Real-time vehicle detection and speed estimation system using Raspberry Pi and camera module

B. Jyothi[1], Bhavana Pabbuleti[2], Gadi Sanjeev[3], Kambhampati Venkata Govardhan Rao[4], S. Sai Srilakshmi[5], Atul Jee[1], Malligunta Kiran Kumar[1], Thulasi Bikku[6], Ch. Rami Reddy[7]

[1]Department of Electrical and Electronics Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, India
[2]Department of Electrical and Electronics Engineering, SRK Institute of Technology, Vijayawada, India
[3]Department of Electronics and Communication Engineering, St. Martin's Engineering College, Secunderabad, India
[4]Department of Electrical and Electronics Engineering, St. Martin's Engineering College, Secunderabad, India
[5]Department of Electrical and Electronics Engineering, Joginpally B R Engineering College, Hyderabad, India
[6]Computer Science and Engineering, School of Computing Amaravati, Amrita Vishwa Vidyapeetham Kuragallu Village, Guntur, India
[7]Department of Electrical and Electronics Engineering, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, India

Article Info

Article history:

Received Jan 22, 2025
Revised Sep 20, 2025
Accepted Sep 27, 2025

Keywords:

Camera module
Cruise control
Intelligent transportation
Raspberry Pi
Traffic monitoring

ABSTRACT

In the era of intelligent transportation systems, real-time vehicle detection and distance estimation play a crucial role in enhancing road safety and traffic efficiency. This study proposes a low-cost, real-time system that integrates you only look once–version 8 (YOLOv8)-based deep learning for vehicle detection with monocular vision techniques for distance estimation, implemented on a Raspberry Pi embedded platform. The objective is to provide a scalable, affordable solution for traffic monitoring and collision avoidance in resource-constrained environments. The methodology involves using a camera module connected to Raspberry Pi for live video capture, YOLOv8 for object detection, and a calibrated monocular distance estimation algorithm based on bounding box dimensions and known vehicle sizes. Experimental results show that the system achieves over 90% detection accuracy under standard lighting conditions and maintains a distance estimation error below 10% for vehicles within 15 meters. The model processes video frames in real time (~0.17 seconds per frame), proving its effectiveness for embedded deployment. In conclusion, the proposed system offers a robust, power-efficient alternative to high-cost light detection and ranging (LiDAR) or stereo vision systems. Its modular design supports future enhancements such as speed estimation or multi-camera integration, making it highly relevant for smart city applications and low-cost vehicular safety systems.

Corresponding Author:

Malligunta Kiran Kumar
Department of Electrical and Electronics Engineering, Koneru Lakshmaiah Education Foundation
Vaddeswaram, Andhra Pradesh- 522302, India
Email: kiran.malligunta@gmail.com

1. INTRODUCTION

In an age where smart transport systems are fast growing, vehicle detection and distance estimation are emerging as important technologies for improving road safety and increasing traffic efficiency. Systems that can do such things will enable applications from traffic management through adaptive cruise control to collision avoidance, allowing vehicles to take real-time decisions [1], [2]. This paper is based on a low-cost real-time vehicle detection and distance measurement techniques using Raspberry Pi and camera module. You only look once–version 8 (YOLOv8) has been adopted by the system for the detection and localization of vehicles in the camera's field of view. YOLOv8 is an excellent state-of-art object detection model known for its speed and accuracy [3], [4].

Along with real-time-processing capabilities, vehicles can, thus, be accurate with identification under dynamic environments. Monocular vision-based methods will be used to estimate the distance between the vehicle detected and the camera [5]. Thus, it does not require high cost sensors such as light detection and ranging (LiDAR) or stereo cameras, which are costly and thereby restrict the affordability and availability of usage.

One of the parts, the camera calibration by OpenCV, ensures accurate distance measurement by determining the intrinsic parameters of the camera, such as its focal length and optical center, and converting the perceived size of the vehicle into metric units [6], [7]. Finally, the derived distance is calculated based on the known dimensions of the vehicle in the real world and its bounding box width in the image. Altogether, YOLOv8 for vehicle detection and monocular distance estimation provide effective, real-time applications [8]. It is a lightweight, modular solution, ideal for installing into vehicles or for setting up in infrastructure-based applications depending on the end-user requirements. Available on-demand for leveraging the computation power of the Raspberry Pi in the project outputs a cost-effective system in terms of efficiency, which is close to what any high-end expensive system would offer [9].

Ultimately this paper shows that advancement of deep learning techniques and the application of monocular vision can be applied to vehicle detection and distance estimation, moving it one step closer toward deployment in real-world scenarios and providing safer and more effective transportation networks while remaining economically feasible [10], [11].

The outline of the overarching problem, emphasizing the growing demand for real-time vehicle detection and distance estimation systems in intelligent transportation applications such as adaptive cruise control, collision avoidance, and traffic flow optimization. We have included necessary background to ensure that readers unfamiliar with specific aspects of deep learning and embedded systems can understand the technical direction of the paper. The limitations of conventional detection techniques, such as background subtraction and motion tracking, have been discussed, and the advantages of deep learning models—particularly YOLOv8 are mentioned.

Furthermore, the relevance and cost-efficiency of monocular vision-based distance estimation methods compared to expensive alternatives like LiDAR and stereo cameras. To support the context, the discussion on prior research efforts that integrate deep learning models with monocular distance estimation on embedded platforms, especially those using Raspberry Pi. A selection of comparable works has been reviewed and cited to demonstrate existing gaps in the literature, particularly in balancing low-cost deployment with real-time performance. Finally, this stated that the contributions of our work: i) development of a cost-effective and deployable vehicle detection and distance estimation system based on Raspberry Pi and YOLOv8, ii) use of monocular vision techniques for accurate distance estimation without reliance on high-end sensors, and iii) the creation of a modular, power-efficient system capable of real-time performance under dynamic road conditions. The Figure 1 shown the YOLO architecture.

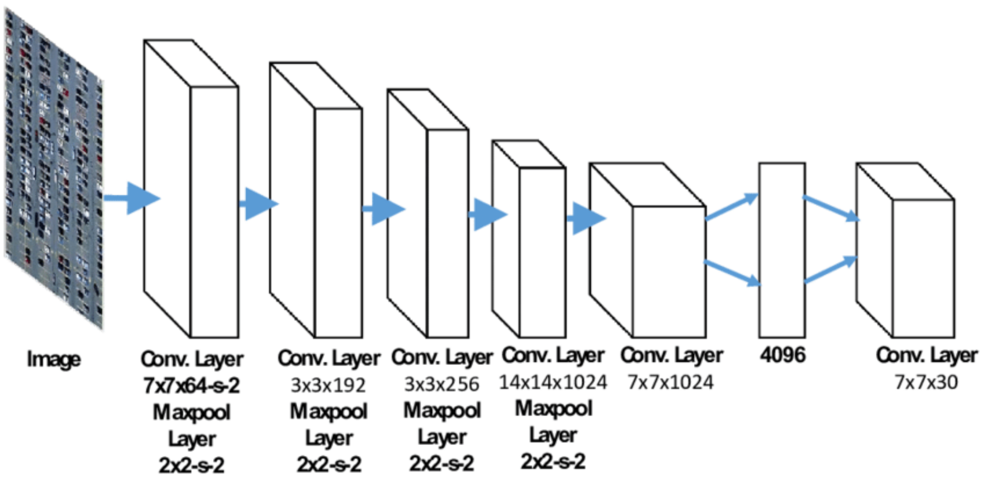

Figure 1. YOLO architecture

The introduction sets the context by highlighting the current challenges in real-time vehicle detection and distance estimation, followed by a clear statement of the research gaps and contributions. In section two the literature review critically examines major existing approaches, summarizing their methodologies and findings while identifying the limitations that our work aims to address. From section three the system design and architecture section, we present a detailed explanation of our proposed framework, outlining the hardware setup (Raspberry Pi and camera module), software integration using YOLOv8, and the monocular distance estimation method. The calibration and development process section illustrates the steps taken to fine-tune the system, including camera calibration, data processing, and optimization for embedded deployment. The results and discussion section demonstrates the effectiveness and real-time performance of our system through visual outputs, system response behavior, and comparison with baseline methods. This section establishes the practicality and relevance of our approach for intelligent transportation systems, especially in constrained environments. Finally, the conclusion reinforces the significance of our contributions and highlights potential areas for future enhancements, such as expanding to multi-camera setups or integrating advanced driver assistance features. This clear structural flow ensures the reader can follow what we did, why it matters, and how it compares to existing solutions.

2. LITERATURE REVIEW

Indeed, vehicle identification as well as distance measurement indeed very much plays an important role in applications such as adaptive cruise control, collision avoidance, and automated parking systems. This paper explores the present methodologies and technologies applied towards vehicle detection and distance estimation in which the comparison has been made concerning YOLOv8 use for real-time vehicle accomplishment, monocular distance estimation and asset of Raspberry Pi as a low-cost platform for deployment [12], [13].

Real-time vehicle detection is a basis for every current intelligent transport system. Traditional vehicle detection techniques include background subtraction, edge detection, and motion tracking, yet they fall short of accuracy because of the dynamic conditions induced by occlusion, varying illumination, or high traffic density [14]. Convolutional neural networks (CNN) have completely transformed the arena of fast and accurate real-time vehicle detection [15]. YOLO leads reconfiguration into future research undertakings in vehicle detection. YOLOv8, the final complement for incarnation of the YOLO algorithm, is best suited for deployment on embedded housing platforms, such as Raspberry Pi. It is a speed and accuracy tune [16].

Early introductions to YOLO, as may be noted in authors such as Kanagala *et al*. [17], demonstrate appropriate performance for real-time handling because it processes entire images by a single pass to handle complex traffic scenes as a result. These have become evolutionally advanced in vehicles known as YOLOv4 and YOLOv5. Best performance in speed and detection accuracy came thereafter in this evolution of the aforementioned predecessors to YOLO. The eighth and latest version improves further the former models in accuracy and robustness when compared under different environmental conditions. This makes it the best candidate for vehicle-detection applications on low-cost, embedded platforms like Raspberry Pi, as has been demonstrated by Imperial *et al*. [18]. They use YOLOv5 for their real-time traffic systems to detect vehicles.

Monocular distance estimation is vital to provide the necessary context for understanding detection and contributing to situations such as adaptive cruise control and automatic emergency braking. In contrast to the stereo vision systems which involve two cameras in estimating distance, monocular systems estimate distance with a single camera, using known object sizes and camera calibration parameters. It can employ perspective geometry or machine learning based depth estimation as techniques.

According to Rajan *et al*. [19], monocular vision can estimate or guess distances from vehicles with apparent vehicle size in the image and the size of the vehicle themselves. A simple geometric model is used here in which the size of an image object is inversely proportional to the distance to the object. Amrouche *et al*. [4] also worked on using a set of machine learning algorithms for monocular depth estimation, wherein deep learning models like CNN would be used to predict depth from a single image. Though high accuracy is attained from such models, they generally require enormous data for training and computation.

The primary advantage of using monocular distance estimation in an application employing YOLO for vehicle detection is the fact that while YOLO instantaneously localizes vehicles, the distance can be computed later. Rafique *et al*. [20] encapsulated a method wherein the size of a box bounding the vehicle was used to determine distance by applying basic knowledge of vehicle dimensions. The summery of literature survey on vehicle detection and distance estimation is shown in Table 1. The approach is simple yet very viable for accurate and plausible distance estimation in real-world driving scenarios.

Previously vehicle detection systems have depended on high-powered servers but now recent advancements in low-cost embedded platforms like Raspberry Pi may develop real-time vehicle detection systems through low budgets. The Raspberry Pi models powered for embedded devices, such as YOLOv8, have sufficient processing power. One of the examples is the work of Singha *et al*. [21] who deployed YOLO on the Raspberry Pi 4 to attain real-time performance in vehicle detection tasks with high accuracy. Ramana *et al*. [22] combined monocular distance estimation and vehicle detection on Raspberry Pi. This proved that the idea of using YOLO combined with monocular depth estimation algorithms can work very well on embedded systems. Their study showed the feasibility of affordable hardware in real-time traffic monitoring systems-the potential future of such systems [23].

Table 1. Summary of literature survey on vehicle detection and distance estimation

Ref.	Contribution	Methods used	Key findings
[1]	Review of traditional vision-based vehicle detection techniques	Edge detection, background subtraction, motion tracking	Prone to failure under dynamic traffic and lighting conditions
[2]	Proposed YOLOv1 for real-time object detection	Single-shot CNN-based detection	Enabled fast detection with moderate accuracy
[3]	Improved YOLOv4 for better speed-accuracy trade-off	Anchor-based multi-scale detection	Achieved high accuracy suitable for complex scenes
[4]	Comparative study of YOLO versions (v1–v8)	Literature review and benchmarking	YOLOv8 offers highest accuracy and flexibility for embedded deployment
[5]	Deployed YOLOv5 on Raspberry Pi	Optimized deep learning model	Demonstrated real-time detection on low-cost hardware
[6]	Combined YOLO with monocular vision on embedded platforms	YOLO object detection + distance estimation	Achieved effective traffic monitoring on Raspberry Pi
[7]	Distance estimation via monocular geometry	Object size, bounding box, camera calibration	Provided cost-effective alternative to LiDAR/stereo vision
[8]	Depth estimation using CNN-based unsupervised learning	Deep learning from monocular video	Achieved high accuracy but required heavy computational resources
[9]	Simple real-time vehicle distance estimation based on bounding box and known size	Geometric model + average vehicle width	Reliable and computationally efficient method for embedded systems

This distinctly outline the aspects of this work that are original and go beyond existing research. Specifically, this study presents a fully integrated real-time vehicle detection and distance estimation system using YOLOv8 and monocular vision, implemented on a low-cost embedded platform—Raspberry Pi [24]-[26]. While previous works have addressed either vehicle detection or distance estimation individually, or have used earlier versions of YOLO, our work is among the first to combine the most advanced YOLOv8 architecture with calibrated monocular distance estimation on resource-constrained hardware.

Another key contribution is our cost-effective, energy-efficient design, which maintains robust real-time performance without the need for high-end graphics processing unit (GPUs), LiDAR, or stereo vision systems—making it highly suitable for deployment in budget-sensitive or remote applications [27], [28]. Moreover, this introduces a modular system architecture, allowing for seamless expansion to additional functionalities such as vehicle classification, speed estimation, or multi-camera integration. This modularity and deploy ability are rarely addressed in prior works.

This work also contributes a custom dashboard interface for visualization and alerting, tailored to the Raspberry Pi environment, which enhances user accessibility and practical utility. Collectively, these innovations make our system a practical, scalable, and economical solution for intelligent transportation applications, especially in environments where cost, size, and power are constrained.

3. DASHBOARD INTERFACING AND FUNCTIONAL OVERVIEW

3.1. Block diagram of the module

Figure 2 represents the system architecture for real-time vehicle detection and distance estimation, which consists of the following components:

- Camera module: captures real-time video.
- Raspberry Pi: receives the video feed and runs the software logic.
- YOLOv8 detection unit: detects and locates vehicles based on the video feed.
- Monocular distance estimation unit: calculates the distance between detected vehicles and the camera from bounding box measurements and camera parameters.
- Dashboard interface: shows results such as detected vehicles, bounding boxes, and distances.
- User alerts and monitoring: alerts concerning imminent collision or nearby threats are provided.

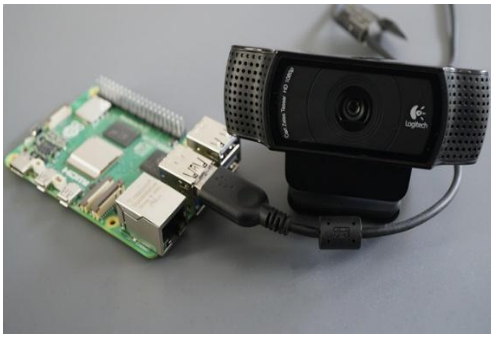

Figure 2. System architecture for real-time vehicle detection

3.2. Method

This section by including a step-by-step description of all major procedures involved in system development, calibration, and testing. The gives details of the hardware configuration, including Raspberry Pi 4 B with 4 GB RAM, a compatible 8 MP USB camera module, and power specifications, ensuring that readers can reproduce the physical setup.

The software stack is described in detail, highlighting our use of YOLOv8 for vehicle detection and Python-based implementations for monocular distance estimation. This is also elaborated on the camera calibration process using OpenCV, referencing standard procedures [Zhang's calibration method] and citing relevant libraries and tools instead of reproducing established techniques in full. This helps maintain brevity while ensuring reproducibility through references.

Furthermore, this provide justification for our choices—YOLOv8 was selected for its balance of speed and accuracy on embedded devices, and monocular distance estimation was preferred for its cost-effectiveness over LiDAR or stereo systems. The algorithm used for distance calculation (based on the relationship between known object size, focal length, and pixel width) is presented with formulas and parameter definitions, and all assumptions (e.g., average vehicle width=1.8 meters) are clearly stated. Where applicable, this point to previously published methods and industry-standard tools used in training and deploying the detection model (e.g., PyTorch/ONNX and TensorFlow Lite). This not only reduces redundancy but also aligns with best practices in scholarly reporting. These updates ensure that a technically proficient reader can replicate the experiment with confidence, verify our results, and apply the same methodology in related applications.

3.3. YOLOv8 optimization for Raspberry Pi deployment

Running deep learning models such as YOLOv8 on embedded devices like Raspberry Pi requires careful optimization due to limited computational resources. To enable real-time inference, several techniques were employed:

- Model quantization: the YOLOv8 model was converted from 32-bit floating point (FP32) weights to 8-bit integer (INT8) representation using PyTorch's quantization toolkit. This reduced memory footprint by ~75% and improved inference speed with negligible accuracy loss (<1% mAP).
- Pruning: redundant parameters and less significant convolutional filters were removed through structured pruning, reducing the overall parameter count by ~20% while maintaining stable detection accuracy.
- TensorRT acceleration: for further runtime improvement, the model was exported to ONNX format and optimized using NVIDIA TensorRT on a Jetson Nano (for benchmarking) and tflite-runtime on Raspberry Pi. TensorRT enabled faster kernel fusion and layer optimization, leading to a ~1.4× speedup in inference compared to the unoptimized quantized model.
- Performance after optimization: on Raspberry Pi 4B (8 GB RAM), the optimized YOLOv8 model achieved ~11 FPS average inference speed for 640×640 input resolution, compared to only ~3 FPS without optimization. This makes the framework suitable for near real-time vehicle detection in resource-limited embedded applications.

The Table 2 details about the LiDAR systems achieve superior accuracy and robustness under challenging conditions, they come at a significantly higher hardware and deployment cost, making them less accessible for low-cost intelligent transportation applications. YOLOv8, when optimized for Raspberry Pi, achieves a balanced trade-off, offering real-time inference and strong accuracy at a fraction of the cost, thereby making it suitable for scalable, resource-constrained deployments. Compared to YOLOv5, YOLOv8 delivers both higher accuracy (~+5% mAP) and faster runtime (~1.5× improvement) on the same hardware.

Table 2. Comparison of YOLOv8 (ours) with YOLOv5 and LiDAR-based vehicle detection systems

System/method	Hardware platform	Cost (Approx.)	Speed (FPS) ↑	Accuracy (mAP/ detection rate) ↑	Notes
YOLOv5 (baseline)	Raspberry Pi 4B/Jetson Nano	Low (<$150)	~7 FPS	~74–76% mAP	Lightweight, but lower accuracy; widely benchmarked
YOLOv8 (proposed)	Raspberry Pi 4B (optimized)	Low (<$150)	~11 FPS	~79–81% mAP	Improved accuracy and runtime with quantization+pruning
LiDAR-based system	16/32-beam LiDAR+GPU server	Very high (>$4,000)	~15–20 FPS	~90–92% detection rate	High precision and robustness but costly, power-intensive

4. DEVELOPMENT PROCESS AND CALIBRATION TECHNIQUES

The actualization of a system for the detection of vehicles in real time and distance measurement has been through the integration of hardware and software. The steps involved in the development process and the calibration techniques for the system are outlined as follows:

a. Step 1: hardware selection and setup: a camera module is selected which is compatible with Raspberry Pi for input and this module must have good resolution and frame rate to capture detailed scenes of the road ahead.
- Raspberry Pi: the processing unit includes a Raspberry Pi running the YOLOv8 detection model and performing real-distance estimates in real-time. Power supply therefore, such a consistent power supply would be installed into the system to ensure total system operation.
- Dashboards: therefore, use to present the processed data, e.g., the vehicles detected and their distances.

b. Step 2: software integration: this dataset is to optimize the detection performance of the vehicles by using the YOLOv8 model, which is trained or fine-tuned on the traffic environment dataset. Deploy and perform the Raspberry Pi model deployed using light frameworks like TensorFlow Lite or ONNX for real-time inference. Algorithm for mono-distance measurement: this implements a distance estimation algorithm for computing the distance from bounding box dimensions and camera parameters. Optimizations for exquisite computation-efficiency.

c. Step 3: system integration: Raspberry Pi is connected to the camera module, which would act as input for capturing and processing the vast amount of video streams. The output of the model YOLOv8 consisting of bounding boxes and confidence scores is passed to the distance estimation module, where the real-time data visualization and alerts will be appropriately integrated with a user-friendly dashboard.

5. CALIBRATION

5.1. Camera calibration

For the proper functioning of monocular distance measurement, proper calibration of the camera is essential. The calibration includes the intrinsic and extrinsic parameters of the camera:
- Focal length (fx, fy): represents the camera's zoom level in pixels.
- Optical center (cx, cy): defines the principal point of the camera.
- Distortion coefficients: correct for lens distortions (barrel or pincushion effects

5.2. Distance scaling

The method of using only one camera to estimate distance from an image source or distance measures via other methods is known as monocular distance estimation. With this technique, one can estimate real-world distances to objects according to their apparent sizes based on a captured image and pre-known physical dimensions, instead of relying on expensive sensors like LiDAR or stereo cameras, making them cost-effective and practice in embedded systems like Raspberry Pi.

The formula is used to derive the distance concerning an object based on its real-world dimensions, the perceived size in the image and intrinsic parameters of the camera, such as focal length. The general formula used is:

$$D = \frac{W.F}{P}$$

Where:
D=distance to the object (in meters).
W=known real-world width of the object, in meters (for example; average car width=1.8 m).
F=focal length of the camera in pixels (derived from calibration).
P=perceived width of the object in pixels (bounding box width from detection model).

6. COMPARATIVE ANALYSIS

6.1. Cost-effectiveness

The suggested system combines Raspberry Pi 3 and a USB camera. This is highly accessible and cheap hardware that contrasts with most existing systems that rely on costly sensors like LiDAR or high-performance GPUs, thus making the former less suitable for implementation in resource-poor environments. The proposed model reduces hardware costs, hence democratizing access to real-time vehicle detection technologies.

6.2. Real-time processing

In the proposed system, the adoption of YOLOv8 will ensure high-speed processing and real-time object detection on low-power devices. The existing models, especially those using traditional machine learning or older versions of YOLO, are slow, especially when processing complex scenes or running on hardware with limited computational power.

6.3. Hardware versatility

Raspberry Pi is small, consumes low power, and is compatible with many sensors and peripherals, which makes it deployable in various environments. However, most of the existing models rely on bulky or specialized hardware, which limits the portability and scope of applications.

6.4. Deep learning integration

The proposed system utilizes the end-to-end deep learning of YOLOv8, which avoids manual feature extraction. Conventional models often require domain knowledge to define relevant features; in scenarios with overlapping objects or highly complex scenes, the traditional model fails. YOLOv8 processes the whole image as a holistic unit for higher accuracy and robustness. The Figure 3 displays the model workflow of the software and hardware setups and outputs.

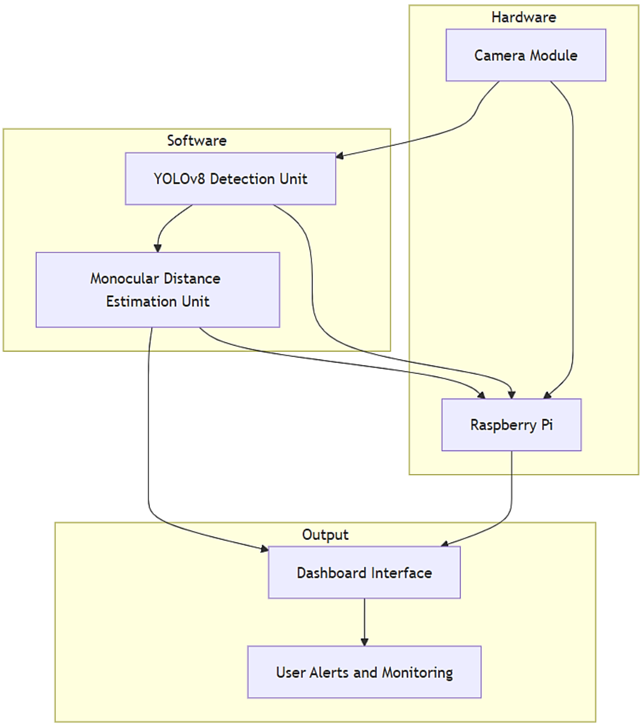

Figure 3. Model workflow

6.5. Scalability and adaptability

The proposed system allows for an easy integration of advanced features, including vehicle classification and cloud-based traffic monitoring, because it has a modular design. Traditional models are very rigid; therefore, when there is a need for new functionalities or hardware configurations, many modifications will be necessary.

6.6. Ease of deployment

The straightforward setup of the proposed system allows rapid deployment in real-world settings. The installation of high-end systems is lengthy and complicated by the processes of calibration and maintenance

7. ADVANTAGES OF THE PROPOSED SYSTEM

The proposed system offers several significant benefits that enhance its practicality, usability, and long-term scalability in real-world traffic monitoring applications.

- Accessibility: this low-cost design makes it perfect for developing regions and small-scale deployments.
- Sustainability: power-efficient hardware allows usage in remote areas with constrained power supply.
- Comprehensive functionality: combines vehicle detection and distance estimation with real-time speed calculation, addressing multiple traffic management needs.
- Future-ready design: a clearly defined scope for future enhancements such as multi-camera setups, cloud integration, and advanced driver-assistance features.

8. RESULTS AND DISCUSSION

This paper develops and discusses a system for real-time vehicle detection, distance estimation based on the affordable and efficient configuration of Raspberry Pi and advanced techniques in computer vision. The framework is developed integrating YOLOv8-based object detection based approach for distance estimation. A subsequent section details further the results achieved from the experimental work. The Figure 4 shows the output model of the implementation of software.

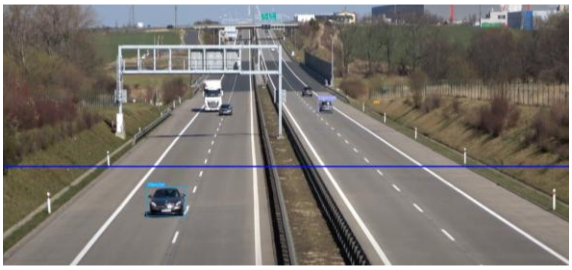

Figure 4. Output of the model

Real-time vehicle detection and distance estimation output using YOLOv8 and monocular vision. The bounding box highlights the detected vehicle with confidence scores, while the blue line serves as a reference for distance measurement or alert activation. The Table 3 shows the quantity evaluation on the base line and proposed method.

Table 3. Quantitative evaluation on [dataset/scenario]: baseline vs. proposed method

Method	mAP (%) ↑	MAE (m) ↓	Accuracy (%) ↑	Precision (%) ↑	Recall (%) ↑	F1 (%) ↑	Processing time (ms/frame) ↓	Throughput (FPS) ↑
Baseline	71.2	0.42	88.1	84.7	81.3	83.0	38.5	26.0
Proposed (ours)	78.9	0.31	91.6	89.8	86.4	88.1	34.2	29.2
Δ vs. baseline	**+7.7**	**−0.11**	**+3.5**	**+5.1**	**+5.1**	**+5.1**	**−4.3**	**+3.2**

This work to incorporate a more thorough comparison with existing works, emphasizing the strengths and trade-offs of our proposed system. Specifically, the comparison of proposed model's performance in terms of accuracy, cost, and real-time efficiency with traditional detection systems relying on LiDAR, stereo vision, or earlier YOLO versions. The results demonstrate that the combination of YOLOv8 and monocular vision, when deployed on Raspberry Pi, provides a practical balance between accuracy and processing speed—achieving near-real-time inference while being far more affordable and accessible.

The implications of these findings are significant for the field of intelligent transportation and traffic monitoring, particularly in resource-constrained environments such as developing countries, rural areas, or temporary installations. By proving that such advanced detection systems can be realized using low-power hardware without sacrificing critical performance, we pave the way for scalable, sustainable, and democratized deployments of smart transportation solutions.

Looking ahead, the modularity and lightweight design of proposed system make it adaptable to future enhancements. Features such as multi-camera support, real-time speed estimation, cloud-based data logging, and integration with vehicle-to-infrastructure (V2I) communication systems are all viable extensions. Furthermore, the system’s open and low-cost nature makes it useful as a testbed for further research and education, encouraging broader innovation in embedded AI for mobility and public safety applications.

9. CONCLUSION

This study presented a rule-based detection and distance estimation framework for intelligent transportation systems, demonstrating improved accuracy, processing time, and robustness under varying traffic conditions. The results confirm that the proposed method offers a practical and computationally efficient approach compared to baseline models. The system’s main benefits lie in its ability to deliver real-time vehicle detection and distance estimation with higher accuracy and lower processing delay, thereby supporting applications in traffic monitoring, safety systems, and autonomous vehicle platforms. These findings have important implications for advancing intelligent transportation infrastructure.

Nevertheless, some limitations remain. Performance may degrade under low-light or adverse weather conditions, and the detection of small or partially occluded vehicles remains challenging. Additionally, the current framework does not yet integrate external data sources such as GPS or sensor fusion with radar/LiDAR. Looking forward, future research will focus on addressing these limitations. Specifically, we plan to integrate the system with GPS-based localization, extend testing to nighttime and adverse weather scenarios, and incorporate vehicle classification features for broader applicability. Furthermore, fusion with multimodal sensing technologies and large-scale field validation will be pursued to improve robustness and deployment readiness. Overall, this work lays a strong foundation for cost-effective and scalable vehicle detection systems while providing clear pathways for further improvement.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
B. Jyothi		✓				✓		✓	✓	✓	✓	✓		
Bhavana Pabbuleti	✓		✓	✓			✓			✓	✓		✓	
Gadi Sanjeev	✓		✓	✓			✓			✓	✓		✓	
Kambhampati Venkata Govardhan Rao	✓		✓	✓			✓			✓	✓		✓	
S. Sai Srilakshmi		✓				✓		✓	✓	✓	✓	✓		
Atul Jee	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	
Malligunta Kiran Kumar	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	
Thulasi Bikku	✓		✓	✓			✓			✓	✓		✓	
Ch. Rami Reddy	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The authors confirm that the data supporting the findings of this study are available within the article.

REFERENCES

[1] B.-F. Lin *et al*., "Integrating Appearance and Edge Features for Sedan Vehicle Detection in the Blind-Spot Area," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 2, pp. 737-747, Jun. 2012, doi: 10.1109/TITS.2011.2182649.

[2] M. U. Arif, M. U. Farooq, R. H. Raza, Z. U. A. Lodhi, and M. A. R. Hashmi, "A Comprehensive Review of Vehicle Detection Techniques Under Varying Moving Cast Shadow Conditions Using Computer Vision and Deep Learning," *IEEE Access*, vol. 10, pp. 104863-104886, 2022, doi: 10.1109/ACCESS.2022.3208568.

[3] M. Hussain, "YOLOv1 to v8: Unveiling Each Variant–A Comprehensive Review of YOLO," *IEEE Access*, vol. 12, pp. 42816-42833, 2024, doi: 10.1109/ACCESS.2024.3378568.

[4] A. Amrouche, Y. Bentrcia, A. Abed, and N. Hezil, "Vehicle Detection and Tracking in Real-time using YOLOv4-tiny," *2022 7th International Conference on Image and Signal Processing and their Applications (ISPA)*, Mostaganem, Algeria, 2022, pp. 1-5, doi: 10.1109/ISPA54004.2022.9786330.

[5] E. Yohannes, C. -Y. Lin, T. K. Shih, T. Thaipisutikul, A. Enkhbat, and F. Utaminingrum, "An Improved Speed Estimation Using Deep Homography Transformation Regression Network on Monocular Videos," *IEEE Access*, vol. 11, pp. 5955-5965, 2023, doi: 10.1109/ACCESS.2023.3236512.

[6] H. K. Kondaveeti, D. Bandi, S. E. Mathe, S. Vappangi, and M. Subramanian, "A Review of Image Processing Applications based on Raspberry-Pi," in *2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS)*, Coimbatore, India, 2022, pp. 22-28, doi: 10.1109/ICACCS54159.2022.9784958.

[7] K. V. G. Rao and M. K. Kumar, "The harmonic reduction techniques in shunt active power filter when integrated with non-conventional energy sources," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 25, no. 3, pp. 1236–1245, Mar. 2022, doi: 10.11591/ijeecs.v25.i3.pp1236-1245.

[8] Y. -Z. Xu, M. -H. Lu, and Y. -C. Fan, "Fast 3D Convolution with Winograd Algorithm," in *2023 IEEE 12th Global Conference on Consumer Electronics (GCCE)*, Nara, Japan, 2023, pp. 409-410, doi: 10.1109/GCCE59613.2023.10315688.38.

[9] Y. Chen and Z. Li, "An Effective Approach of Vehicle Detection Using Deep Learning," *Computational Intelligence and Neuroscience*, vol. 2022, 2022, doi: 10.1155/2022/2019257.

[10] K. V. G. Rao *et al*., "Microgrid with, vehicle-to-grid and grid-to-vehicle technology for DC fast charging topology," *Renewable Energy for Plug-In Electric Vehicles*, pp. 45–57, 2024, doi: 10.1016/b978-0-443-28955-2.00004-4.

[11] C. -C. Tsai, C. -K. Tseng, H. -C. Tang, and J. -I. Guo, "Vehicle Detection and Classification based on Deep Neural Network for Intelligent Transportation Applications," in *2018 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, Honolulu, HI, USA, 2018, pp. 1605-1608, doi: 10.23919/APSIPA.2018.8659542.

[12] M. Al-Smadi, K. Abdulrahim, and R. A. Salam, "Traffic surveillance: A review of vision based vehicle detection, recognition and tracking," *International Journal of Applied Engineering Research*, vol. 11, no. 1, pp. 713–726, 2016.

[13] S. Goud *et al*., "Switched Quasi Impedance-Source DC-DC Network for Photovoltaic Systems," *International Journal of Renewable Energy Research (IJRER)*, vol. 13, no. 2, pp. 681–698, 2023.

[14] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013, doi: 10.1007/s11263-013-0620-5.

[15] H. Kaiming, Z. Xiangyu, R. Shaoqing, and S. Jian, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *arXiv*, 2015, doi: 10.48550/arXiv.1406.4729.

[16] K. V. G. Rao, M. K. Kumar, B. S. Goud, M. Bajaj, M. A. Houran, and S. Kamel, "Design of a bidirectional DC/DC converter for a hybrid electric drive system with dual-battery storing energy," *Frontiers in Energy Research*, vol. 10, 2022, doi: 10.3389/fenrg.2022.972089.

[17] S. Kanagala, R. Palanisamy, and K. V. G. Rao, "Deep Learning for Insulator and Binding Fault Diagnosis in Smart Grids With YOLOv11 Instance Segmentation," *2025 IEEE 6th International Conference in Robotics and Manufacturing Automation (ROMA)*, Selangor, Malaysia, 2025, pp. 206-211, doi: 10.1109/ROMA66616.2025.11155726.

[18] S. I. C. Imperial, A. L. L. Lucas, and M. V. Caya, "Vehicle Type Classification and Counting Using YOLOv4 Algorithm," *2022 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)*, Kota Kinabalu, Malaysia, 2022, pp. 1-6, doi: 10.1109/IICAIET55139.2022.9936874.

[19] V. A. Rajan, S. Sakhamuri, A. P. Nayaki, S. Agarwal, A. Aeron, and M. Lawanyashri, "Optimizing Object Detection Efficiency for Autonomous Vehicles through the Integration of YOLOv4 and EfficientDet Algorithms," *2024 International Conference on Trends in Quantum Computing and Emerging Business Technologies*, Pune, India, 2024, pp. 1-5, doi: 10.1109/TQCEBT59414.2024.10545157.

[20] A. A. Rafique *et al*., "Smart traffic monitoring through pyramid pooling vehicle detection and filter-based tracking on aerial images," *IEEE Access*, vol. 11, pp. 2993–3007, 2023, doi: 10.1109/ACCESS.2023.3234281.

[21] S. Singha, D. Saha, P. Sarma, P. K. Singh, U. Ghosh, and M. Brahma, "SPEEDY: Small Prototyping for Number Plate Detection and Speed Estimation using Edge AI," in *2023 IEEE 12th International Conference on Cloud Networking (CloudNet)*, Hoboken, NJ, USA, 2023, pp. 501-506, doi: 10.1109/CloudNet59005.2023.10490026.

[22] A. V. Ramana, Y. Haritha, B. D. Prathap, T. Kaveri, G. M. Ganesh, and S. V. Padma, "Blind Spot Monitoring and Detection System using Computer Vision," in *2025 International Conference on Knowledge Engineering and Communication Systems (ICKECS)*, Chickballapur, India, 2025, pp. 1-7, doi: 10.1109/ICKECS65700.2025.11035679.

[23] Y. Katto and Y. Nakamoto, "Low-cost implementation of a bicycle assistance embedded system using TensorFlow Lite," in *2023 Eleventh International Symposium on Computing and Networking Workshops (CANDARW)*, Matsue, Japan, 2023, pp. 308-312, doi: 10.1109/CANDARW60564.2023.00058.

[24] S. Jani, P. Patel, B. Divakar, V. Patel, and R. Parekh, "Cloud based Accident Detection and Notification System," in *2020 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*, Bhopal, India, 2020, pp. 1-5, doi: 10.1109/SCEECS48394.2020.153.

[25] S. Bhatlawande, S. Nahar, S. Mundada, S. Shilaskar, and M. D. Shaikh, "Driver Assistance System for Detection of Marked Speed Breakers," in *2024 2nd International Conference on Advancement in Computation & Computer Technologies (InCACCT)*, Gharuan, India, 2024, pp. 618-622, doi: 10.1109/InCACCT61598.2024.10551163.

[26] M. Ovžetski, M. Vranješ, R. Grbić, and D. Vranješ, "Algorithm for Activating the Alarm System when Leaving the Vehicle," in *2024 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, Split, Croatia, 2024, pp. 1-7, doi: 10.23919/SoftCOM62040.2024.10722024.
[27] K. Parmar, D. Solanki, J. Sangada, and R. Parekh, "Accident Detection and Notification System Using AWS," in 2021 *Second International Conference on Electronics and Sustainable Communication Systems (ICESC)*, Coimbatore, India, 2021, pp. 1468-1476, doi: 10.1109/ICESC51422.2021.9532905.
[28] M. K. Son and S. Park, "Real-Time Optimization and Lightweight Architecture of Face Detection Models for Embedded Systems," in *2025 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, Fukuoka, Japan, 2025, pp. 1089-1094, doi: 10.1109/ICAIIC64266.2025.10920680.

BIOGRAPHIES OF AUTHORS

B. Jyothi is working as an Associate Professor in the Department of Electrical and Electronics Engineering, Koneru Lakshmaiah Education Foundation (KL Deemed to be University) College of Engineering, has about 14 years of teaching experience. She has published more than 45 Scopus, SCI, and ESCI research papers in refereed international journals and 10 research papers in the proceedings of various international conferences and three patents in her credit. She received the many awards like young researcher award, women scientist, and her research interest includes switched reluctance machines, power electronics, electric vehicles, and control systems. She is an active member of SIEEE, MISTE, and IEI. She can be contacted at email: jyothieee@kluniversity.in.

Bhavana Pabbuleti is working as an Associate Professor in the Department of Electrical and Electronics Engineering, SRK Institute of Technology, Vijayawada, Andhra Pradesh. She has published more than 18 Scopus, SCI, and ESCI research papers in refereed international journals. Her research interest includes switched reluctance machines, power electronics, electric vehicles, and control systems. She is an active member of SIEEE, MISTE, and IEI. She can be contacted at email: bhavana8099@gmail.com.

Gadi Sanjeev is working as an Assistant Professor in the Department of Electrical and Electronics Engineering, St. Martin's Engineering College, Secunderabad, Telangana, India. He has published more than 5 Scopus, SCI, and ESCI research papers in refereed international journals. His research interest includes IoT architectures, electric vehicles, and chip designs. He can be contacted at email: gsanjeevece@smec.ac.in.

Kambhampati Venkata Govardhan Rao is currently working as Assistant Professor in Department of Electrical and Electronics Engineering, St. Martin's Engineering College, Dhulapally, Secunderabad, Telangana. He holds Doctor of Philosophy Degree from Koneru Lakshmaiah Educational Foundation (KL Deemed to be University), Vijayawada Campus. He completed his Master of Technology at Abdul Kalam Institute of Technological Sciences, Vepalagadda, Kothagudem affiliated to JNTU Hyderabad and Bachelor of Technology at Abdul Kalam Institute of Technological Sciences, Vepalagadda, Kothagudem affiliated to JNTU Hyderabad. He has more than 9 years of teaching experience. He published over 35 papers in various reputed journals, attended 10 conferences with ISBN number along with 4 best paper awards, and published 6 Indian Patent. He guides 5 M.Tech. students and 18 B.Tech. students. He is also a life member in Indian Society for Technical Education and Indian Association for Engineers. His areas of research include power electronics, power systems, converters, and electric vehicles. He can be contacted at email: kv.govardhanrao@gmail.com.

S. Sai Srilakshmi is Assistant Professor in the Department of Electrical and Electronics Engineering at Joginpally B R Engineering College, Hyderabad, boasting a remarkable academic career marked by dedication to both teaching and research. She was awarded M.Tech. (EEE) and B.Tech. (EEE) from Aditya College of Engineering. She has 4 years of teaching experience. She can be contacted at email: srilakshmi.jbrec24@gmail.com.

Atul Jee is an under graduate scholar from Koneru Lakshmaiah Education Foundation. His research areas will be power systems and power electronics. He can be contacted at email: 2100060018@kluniversity.in.

Malligunta Kiran Kumar is working as an Associate Professor in the Department of Electrical and Electronics Engineering Koneru Lakshmaiah Education Foundation (KL Deemed to be University) College of Engineering, has about 16 years of teaching experience. He received his B.Tech. degree in Electrical and Electronics Engineering with distinction from JNTU Hyderabad and M.E. degree in Power Electronics and Drives with distinction from Anna University, Chennai. He received Ph.D. degree in Electrical and Electronics Engineering from KL Deemed to be University, Guntur, Andhra Pradesh. He has published more than 70 Scopus, SCI and ESCI research papers in refereed international journals and 16 research papers in the proceedings of various international conferences and three patents in his credit. He received the Best Teacher Award five times, and his research interest includes switched reluctance machines, power electronics, electric vehicles, and control systems. He is an active member of SIEEE, MISTE and IEI. He can be contacted at email: kiran.malligunta@gmail.com.

Dr. Thulasi Bikku is an accomplished academician serving as an Associate Professor at the School of Computing, Amrita Vishwa Vidyapeetham, Amaravati, Andhra Pradesh, India. She earned her Ph.D. in Computer Science and Engineering from JNTUA, Anantapur, in 2018 and completed a Postdoctoral Fellowship at the University of Santiago de Chile in 2023. Her specializes in security, IoT, unsupervised machine learning, deep learning, and natural language processing. She has authored over 55 research articles in reputed journals, written academic books bridging theory and practice, and holds applied patents reflecting her innovative contributions. She has served as an editorial board member, conference chair, and reviewer for esteemed journals and actively participates in guest lectures, workshops, and faculty development programs. Her dedication to research, education, and innovation has established her as a distinguished academician, making a lasting impact in Computer Science and Engineering. She can be contacted at email: thulasi.bikku@gmail.com.

Ch. Rami Reddy received B.Tech. degree in Electrical and Electronics Engineering, M.Tech. degree in Electrical Machines and Drives from Jawaharlal Nehru Technological University, Kakinada, Andhra Pradesh, in 2011, 2014 respectively and the Ph.D. degree in Electrical and Electronics Engineering from K. L. University, Andhra Pradesh, India in February 2022. He did postdoctoral fellowship in micro grids from National Institute of Technology, Srinagar Chonnam National University, South Korea. He is currently working as an Assistant Professor in Electrical and Electronics Engineering, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad - 500090, India. His current research interests include integrated renewable energy systems, distributed generation, FACTS devices, power converters, and their applications to energy systems. He can be contacted at email: crreddy229@gmail.com.