# Urban Traffic Detection for Autonomous Vehicles

Nishant Milind Karle
*Electronics & Telecommunication Engineering*
*A. P. Shah Institute of Technology,*
Thane, India
nishantkarle3@gmail.com

Mohd. Awaish
*Electronics & Telecommunication Engineering*
*A. P. Shah Institute of Technology,*
Thane, India
mohdawaish.in@gmail.com

Ritesh RamBahadur Singh
*Electronics & Telecommunication Engineering*
*A. P. Shah Institute of Technology,*
Thane, India
ritesh143.rs13@gmail.com

Mahesh Pawaskar
*Computer Science & Engineering*
(AIML)
*A. P. Shah Institute of Technology,*
Thane, India
mahesh.pawaskar@gmail.com

***Abstract*— Autonomous vehicles leverage advanced sensors, artificial intelligence, and automation, enabling self-navigation without human intervention. These vehicles hold the potential to significantly improve road safety, enhance efficiency, and revolutionize transportation systems by reshaping how vehicles perceive, interpret, and respond to their environment.**

**The demand for such vehicles arises from the desire for improved urban planning, decreased parking needs, and flexible public transportation. Automation reduces errors, optimizes traffic flow, and produces favorable economic results.**

**This study underscores the crucial importance of advanced traffic and lane detection in reinforcing the reliability and safety of autonomous vehicles, playing a pivotal role in their ongoing evolution.**

**The proposed system operates in real-time, employing dynamic traffic data to inform decision-making. It integrates inputs from cameras, processing parameters such as lane positions, obstacles, and traffic symbols. A centralized control system, comprising Raspberry Pi and Arduino as master-slave components, employs specialized models for lane, object, and traffic symbol detection. This architecture guarantees continuous real-time decision-making and optimizes resource allocation, promoting a resilient and adaptive autonomous driving paradigm. The comprehensive nature of this approach not only aligns with contemporary transportation requirements but also proactively tackles the challenges anticipated in the future urban mobility landscape.**

***Keywords*— Autonomous, Raspberry Pi, Arduino UNO, Detection model, Control Mechanism, Lanes, Object, Traffic Symbol.***

## I. INTRODUCTION

In the realm of prioritizing road safety, the "Traffic Road Sign Detection Algorithms and Machine Learning (ML) with Python" project emerges as a groundbreaking endeavor. Integrating computer vision, machine learning, and Python programming, the project aims to automate the recognition and detection of traffic road signs. Overlooking these symbols, which significantly contribute to driver awareness and traffic management, is common. The project gains exceptional importance amidst global road safety concerns, where traffic accidents impact millions annually. It addresses human judgment limitations by employing advanced computer vision and machine learning, transforming road signs into data-rich objects analyzed, classified, and acted upon. With multilayered machine learning, the system not only recognizes signs but adapts to various conditions, sign variations, and predicts driver behavior based on observed patterns.

The strategic selection of Python as the primary programming language underscores the project's dedication to versatility, ease of use, and a rich library ecosystem. This choice facilitates seamless integration of computer vision and machine learning techniques, enabling the development of a robust and user-friendly system deployable across platforms.

At its core, the project aims to enhance road safety by automating the recognition and detection of traffic road signs. The system empowers vehicles with heightened awareness and the capability to respond to changing traffic conditions. Whether identifying stop signs, yield signs, or speed limit signs, the transformative potential of this system lies in its ability to augment driver awareness and decision-making.

## II. LITERATURE SURVEY

Prakash et al. proposed system investigates autonomous driving technology, aiming for human-free navigation through adept detection and responsive interaction with environmental stimuli. It addresses challenges in computer vision and machine learning, emphasizing precise object localization while balancing efficiency, precision, and simplicity. State-of-the-art models like R-CNN, Fast R-CNN, Faster R-CNN, and YOLO are explored, showcasing advancements in deep learning for autonomous driving. [1]

Prasad et al. proposed system delves into the rich history and modern advancements of autonomous vehicles, dating back to the 1920s with radio-controlled vehicles. Positioned as a pivotal trajectory in automation, it thoroughly examines diverse facets of autonomous driving, emphasizing software stack intricacies like ROS, ML, DL, and OpenCV frameworks. Hardware components, including sensors and cameras, are explored for their vital role in path tracking, computer vision-based control, object avoidance, and real-time object differentiation through filters, highlighting their significance in car control. [2]

Gautam et al. proposed system aims to enhance autonomous vehicle safety through an Automatic Traffic Light Detection System (ALTDS). Tailored for precise traffic light detection in both autonomous vehicles and Driver Assistance Systems (DAS), this vision-based system employs a camera as the primary sensor, eliminating the need for additional configurations. It emphasizes the importance of acquiring real-time images in dynamic traffic scenarios and explores advanced techniques within ALTDS, addressing hardware and software limitations, legal considerations, and simulation environments to refine traffic light detection capabilities. [3]

R. Raffik et al. proposed system investigates advancements in autonomous vehicle technologies, emphasizing communication enhancements for extended operational capabilities. It advocates for resilient connectivity frameworks utilizing relay devices like Raspberry Pi to amplify connectivity and streamline real-time data transmission. By employing sophisticated network configurations, the relay chain methodology ensures secure and untraceable data transmission, particularly in military surveillance applications. [4]

K. B. Swain et al. proposed system, an Integrated Autonomous Vehicle (IAV), is designed for covert navigation within secure zones to avoid enemy detection. It serves various purposes, such as detecting and disposing of suspicious explosives and identifying threats in war zones or border security areas. Equipped with sensors and a Raspberry Pi for control, the vehicle's functions are programmed in LabVIEW for efficient data interaction. [5]

Bhaskar Barua et al. proposed system involves the development of a prototype for evaluating Self-Driving Car systems in a simulation environment. It integrates camera-based navigation systems to improve performance. The simulation features a 3D virtual city mimicking real-world conditions, including traffic cars, signals, and obstacles. The goal is to ensure seamless navigation, compliance with traffic regulations, and obstacle avoidance, with a focus on optimizing speed, efficiency, and comfort to minimize fuel consumption and reduce perceptible jerks during the journey. [6]

D. Vitas et al. proposed system introduces a vision-only approach for autonomous driving and driver-assistance systems. It focuses on traffic light recognition using adaptive thresholding and deep learning for region proposal and localization, respectively. Leveraging the LISA dataset with custom augmentation techniques, it achieves a true detection rate of 89.60% in classification and accurately locates traffic lights in 92.67% of cases through regression. [7]

Guo Mu et al. proposed system addresses the critical need for robust traffic light detection and recognition in urban autonomous driving scenarios. Specifically designed for autonomous vehicles, the camera-based algorithm offers real-time performance. Unlike existing methods primarily designed for fixed position detection, this algorithm tackles real-world conditions. It employs RGB to HSV conversion for preprocessing, transcending color thresholding for initial filtering, and HOG features with SVM for recognition. Evaluation on our autonomous vehicle demonstrates sufficient accuracy for urban environments. [8]

A. A. Assidiq et al. proposed system addresses the critical need for enhanced safety and reduced road accidents through Advanced Driver Assistance Systems. Lane detection, a complex task for future vehicles, forms a pivotal aspect. Utilizing vision systems, our approach offers real-time lane detection robust to lighting changes and shadows. By employing hyperbolas and Hough transform, our system demonstrates robustness across various road conditions and weather scenarios, meeting real-time operational requirements effectively. [9]

J. E. Hoffmann et al. proposed system addresses the challenges posed by the deepening of neural networks, which impacts processing times in tasks like object classification and localization. Our multistage algorithm integrates YOLOv3 for object detection and HART for tracking, followed by an adaptive post-localization stage shift system. Evaluation showcases superior performance, achieving a 49% frame rate gain over YOLOv3 while maintaining competitive accuracy. [10]

## III. PROPOSED SYSTEM

The block diagram is partitioned into four distinct segments, each comprising both hardware and software components, delineated in a sequential left-to-right progression.

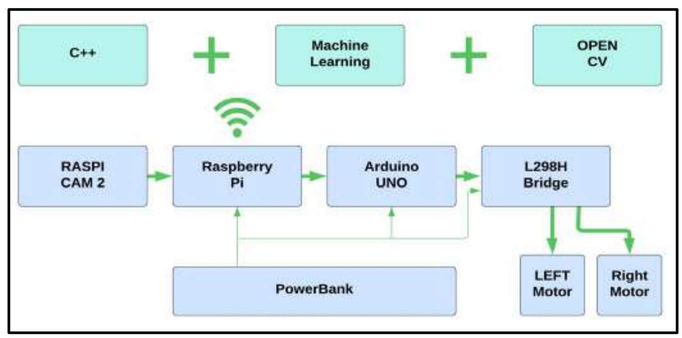


Fig. 1. Hardware System Block Diagram with Essential Software Components.

### A. *Sensor Data Acquisition:*

The Raspberry Pi Camera captures and systematically stores data in a designated directory as images. The image-capturing process, executed with a 0.1-second delay, ensures consistent and precise sensor data collection.

### B. *Detection Mechanisms:*

The obtained images undergo processing via three unique detection models: Lane Detection, Object Detection, and Traffic Symbol Detection. Each model produces individual outputs, collectively contributing to autonomous vehicle control. These outputs act as crucial input signals, shaping the decision-making system that governs and regulates the vehicle's behavior.

### C. *Vehicle Control Mechanism:*

The vehicle's control functions integrate outputs from detection models to ensure operational compliance with predefined conditions. This process fosters efficient coordination between detection models and the vehicle's parameters, enabling responsive control aligned with detected environmental conditions.

## IV. Technological stack

Utilized Hardware Modules:

1. Arduino UNO
2. Raspberry Pi 3B+
3. Raspi Cam 5MP
4. L298H Bridge Motor Driver
5. Battery-Operated Motors
6. 3S Li-ion Battery

Utilized Programming Languages, Libraries, and Tools:

1. Arduino IDE - Employing Embedded C
2. PyCharm – Utilizing Python
3. Ultralytics, cv2, matplotlib.pyplot, Keras

The following are the key reasons for selecting Raspberry Pi 3B+ and Arduino UNO as a master slave device setup:

**Synergetic Hardware Specialization:** The deliberate choice of Raspberry Pi 3B+ and Arduino in a master-slave serial setup capitalizes on their respective strengths. Raspberry Pi handles high-level computations and coordination, while Arduino manages real-time control tasks like sensor interfacing, optimizing system efficiency.

**Cost-Effective Integration:** Combining Raspberry Pi and Arduino optimally balances computational power and affordability. This integration enhances economic feasibility, crucial for scalable implementation of autonomous vehicle systems, leveraging Arduino's cost-effective, real-time capabilities while maintaining computational prowess.

**Community-Backed Reliability:** Raspberry Pi and Arduino, supported by robust communities, ensure project reliability and sustainability. Extensive online repositories provide knowledge, forums, and libraries, simplifying development and maintenance. Leveraging community support enhances the resilience and adaptability of the autonomous vehicle system.

**Agile Prototyping and Customization:** Selecting Raspberry Pi and Arduino for equipment foundation emphasizes spry prototyping and customization. With natural interfacing, they enable quick advancement and iterative testing. Their compatibility with a wide extend of sensors, actuators, and accessories quickens prototyping, easily adjusting to changing extend needs.

## V. METHODOLOGY

The procedural sequence of the implementation steps is as delineated below*:*

### *A. Input Data Acquisition:*

The framework utilizes the Raspberry Pi Camera for image capture, storing them in a designated "images" directory. These images constitute the primary dataset for extracting crucial lane position, object detection, and traffic symbol identification information. This step initiates the data acquisition process, essential for subsequent analysis and decision-making mechanisms within the autonomous vehicle system.

### *B. Lane Detection Model*

The lane detection model employs a systematic pipeline for robust lane identification in urban traffic scenarios. The process initiates with image ingestion and conversion to RGB format, followed by extraction of critical parameters such as height and width. A region of interest is then defined to concentrate on relevant lane detection areas. Subsequent steps include grayscale conversion for compatibility with the Canny Edge Detector, application of the Canny Edge Detector, and masking the image based on the predetermined region of interest. Feature extraction employs the Hough transform, utilizing the Hough Lines function to identify lines corresponding to lanes. Detected lines are categorized into left and right fit lines based on their slopes. The average of these fit lines is computed, contributing to the determination of the centerline, prominently displayed in the output image. This systematic process ensures precise lane detection in dynamic urban environments.

### *C. Object Detection Model*

The Python script employs the Ultralytics YOLO (You Only Look Once) object detection framework, utilizing the `YOLO` class, and integrates OpenCV and CVZone. Capturing frames from a video stream, the script applies the pre-trained YOLO model (`yolov8l.pt`). Each frame undergoes iterative processing, with the YOLO model detecting objects and producing a consolidated output with detected class and confidence level. Visual annotation includes overlaying bounding boxes and displaying class names and confidence levels. The script, utilizing a predefined list of class names, demonstrates real-time YOLO-based object detection with visual annotations.

### *D. Traffic Symbol Detection Model*

The Python script seamlessly integrates computer vision and machine learning to enable real-time detection of traffic signs in video streams, leveraging a pre-trained model. Using a Raspberry Pi Camera, OpenCV, and TensorFlow's Keras module, the script loads the pre-trained model ('traffif_sign_model.h5'). Image preprocessing involves converting frames to the HSV color space, applying color thresholding, and optional erosion and dilation operations. Contours are then detected, filtered, and potential traffic signs' bounding boxes obtained. The script classifies these regions using the loaded model, annotating the original video stream with predicted class names and probabilities above a set threshold (0.75). Dependencies include NumPy, OpenCV, and TensorFlow's Keras module. The script outputs a displayed video stream enriched with annotated traffic sign classifications, serving as a valuable tool for traffic monitoring and autonomous vehicle systems.

### *E. Control Mechanism*

The lane detection, object detection, and traffic symbol detection model outputs are unified into a singular input for the Arduino UNO. This consolidation, integral to the subsequent serial communication, reduces computational load on the slave device. The Arduino UNO meticulously assesses the processed input from the Raspberry Pi, employing predefined boundary conditions tailored to specific commands (e.g., Forward, Left soft, Right soft, Left extreme, Right extreme, Stop). This meticulous approach guarantees accurate vehicle control in response to dynamic traffic conditions, optimizing the overall control mechanism.

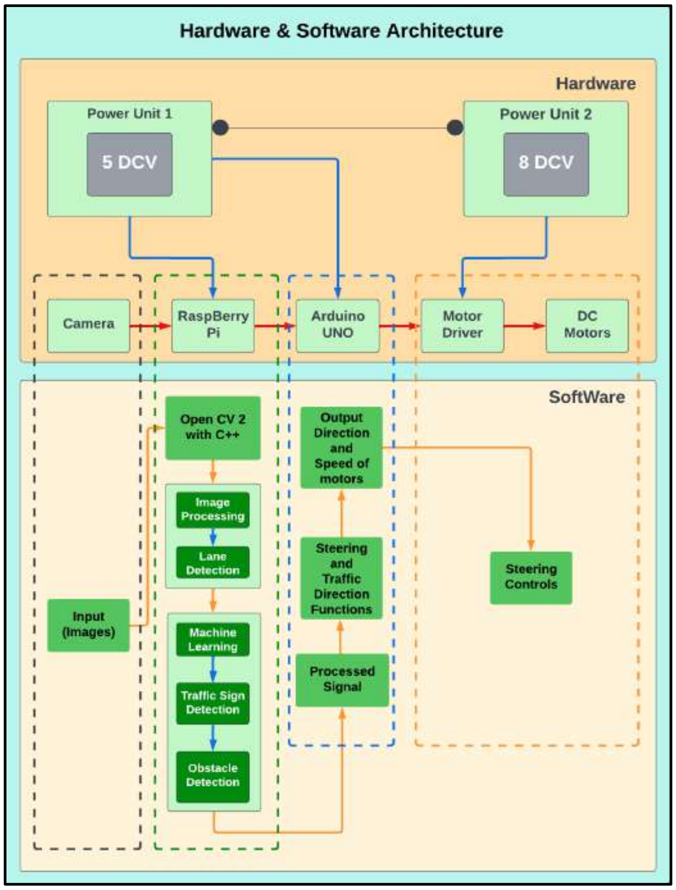


Fig. 2. Representation of the workflow encompassing processes ranging from image acquisition to steering control, with their respective hardware and software components.

## VI. RESULTS AND DISCUSSION

A prototype vehicle was built to implement the work. All the detection algorithms were simulated before implementing them on the prototype. The results for the same are shown below.

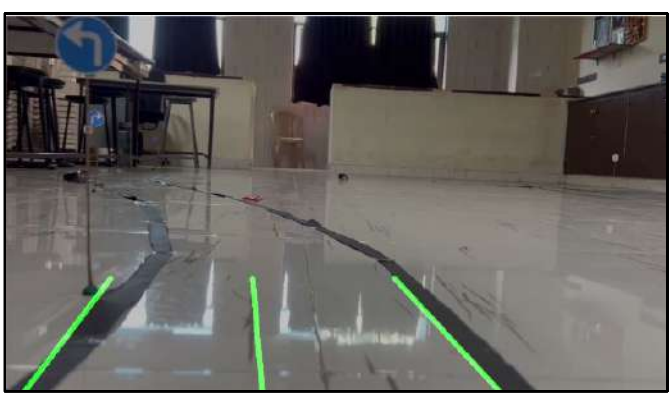

Fig. 3. Simulation output 1 generated by the lane detection model.

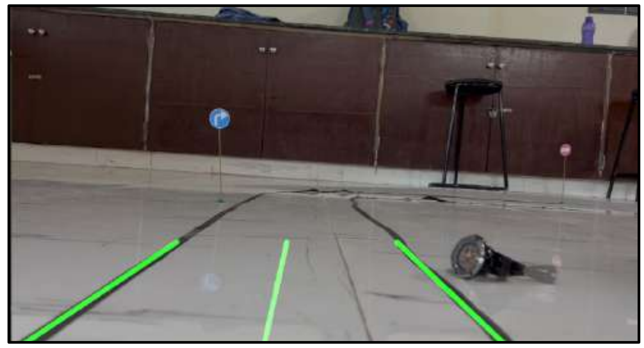

Fig. 4. Simulation output 2 generated by the lane detection model.

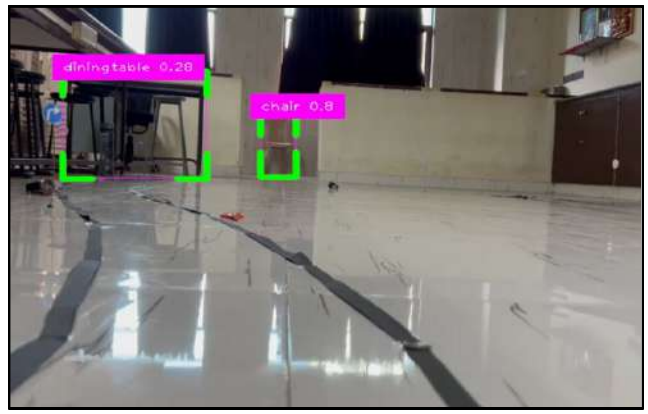


Fig. 5. Simulation output 1 generated by the object detection model.

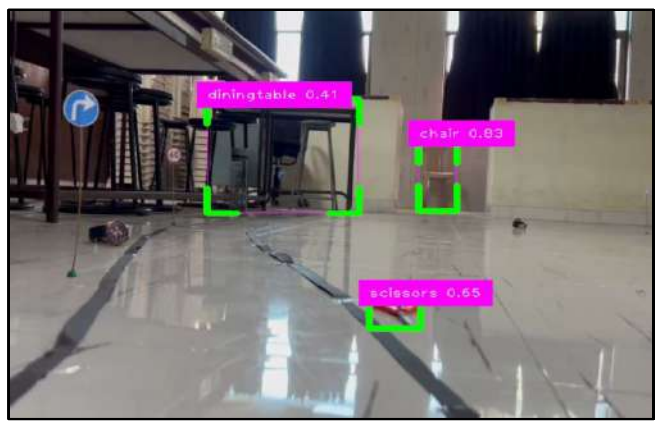


Fig. 6. Simulation output 2 generated by the object detection model.

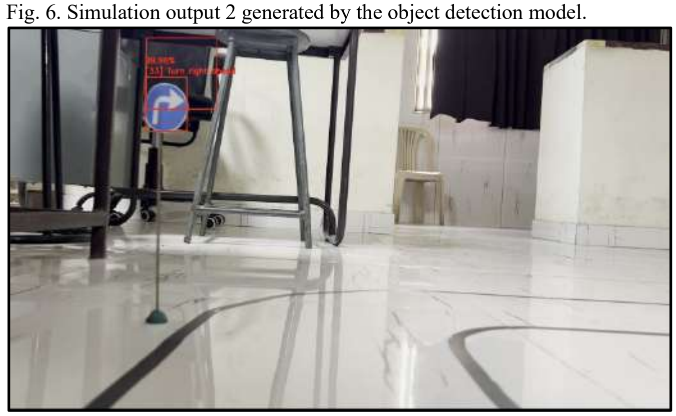

Fig. 7. Simulation output 1 generated by traffic symbol detection model

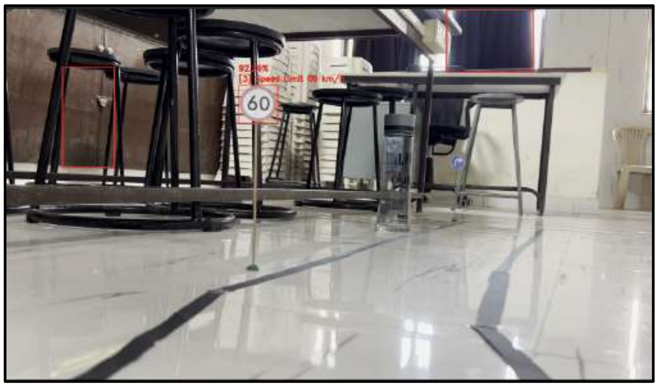

Fig. 8. Simulation output 2 generated by traffic symbol detection model

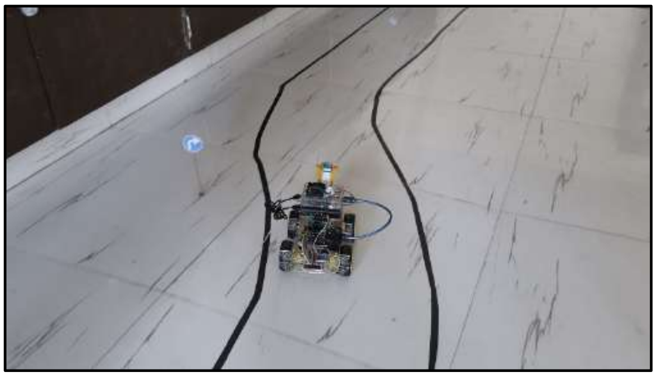
Fig. 9. Operation of the prototype vehicle

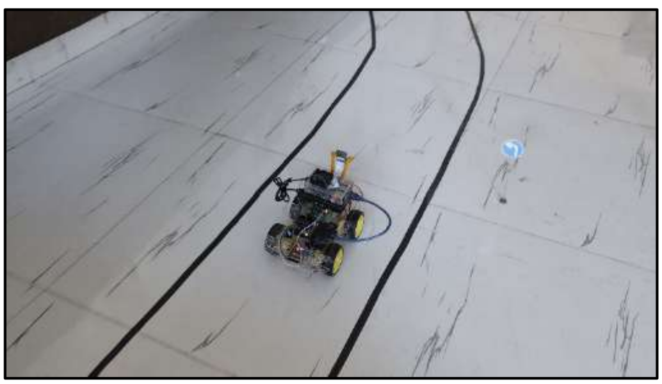
Fig. 10. Operation of the prototype vehicle

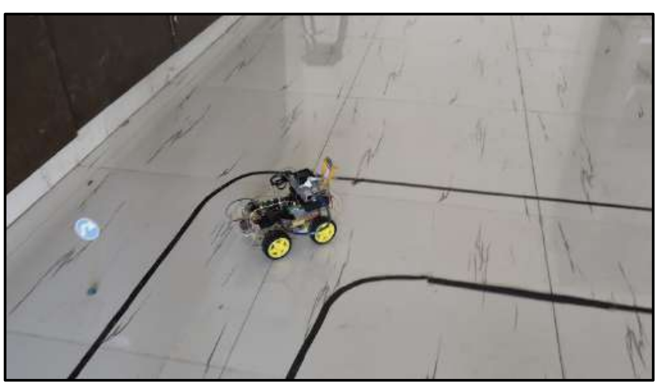
Fig. 11. Operation of the prototype vehicle

## VII. FUTURE SCOPE

The prospective applications of autonomous vehicles are broad, ranging from delivery services to scheduled public commuting. As technology advances, we expect to observe more innovative applications in the transportation sector.

*Improved Road Safety:* The endeavor entails ongoing progress in real-time monitoring and decision-making, aiming to reduce accidents by proficiently identifying and responding to dynamic obstacles, pedestrians, and ensuring strict adherence to traffic regulations.

*Vehicle-to-Vehicle (V2V) Communication:* Investigating the feasibility of real-time communication among vehicles to facilitate advanced information exchange on road conditions, promoting enhanced traffic coordination, and reinforcing overall road safety.

*Enhancing Logistics and Delivery Services:* This involves expanding the system's application to optimize delivery vehicle route planning, automating processes, minimizing delays, and refining logistics management for increased efficiency.

*Integration into Smart Cities:* This entails the enhanced integration of autonomous vehicles into smart city infrastructure, contributing significantly to sustainable and efficient transportation. These advancements align with broader objectives, fostering safer, interconnected, and efficient urban mobility.

## VIII. CONCLUSION

Our research in urban traffic detection for autonomous vehicles establishes a foundational contribution to the forefront of advancements. Integration of sophisticated detection models, encompassing lane, object, and traffic symbol detection, provides a tailored solution for navigating complex urban environments. The streamlined communication protocol with Arduino UNO ensures precise control, enabling responsive adaptations to dynamic traffic conditions.

The system detects the lanes, objects, and traffic symbols using individual models dedicated to each of them. Each model produces individual output. These outputs are then integrated together. Further, based on the boundary conditions set in the Raspberry Pi, a single command is sent to the Arduino UNO, which in turn operates the motors.

The implemented image preprocessing pipeline enhances information extraction, covering lane positions, objects, and traffic symbols. The contour detection algorithm identifies potential traffic signs, while subsequent image classification annotates detected signs with respective class names and probabilities accurately.

The core of our control mechanism lies in Arduino UNO's scrutiny of the processed input signal from Raspberry Pi. Employing predefined boundary conditions for specific commands ensures precise vehicle management, seamlessly adapting to changing traffic conditions.

This work highlights the vital interplay between detection models and control mechanisms, a crucial focus for advancing autonomous vehicle capabilities. Our work represents a pivotal step in the ongoing evolution of intelligent transportation systems, showcasing potential for robust and aware autonomous driving solutions.

## REFERENCES

[1] Prakash, M, M. Janarthanan, and D. Devi. "Multiple Object Detection or Autonomous Car using YOLO and CNN." In 2023 7th International Conference on Intelligent Computing and Control Systems (ICICCS), pp. 597-601. IEEE, 2023.

[2] Prasad, Abhisek Omkar, Pradumn Mishra, Urja Jain, Anish Pandey, Anushka Sinha, Anil Singh Yadav , Rajan Kumar. "Design and development of software of an autonomous vehicle using robot operating system." Robotics and Autonomous Systems161 (2023): 104340.

[3] Gautam, Sari ta, and Anuj Kumar. "Image-based automatic traffic light s detects ion system for autonomous cars: a review." Multimedia Tools and Applications (2023): 1- 48.

[4] R. Raffik, V. Singh, S. Kumar, J. Alanya-Beltran, P. Y. Jethabhai and R. Singh, "IOT based Electric Vehicles - Implementation," 2022 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES), Chennai, India, 2022, pp. 1-5, doi: 10.1109/ICSES55317.2022.991405

[5] K. B. Swain, S. Dash and S. S. Gouda, "Raspberry PI based Integrated Autonomous Vehicle using LabVIEW," 2017 Third International Conference on Sensing, Signal Processing and Security (ICSSS), Chennai, India, 2017, pp. 69-73, doi: 10.1109/SSPS.2017.8071567 C. J. Kaufman, Rocky Mountain Research Lab., Boulder, CO, private communication, May 1995.

[6] Bhaskar Barua, C. G. (2019). A Self-Driving Car Implementation Using Computer Vision for Detection and Navigation. IEEE.

[7] D. Vitas, M. Tomic and M. Burul, "Traffic Light Detection in Autonomous Driving Systems," in IEEE Consumer Electronics Magazine, vol. 9, no. 4, pp. 90-96, 1 July 2020, doi: 10.1109/MCE.2020.2969156.

[8] Guo Mu, Zhang Xinyu, Li Deyi, Zhang Tianlei, An Lifeng, Traffic light detection and recognition for autonomous vehicles, The Journal of China Universities of Posts and Telecommunications, Volume 22, Issue 1, 2015, Pages 50-56, ISSN 1005-8885, https://doi.org/10.1016/S1005-8885(15)60624-0.

[9] A. A. Assidiq, O. O. Khalifa, M. R. Islam and S. Khan, "Real time lane detection for autonomous vehicles," *2008 International Conference on Computer and Communication Engineering*, Kuala Lumpur, Malaysia, 2008, pp. 82-88, doi: 10.1109/ICCCE.2008.4580573.

[10] J. E. Hoffmann, H. G. Tosso, M. M. D. Santos, J. F. Justo, A. W. Malik and A. U. Rahman, "Real-Time Adaptive Object Detection and Tracking for Autonomous Vehicles," in *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 3, pp. 450-459, Sept. 2021, doi: 10.1109/TIV.2020.3037928.