le, My-Ha & Tran, Truc & Pham, Huy. (2018). Improving Traffic Signs Recognition Based Region Proposal and Deep Neural Networks. 10.1007/978-3-319-75420-8_57.

FIGURE 1. SYSTEM ARCHITECTURE.

SHOWS THE DESIGN RIGHT FROM INPUT TO LABELLING.

V. RESULTS AND DISCUSSION

A. Performance on Dataset Images

Using evaluation samples from the dataset, the model detected most traffic signs with strong confidence levels. Signs that have clear geometric shapes and high-contrast colors—such as red-bordered regulatory signs—were usually identified without difficulty. A few samples containing low-resolution signs or backgrounds with similar colors caused minor reductions in confidence, but detections still appeared in most cases. This behavior is consistent with what is expected from a single-stage detector trained on a moderately sized dataset.

B. Real-Time Detection in Video Playback

When the system was tested on prerecorded video clips simulating road movement, it maintained a steady detection rate. The system generally kept pace with the movement in the footage. This demonstrates that the processing loop, which includes frame extraction, model inference, and rendering, works efficiently even when the scene changes rapidly. The model was able to highlight signs as they entered the frame and track them until they were no longer visible.

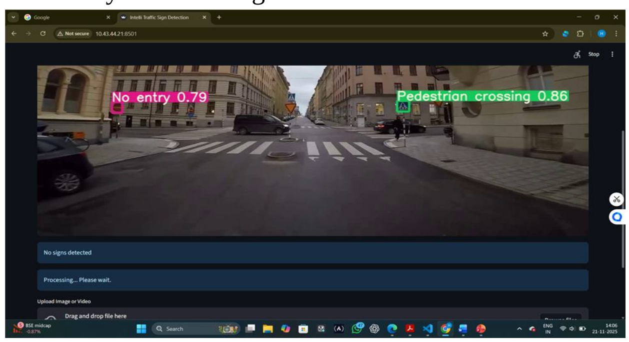


Figure 2. Video detection Detects accurately in a live video feed.

C. Audio Alert Responsiveness

The voice-alert mechanism performed reliably across all modes of testing. The system announced a sign only when it newly appeared, which prevented repetitive messaging and made the alerts more meaningful. There were no delays caused by external dependencies in the system. pyttsx3 used to provide text to speech translation for users and the alerts are real-time.

D. Overall Observations

Taken together, the experiment results suggest that IntelliSign is well suited for real-time awareness tasks. The detection engine, audio module, and interface operate together smoothly, and the system remains stable for extended use. While extreme lighting or very distant signs present challenges, the performance observed during testing demonstrates that the system can reliably support traffic-sign awareness in practical scenarios. Thus, the system provides best user experience and is friendly. It improves road safety and analytics.

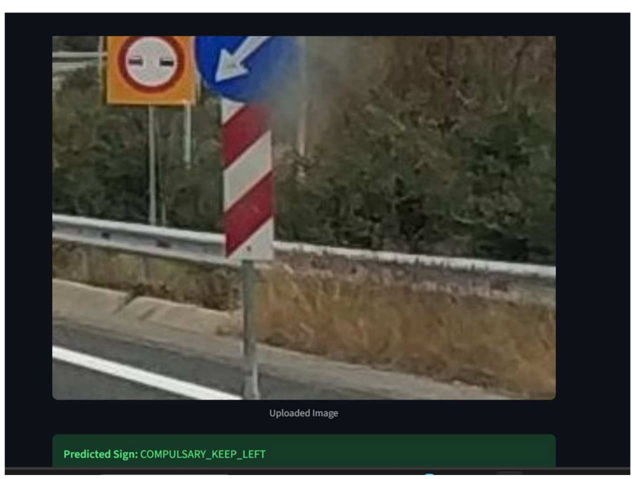


Figure 3. Providing alerts Detects Voice and text based alerts in various environments.

E. Future Enhancements

1) Vision transformers (YOLOv8/RT-DETR)
2) Multi-language alert system
3) GPS tagging of detected signs
4) On-vehicle embedded deployment
5) Low-light enhancement modules

VI. CONCLUSION

The IntelliSign system was developed with the intention of providing a practical, software-based solution for detecting Indian traffic signs in real time. Through the combination of a YOLOv5 detection engine, a lightweight audio alert module, and an accessible Streamlit interface, the system demonstrates how modern computer vision techniques can be applied to everyday driving-awareness scenarios without relying on specialized hardware. The evaluation results indicate that IntelliSign handles common variations in scene conditions and input modes reasonably well. Signs are detected promptly, visual feedback is clear, and the audio notifications support timely understanding of what appears in the environment. While the system performs effectively in standard usage situations, some limitations remain. Scenes with harsh lighting, extreme glare, or very distant signs occasionally reduced confidence or delayed detection. Addressing these cases may require further fine-tuning, such as training with additional samples or applying preprocessing techniques suited to low-contrast environments. Despite these challenges, IntelliSign serves as a strong demonstration of how a compact and easily deployable detection pipeline can contribute to improving driver awareness. It provides a foundation that can be extended with future enhancements such as multi-language alerts, broader sign coverage, or integration with navigation tools.

REFERENCES

[1] M. B. Jensen, K. Nasrollahi and T. B. Moeslund, "Evaluating state-of-the-art object detectors on challenging traffic light data," Proc. IEEE CVPR Workshops, pp. 9–15, 2017.
[2] A. Santos, P. A. R. Abu, C. Oppus and R. S. J. Reyes, "Real-time traffic sign detection and recognition system for assistive driving," Ateneo Research Archive, 2020.
[3] S. Huang, H. Lin and C. Chang, "An in-car camera system for traffic sign detection and recognition," in Proc. IFSA–SCIS, pp. 1–6, 2017.
[4] Z. Bi, L. Yu, H. Gao et al., "Improved VGG model-based efficient traffic sign recognition for safe driving in 5G scenarios," Int. J. Mach. Learn. Cybern., 2020.
[5] M. Karaduman and H. Eren, "Deep learning-based traffic direction sign detection and determining driving style," in Proc. Int. Conf. Computer Science and Engineering (UBMK), pp. 1046–1050, 2017.
[6] World Health Organization, "Road traffic injuries," Fact Sheet, Mar. 2022.
[7] Ultralytics, "YOLOv5: PyTorch implementation,"
[8] Kaggle, "Traffic Sign Localization and Detection (SSD-Annotated) Dataset."
[9] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, real-time object detection," in Proc. IEEE CVPR, pp. 779–788, 2016.
[10] A. Bochkovskiy, C.-Y. Wang and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," arXiv preprint, arXiv:2004.10934, 2020.
[11] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in Proc. IEEE CVPR, 2016
[12] W. Liu et al., "SSD: Single shot multibox detector," in Proc. ECCV, 2016.
[13] M. Everingham et al., "The Pascal Visual Object Classes (VOC) Challenge," Int. J. Comput. Vision, vol. 88, pp. 303–338, 2010.
[14] T.-Y. Lin et al., "Feature Pyramid Networks for object detection," in Proc. IEEE CVPR, 2017
[15] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in Proc. ICML, 2019.
[16] X. Wang et al., "A survey of YOLO-based object detection systems and future trends," arXiv preprint, arXiv:2104.06066, 2021.
[17] "Advances in vision-based intelligent transportation systems," Pattern Recognition Letters, Elsevier, 2020.
[18] "Computer vision techniques for road safety analysis," Machine Vision and Applications, Springer, 2019.
[19] "Real-time visual perception models for intelligent vehicles," in Proc. IEEE Intelligent Transportation Systems Conference, IEEE ITS Society, 2018.