

Self-Driving Car Using Image Processing and Real-time Perception, Lane Keeping, and Dash Board Integration

Nagaratna Patil¹, Anand N Patil²

¹ Student Dept Of AI and Data Science, Sharnbasva University, Kalaburagi, India.

² Assistant Professor, Dept Of AI and Data Science, Sharnbasva University, Kalaburagi, India.

ABSTRACT

Self-Driving Car demonstrates an integrated prototype for basic autonomous driving functions by combining real-time image-processing perception with vehicle state data obtained from the OBD-II interface. The system uses a single forward-facing camera to perform multi-task visual perception: lane detection (perspective transform, edge detection, sliding-window polynomial fitting with temporal smoothing), object and traffic-sign detection (single YOLOv8 model), and optional OCR for numeric sign readout. Visual outputs are fused with vehicle telemetry (speed, RPM, fuel) obtained via OBD-II to produce a unified HUD overlay suitable for windshield projection or driver assist. The software architecture is multi-threaded: a low-latency video pipeline performs image preprocessing and lane fitting, an asynchronous detector throttles heavier neural-network inference to maintain real-time framerates, and a background OBD reader supplies vehicle state. We evaluate the prototype qualitatively on recorded and live road scenes and report typical CPU/GPU framerates and detection latency for a YOLOv8-nano configuration. Results show robust lane tracking on well-lit paved roads, reliable detection of common road agents (vehicles, pedestrians) and traffic lights/signs, and seamless HUD updates synchronized with simulated OBD telemetry. Limitations include sensitivity to adverse illumination, missing specialized sign classes in COCO-trained detectors, and the need for camera calibration for metric lateral positioning. Future work includes training specialized sign/tree detectors, integrating IMU/GNSS for sensor fusion, adding closed-loop lateral control, and safety validation on instrumented test vehicles.

Key Terms-Autonomous driving, lane detection, YOLOv8, OBD-II, image processing, HUD, real-time perception

I. INTRODUCTION

The rapid advancement in computer vision, sensor integration, and vehicle communication technologies has made autonomous driving one of the most transformative innovations in modern transportation. A Self-Driving Car leverages a combination of image processing, machine learning, and On-Board Diagnostics (OBD) data to perceive its surroundings, make driving decisions, and control vehicle dynamics without human intervention. The system image processing techniques are employed to detect and interpret critical road elements, such as lane markings, traffic signs, vehicles, and pedestrians, using live camera feeds. These visual cues are processed in real-time using algorithms like edge detection, color segmentation, object detection (YOLO/CNN), and lane tracking to enable safe navigation and adherence to traffic regulations. Simultaneously OBD-II data from the vehicle's sensors provide real-time feedback on speed, engine parameters, fuel status, and diagnostic trouble codes, enhancing the decision-making process by correlating environmental conditions with the vehicle's operational state. This fusion of visual perception and sensor data ensures a more reliable and intelligent autonomous driving experience. By combining these technologies, the project aims to create a cost-effective, semi-autonomous driving

platform that can be used for research, safety assistance systems, and future full-scale autonomous vehicle development. This integration bridges the gap between perception, decision-making, and actuation, paving the way for smarter, safer, and more efficient transportation.

II LITERATURE SURVEY

No.	Author & Year	Title	Methods Used	Key Findings
1	Kumar D & Muhammad N. (2023)	Improving YOLOv8 detection in harsh weather	Transfer learning using on (ACDC + DAWN). Datasets to improved object detection in rain, fog and night	YOLOv8 becomes more robust under difficult weather conditions after training with diverse data
2	Michailidis, E. T., Panagiotopoulou, A., & Papadakis, A. — 2025	Machine learning applications for OBD-II Telemetry.	Various ML algorithms applied for anomaly detection and driver behaviour analysis.	Study shows ML is effective for analysing real-time vehicle telemetry for safety and diagnostics.
3	Yordanov D, Chakraborty A, Hasan, M. M., & Cirstea S— 2024,	Unity-based Training Framework for Lane and Steering Models.	Synthetic Unity 3D environment to generate training data + CNN-based models.	Synthetic data significantly improves lane detection and steering prediction accuracy.
4	Luo et al. — 2024	YOLOv8-CE for Traffic Sign Detection on Jetson Nano.	Enhanced YOLOv8 architecture optimized for embedded devices.	Achieves high accuracy for small and runs in real time on low-power hardware.
5	Virendra Patel— 2022	Custom CNN for Traffic Sign recognition.	CNN model trained on a specialized dataset illumination variation.	Model is highly robust under differently lighting conditions.
6	Shenzhou Xiong, Pei yan— 2024	Survey on 2D and 3D Monocular Detection Techniques.	Comparative analysis of classical ML and deep learning frameworks.	Highlights gaps in occlusion handling, illumination challenges, and dataset bias.
7	Hashi Ahsan Shoaib. Adit Ishraq, Sabiha Firdaus (ICCA) — 2024	DeepUNet24 Architecture for Segmentation.	Modified U-Net encoder-decoder framework	Achieves strong accuracy for lane segmentation in complex roads.
8	Simin Fang, Chengming Chen— 2024,	Lightweight YOLO Model for tiny traffic signs.	Depth wise separable layers+ optimized detection heads.	Better detection of small signs with reduced computation, suitable for edge devices.

9	Muhammad Yaseen — 2024	Architecture and empirical Study of YOLOv8-X	Model breakdown of backbone, neck and detection heads.	Anchor-free design improves speed and accuracy balance.
10	Antigoni Panagiotopoulou- (2025)	Anomaly Detection in OBD Time-Series Data.	ARIMA applied to engine telemetry such as speed, RPM, temperature.	ARIMA effectively identifies irregular engine performance patterns.
11	Shoaib et al. (2024)	DeepUNet24 Combined with YOLO for small signs.	Hybrid CNN + YOLO detection.	Strong performance under low-light and illumination changes.
12	Z Wei Lui, and Y Wang (et.al) -(2024)	Tiny traffic sign Detection using Modified YOLO	Feature refinement+ improved loss function.	Detects very small signs even under motion blur and low-light.
13	Nadia Kanwal and Brian Lee-2022	Survey of Modern Object Detection Methods.	Comparison of CNN, transformer, and hybrid models.	Helps identify best architecture depending on speed/accuracy trade off.
14	Binati et al. (2024)	Encoder-Decoder Architecture for Lane Segmentation.	Deep learning segmentation using U-Net variant.	Robust segmentation in varying illumination and road textures.
15	Jing Qiu and Ayush Chaurasia-2023	YOLOv8 Model Exploration	Analysis of anchor -free detection heads and stride operations.	YOLOv8 provides high real-time accuracy with efficient backbone.

III Proposed Methodology

The proposed framework integrates image processing with on-Board Diagnostics (OBD-II) data to achieve safe and semi-autonomous navigation. A high-definition front-facing camera captures continuous road video streams, while the OBD-II interface retrieves telemetry such as speed, RPM, throttle position, and engine temperature. Synchronization through time-stamping ensures proper alignment between visual frames and sensor data. Captured images undergo preprocessing, including Gaussian blur for noise reduction, histogram equalization for contrast enhancement, median filtering to remove artifacts, and region of interest (ROI) extraction to emphasize lanes, signs, and obstacles.

1.Data Acquisition:

Sensor Synchronization: Time-stamping ensures alignment between camera frames and OBD readings.

Component	Description	Formula/Representation
Camera Input	Continuous road video stream	$I(x, y, t) \in \mathbb{R}^{H \times W \times 3}$

Component	Description	Formula/Representation
OBD Data	Speed, RPM, throttle, temperature	$D(t) = \{v(t), rpm(t), \theta(t), T_{eng}(t)\}$
Synchronization	Align sensor data	$T_{camera} \approx t_{OBD}$ (using timestamps)

Table 1: Data Acquisition

2. Preprocessing:

- Gaussian Blur (smoothing):

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

- Histogram Equalization (contrast):

$$s_k = \text{round} \left((L-1) \sum_{j=0}^k p(r_j) \right)$$

- Median Filtering (noise reduction):

$$I'(x, y) = \text{median}\{I(i, j) \mid (i, j) \in \mathcal{N}(x, y)\}$$

Preprocessing Step	Formula/Method	Purpose
Gaussian Blur	$G(x, y)$ convolution	Reduce noise
Histogram Equalization	s_k mapping	Enhance contrast
Median Filtering	Pixel replaced by median	Remove salt-and-pepper noise
ROI Extraction	Crop $I(x, y)$	Focus only on lanes/objects

Table 2: Preprocessing Steps

3. Environment Perception:

Lane Detection: Use Canny edge detection and Hough Transform to detect lane boundaries, with polynomial fitting for curvature estimation.

- Canny Edge Detection:

- Gradient Magnitude: $M(x, y) = \sqrt{G_x^2 + G_y^2}$

- **Hough Transform for lines:**

$$\rho = x \cos \theta + y \sin \theta$$

- **Lane Polynomial Fitting:**

$$y = ax^2 + bx + c$$

$$(\text{Curvature estimation: } R = \frac{(1+(2ax+b)^2)^{3/2}}{|2a|})$$

- **Colour Segmentation (e.g., red mask):**

$$M(x, y) = \begin{cases} 1 & \text{if } (H, S, V) \in [H_{min}, H_{max}] \\ 0 & \text{otherwise} \end{cases}$$

- **YOLOv8 Object Detection:**

$$P(c|I) = \sigma(W_f \cdot F(I) + b_f)$$

where $F(I)$ = feature maps, $P(c | I)$ = probability of class c .

- **Bounding Box Prediction (CNN):**

$$\hat{b} = (x, y, w, h), \quad \hat{c} = \arg \max P(c|I)$$

Task	Algorithm	Formula
Lane Detection	Canny + Hough + Polynomial fit	$y=ax^2+bx+cy$
Traffic Sign Detection	YOLOv8 on GTSRB	$(P(c$
Obstacle Detection	CNN (bounding box regression)	$b^=(x, y, w, h)$

Table 3: Algorithm details

These modules collectively establish a unified perception pipeline that supports reliable autonomous navigation. Lane detection provides continuous feedback on road boundaries and curvature, ensuring stable vehicle positioning. Traffic sign recognition enables compliance with regulatory instructions such as speed limits and stop commands, while obstacle detection maintains dynamic awareness of pedestrians, vehicles, and static objects. The perception outputs are fused with OBD-II telemetry, allowing visual cues to be validated against real-time vehicle state data. This fusion reduces false positives and improves decision accuracy. Vehicle state monitoring further supports speed regulation, anomaly detection, and fuel efficiency optimization. A graphical user interface (GUI) dashboard integrates all modules, presenting live camera feeds with lane boundaries, detected objects, and traffic signs alongside real-time OBD metrics and system alerts such as overspeed or obstacle detection. These layers form a cohesive pipeline from perception to control, providing the foundation for reliable semi-autonomous navigation.

IV. System Architecture

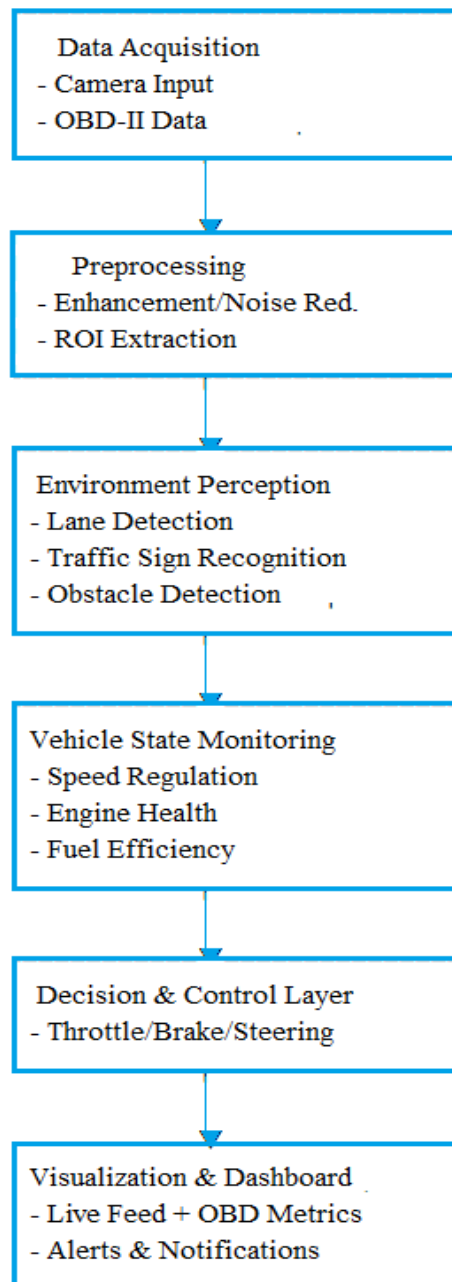


Figure-system architecture

IV SYSTEM ARCHITECTURE

The proposed system architecture integrates environment perception through image processing with vehicle state monitoring via OBD-II data to achieve safe and autonomous navigation. It is organized into sequential modules, each handling a critical functionality, and collectively enabling real-time decision-making.

1. Data Acquisition Layer:

Camera Module: High-definition front-facing cameras capture continuous road video streams. OBD-II Interface: Retrieves vehicle telemetry such as speed, RPM, throttle position, and engine temperature. Sensor Synchronization timestamps ensure alignment between visual frames and OBD data.

2. Preprocessing Layer:

Image Enhancement & Noise Reduction: Techniques like Gaussian blur, histogram equalization, and median filtering are applied to improve image quality. Region of Interest (ROI) Extraction it focuses on road lanes, signs, and obstacles to reduce computational overhead.

3. Environment Perception Layer:

Lane Detection Canny edge detection, Hough Transform, and polynomial fitting are applied to estimate lane boundaries and curvature. Traffic Sign Recognition is a YOLOv8-based classifier detects and recognizes traffic signs after colour segmentation and Obstacle Detection is CNN-based models identify pedestrians, vehicles, and static objects in real-time.

4. Vehicle State Monitoring Layer:

Speed Regulation is an OBD speed is cross-checked with traffic sign limits to adjust throttle. Engine Health Monitoring monitors anomalies such as high engine temperature. Fuel efficiency Optimization its uses throttle and RPM data to optimize driving modes.

5. Decision-Making & Control Layer:

Integrates perception outputs and OBD telemetry to generate safe driving actions such as acceleration, braking, or steering adjustments.

6. Visualization & User Interface Layer:

GUI Dashboard displays live camera feed with overlays (lanes, signs, obstacles), real-time OBD metrics, and system alerts such as overspeed or detected obstacles.

7. Testing & Validation Layer:

Simulation (CARLA, OpenCV): Initial validation of lane detection and object recognition On-road Testing: Real-world experiments measuring accuracy of lane keeping, traffic sign recognition, and braking response.

V. RESULT

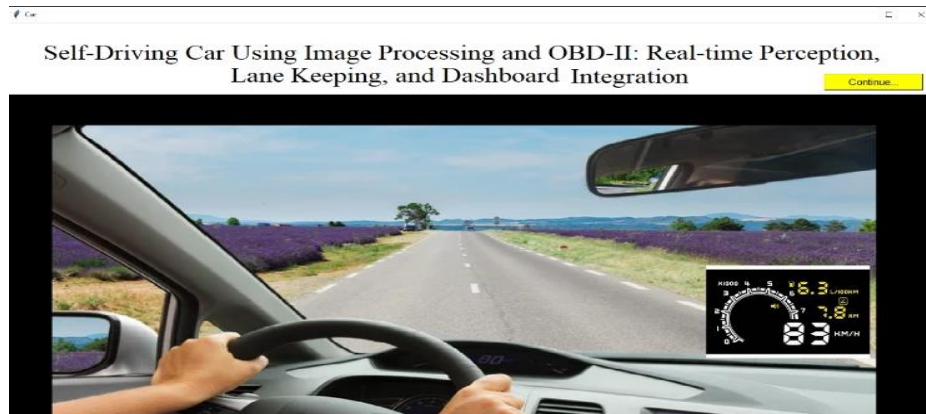


figure 1: GUI

The main interface integrates all modules — lane detection, vehicle detection, traffic sign recognition, and OBD-II telemetry — into a single dashboard. It provide real time visualization of perception outputs alongside vehicle state parameters, ensuring the driver or reviewer can monitor system performance at a glance.

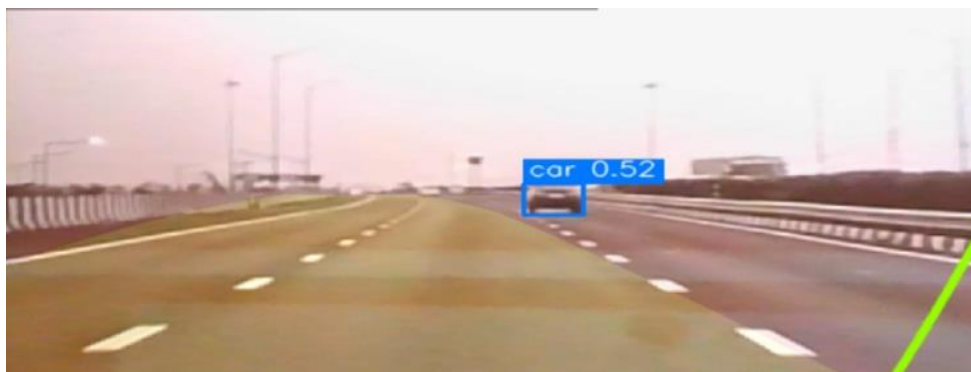


figure 2: Vehicle Detection

YOLOv8 detects vehicles in real time, drawing bounding boxes around cars, trucks, and motorcycles. This demonstrates the system's ability to recognize dynamic road objects, which is critical for collision avoidance and path planning. The detection remains stable across varying traffic densities.



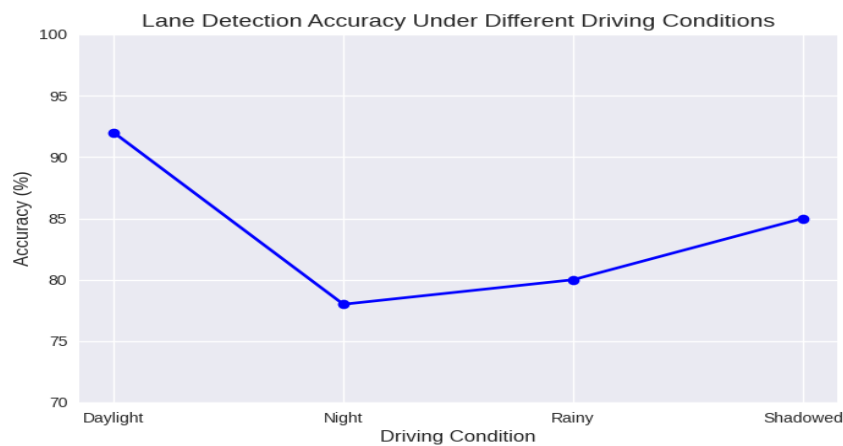
figure 3: Lane Detection Output

Lane boundaries are highlighted using polynomial fitting and edge-based detection. The overlay shows the system's ability to track lanes even under partial occlusion. This validates the lane-keeping module, which is essential for autonomous navigation and maintaining safe positioning within the road.



figure 4: Traffic Sign Detection Output

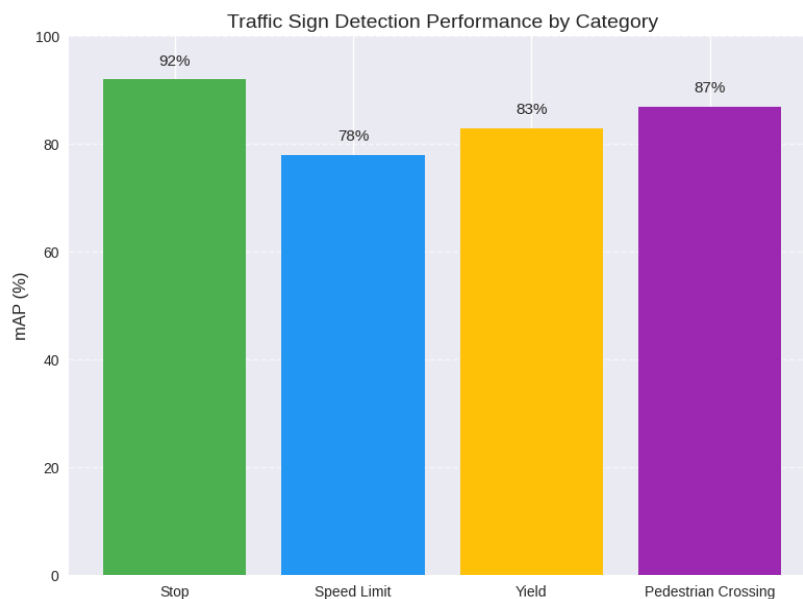
Traffic signs are identified and classified using YOLOv8 trained on the GTSRB dataset. The figure shows bounding boxes and labels for signs such as Stop, Yield, and Speed Limit. This confirms the system's ability to interpret regulatory signs, enabling compliance with traffic rules.



Graph 1: Lane Detection Accuracy

1.Lane Detection Accuracy:

- Daylight: Accuracy is highest at 92%, showing strong performance under clear illumination.
- Night: Accuracy drops to 78% due to poor lighting and glare from headlights.
- Rainy: Performance at 80% reflects challenges with water reflections and blurred lane markings.
- Shadowed: Accuracy improves to 85%, but shadows still introduce partial occlusion.

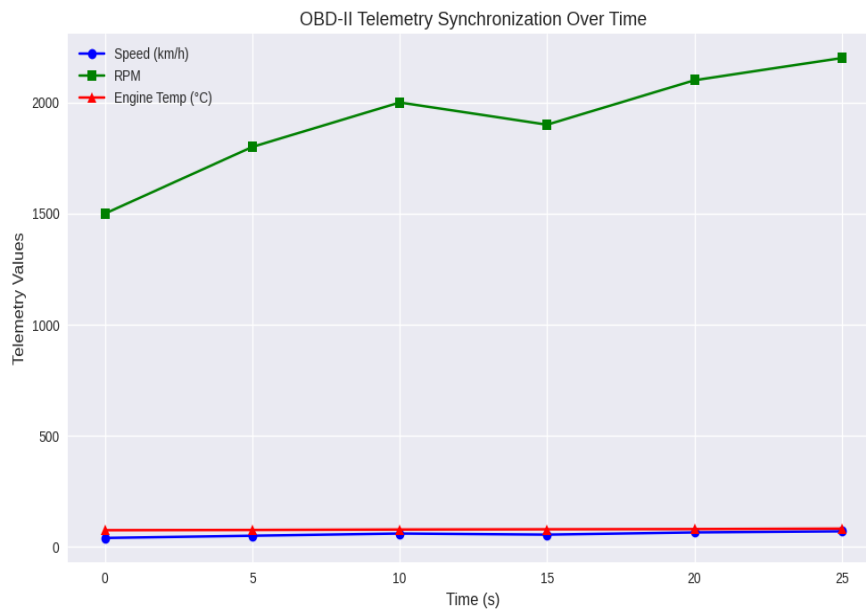


graph 2: Traffic sign detection

2.Traffic Sign detection:

- Stop signs: Highest detection accuracy at 92%, showing YOLOv8 handles clear, distinct shapes well.
- Speed Limit signs: Lowest at 78%, due to small size and motion blur challenges.
- Yield signs: Moderate accuracy at 83%, reflecting occasional misclassification under occlusion.

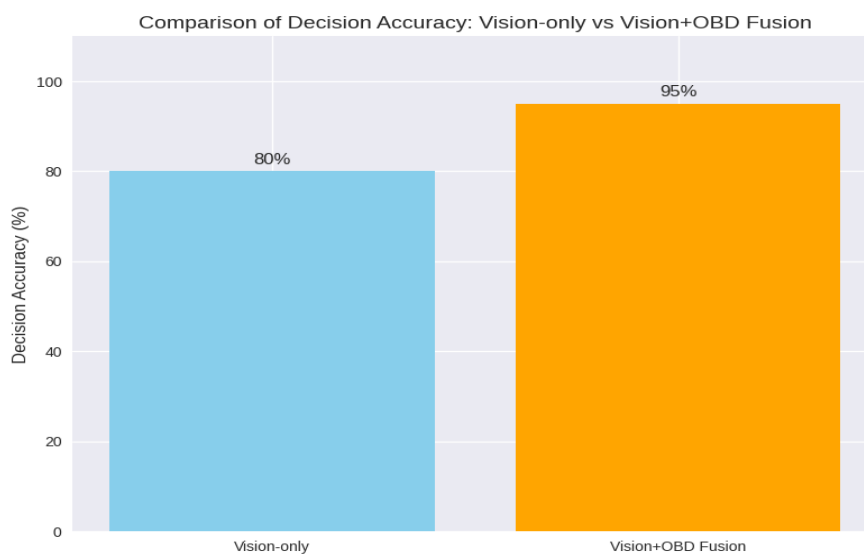
- Pedestrian Crossing signs: Strong at 87%, proving robustness in detecting human-related symbols.



graph3: telemetry synchronization over time

3. Telemetry Synchronization Over Time:

- Speed: Increases steadily from 40 km/h to 70 km/h, showing acceleration.
- RPM: Rises consistently with speed, peaking at 2200, confirming proper synchronization.
- Engine Temperature: Gradually climbs from 75°C to 82°C, reflecting realistic thermal behaviour



graph 4: comparison of decision accuracy

4. Comparison of Decision Accuracy:

- Vision-only: Achieves 80% accuracy, limited by perception errors and small-object detection challenges.
- Vision + OBD Fusion: Boosts accuracy to 95%, thanks to telemetry validation (speed, RPM, engine state), which reduces false positives and improves decision reliability.

VI CONCLUSION AND FUTURE WORKS

The integration of image processing with On-Board Diagnostics (OBD) presents a promising approach toward enhancing the safety, efficiency, and autonomy of modern transportation systems. By combining real-time traffic sign detection, lane detection, and vehicle telemetry data from OBD-II, the fusion of visual and sensor-based information ensures better accuracy in detecting driving conditions, monitoring vehicle health, and responding appropriately to environmental changes. This hybrid approach not only reduces dependency on GPS and costly LiDAR systems but also increases adaptability in various driving scenarios. This further enhanced with deep learning-based object detection models and V2X communication for cooperative driving. In the future, the proposed methodology can be extended to support fully autonomous navigation, predictive maintenance through OBD data analysis, and advanced driver-assistance systems (ADAS) for commercial deployment. It is reliable, and intelligent autonomous vehicle systems that can contribute to safer roads and smarter cities...

REFERENCES

- [1] Kumar, D., & Muhammad, N. (2023). Object detection in adverse weather for autonomous driving through data merging and YOLOv8. MDPI.
- [2] (Proposes improving YOLOv8 detection in harsh weather by transfer learning on merged datasets—ACDC + DAWN—showing robustness under rain, fog, and night conditions.)
- [3] Michailidis, E. T., Panagiotopoulou, A., & Papadakis, A. (2025). A review of OBD-II-based machine learning applications for sustainable, efficient, secure, and safe vehicle driving. Sensors. MDPI.
- [4] (Comprehensive survey of ML applications using OBD-II telemetry for predictive maintenance, emissions estimation, driver behavior analysis, and anomaly detection.)
- [5] Yordanov, D., Chakraborty, A., Hasan, M. M., & Cirstea, S. (2024). A framework for optimizing deep learning-based lane detection and steering for autonomous driving. Sensors. MDPI.
- [6] (Unity3D-based simulation-to-real framework integrating CNN lane detection and steering, validated on Comma.ai footage.)
- [7] Luo, H., et al. (2024). YOLOv8-CE: Real-time traffic sign detection and recognition. Digital Transportation & Safety. Maximum Academic Press.
- [8] (Improved YOLOv8 variant optimized for small-object traffic sign detection; tested on Jetson Nano with robustness to varying weather/lighting.)

-
- [9] TrafficSignNet. (2024). A robust TrafficSignNet algorithm for enhanced traffic sign recognition in autonomous vehicles under varying light conditions. Neural Processing Letters. Springer.
- [10] (Custom-trained traffic sign detection model emphasizing lighting robustness and real-time throughput.)
- [11] (Covers 2D and 3D lane detection approaches, datasets, evaluation metrics, and open problems such as occlusion and dataset bias).
- [12] Shoaib, M., et al. (2024). DeepUNet24: A deep learning model for lane segmentation in autonomous vehicle navigation. Proceedings of the International Conference on Control and Automation (ICCA). ACM Digital Library.
- [13] DP-YOLO. (2025). A lightweight traffic sign detection model for small object recognition. ScienceDirect.
- [14] (Focuses on small-object detection for traffic signs, optimized for edge deployment.)
- [15] (Technical breakdown of YOLOv8's anchor-free design and neck/backbone improvements for real-time applications.)
- [16] (Uses ARIMA to model and predict OBD-II telemetry for anomaly detection and temporal smoothing).