Journal of Electrical Engineering and Automation (ISSN: 2582-3051)
www.irojournals.com/iroeea/

Enhancing Safe and Precise Maneuvering in Autonomous Electric Vehicle

Ashwin S.[1], Nafila B.[2], Sajana S.[3], Sutharsan S.[4], Arthi K.[5]

Department of Electrical and Electronics Engineering, Coimbatore Institute of Technology, Coimbatore, India

Email: [1]ashwine041@gmail.com, [2]mp12bnafila2003@gmail.com, [3]sajanasaj8@gmail.com, [4]suthars2021@gmail.com, [5]arthi.k@cit.edu.in

Abstract

This research deals with the design of an autonomous vehicle navigation system based on a master-slave computational paradigm, combining Raspberry Pi 4 and Arduino UNO for visual perception in real-time, decision-making, and actuation. The Raspberry Pi 4, supported by a Pi Camera module, performs lane detection and near-object identification through image processing techniques, and then sends over extracted data through serial communication to Arduino UNO. Contrary to traditional autonomous driving systems that are based largely on monolithic processing architectures, this two-part control paradigm maximally splits computational loads, thus maximizing real-time responsiveness and efficiency of computation. The approach coordinates an optimal interaction between image-based navigation and multi-sensor fusion to guarantee improved trajectory planning and obstacle avoidance. In addition, the servo-actuated ultrasonic module provides an extended spatial detection range over traditional static-sensor deployments, thus allowing for better environmental adaptability, especially in small or dynamically changing environments.

Keywords: Autonomous Driving, Obstacle Detection, Master-Slave Communication.

1. Introduction

This study is based on processing sensor data to identify obstacles along the path of the robot. The algorithm has brought together several methods to detect obstacles accurately in

DOI: https://doi.org/10.36548/jeea.2025.1.002
Received: 18.02.2024, received in revised form: 09.03.2025, accepted: 23.03.2025, published: 07.04.2025

real-time. The research aimed at improving autonomy and efficiency of robots in complex and dynamic settings [1]. It details the implementation of a control algorithm that enables the vehicle to navigate by detecting and avoiding obstacles in its path. The model uses sensors to gather environmental data and process it in real-time for decision-making. The study demonstrates how basic robotic concepts can be applied to a small-scale electric vehicle. This work contributes to the field of autonomous vehicle navigation, particularly in obstacle-rich environments [2]. The approach uses visual data from cameras to detect obstacles in the environment in real-time. It emphasizes processing speed and efficiency, which is important for autonomous navigation in dynamic environments. The goal of the system is to enhance the safety and autonomy of USVs (Unmanned Surface Vehicles) by allowing them to respond rapidly to obstacles. This work contributes to advancing autonomous marine vehicles, particularly in challenging environments [3].

To estimate friction with enhanced precision in real time, the authors utilize an EKF (Extended Kalman Filter). With better friction estimations, safety features, including anti-lock braking and stability control, are thus improved by estimating the vehicle's traction in more efficient ways. Hence, more robust friction information using EKF provides higher sensitivity to act on and stabilize the vehicle during maneuvers under any type of road surface. The proposed research can help to enhance vehicle safety and performance at various types of road surfaces [4].

This study integrates a friction adaptation model to improve the accuracy of state estimation, essential for vehicle control. By accurately estimating parameters like vehicle speed and tire forces, the method enhances the overall performance and stability of electric vehicles. The approach aims to optimize control strategies, especially in dynamic or challenging driving conditions. This research contributes to more precise and reliable operation of electric vehicles in real-world environments [5].

It examines the risks associated with high-voltage systems, battery management, and electrical components in EVs. The authors propose safety protocols and standards to mitigate hazards like electric shocks, fires, and system failures. The research also explores safety mechanisms for protecting both the vehicle users and technicians during maintenance. This work contributes to ensuring the safe operation and development of electric vehicle technologies [6].

The system uses fuzzy logic to process inputs from sensors and make real-time decisions on braking or steering to avoid collisions. This approach aims to enhance vehicle safety by preventing accidents in various driving conditions. The method adapts to different scenarios, making it more flexible than traditional rule-based systems. The research contributes to the development of intelligent safety systems that can autonomously prevent collisions in complex environments [7]. The system uses a combination of sensors to monitor the vehicle's environment and detect potential hazards. It incorporates Lebesgue sampling; a technique that helps optimize sensor data collection for more accurate decision-making. The goal is to prevent accidents and ensure stable operation of articulated vehicles, especially in challenging driving conditions. This research contributes to enhancing the safety and reliability of heavy-duty vehicles in dynamic environments [8].

The system allows the vehicle to move without depending on pre-mapped roads, but rather uses real-time data from the laser scanner to detect lane boundaries. This approach is meant to reduce the cost and complexity of autonomous systems while maintaining reliable navigation. The method is designed for autonomous service vehicles, which often operate in dynamic and less structured environments. This research contributes to making autonomous vehicles more accessible and adaptable for real-world applications [9].

The authors suggest an optimization model that can balance energy efficiency with other aspects like vehicle dynamics, road conditions, and traffic. This study focuses on the overall improvement of range and performance of AEVs (Autonomous Electric Vehicles) while maintaining safety and efficiency. Constraints are considered, including battery capacity and driving conditions, in order to optimize energy management. This work further leads to the creation of more sustainable and efficient autonomous electric vehicles [10].

The algorithm relies on three important features—lane edges, lane markings, and the vehicle's position relative to the lane—to detect and track lanes in real time. TFALDA (Three-feature based automatic lane detection algorithm) is designed to work under different road conditions and various environmental factors to improve lane-keeping performance. The method improves the safety and reliability of autonomous vehicles by ensuring accurate lane detection. This research contributes to the development of robust systems for autonomous driving in complex, dynamic environments [11-16].

The proposed approach towards self-driving outperforms the existing because it merges a hybrid processing design with the advanced image processing capabilities of Raspberry Pi and the real-time motor control optimality of Arduino. Unlike the typical systems which employ either a high-power computer module or microcontrollers, the master-slave architecture finds equilibrium between performance and cost. The dynamic obstacle detection system boosts security with an ultrasonic sensor mounted on a servo, with multi-angle obstacle evaluation, unlike typical low-cost autonomous vehicles. The system also features redundant IR-based protection for improving reliability in poor environments and ensuring greater flexibility in real-world implementation.

2. Proposed System

2.1 Overview of the System

This modular system design ensures optimal computational efficiency, balances the task distribution between the controllers, and provides robust navigation and obstacle avoidance. A master-slave combination of a Raspberry Pi 4 and Arduino UNO was integrated to ensure effective and reliable vehicle control. The Raspberry Pi, having a Pi camera module, performed lane and object detection by processing the visual data in real-time and communicate the processed signals serially to the Arduino. It controls the motor using an L298 motor driver and is equipped with an ultrasonic sensor mounted on a servo motor for angle-based obstacle detection. Two IR sensors are used to enhance line following as well as to provide secondary safety.

2.2 Block Diagram

The Block Diagram of our Autonomous driving vehicle is shown as in Figure.1,

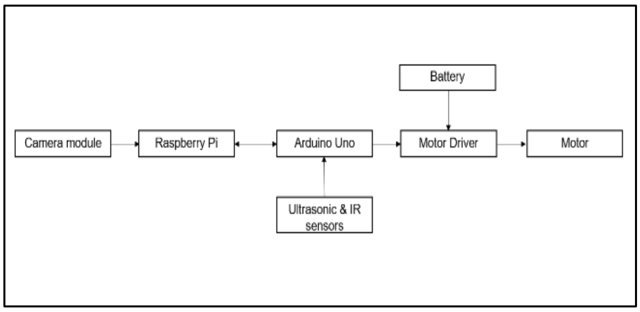

Figure 1. Block Diagram

The entire system is powered by a battery pack consisting of two 3.7V lithium-ion batteries connected in series, supplying power to both controllers and the motors. The Raspberry Pi receives input from the camera module, processes the visual data, and makes decisions accordingly. The camera module is programmed to detect lanes for navigation and identify objects in front of the vehicle. During turns, the system calculates the angle of lane deviation and sends this data to the Arduino Uno. The Arduino Uno then commands the servo motor to rotate at the desired angle, ensuring that the ultrasonic sensor scans for obstacles at turning points as well.

The infrared (IR) sensor detects lane markings by identifying changes in surface colour. These sensors are connected to the Arduino Uno to reduce the computational load on the Raspberry Pi and ensure efficient operation of the microcontroller. Serial communication is established between the Raspberry Pi and Arduino for data exchange. The motor driver generates the necessary actuation signals to control the four motors of the vehicle.

2.3 Flowchart

Figure.2 shows the decision-making flowchart of an autonomous vehicle with the help of IR sensors, an ultrasonic sensor, and a camera module for navigating and avoiding obstacles. The system starts by taking input from an IR sensor for lane detection. If there is no lane, the vehicle halts; otherwise, it continues by taking input from the camera module. The camera module checks if the way ahead is free. If there are no obstacles, the vehicle moves ahead and takes further input from the ultrasonic sensor for further confirmation. But if there is an obstacle, the system activates an obstacle avoidance mechanism.

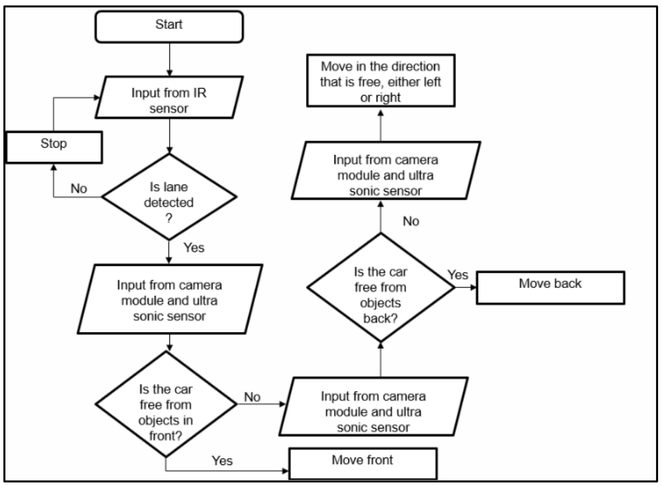

Figure 2. Flowchart

To steer clear against the obstacles, the car first ascertains whether it can proceed backward safely by taking input from the camera module and the ultrasonic sensor. If the rear is free, it backs up; if not, it checks if the left side is clear. If it is clear, the vehicle goes left; if not, it goes right as a last resort. After any such movement, the system keeps taking input from the IR sensor, camera module, and ultrasonic sensor to re-analyse its surroundings and figure out the next course of action. This never-ending loop makes sure that the vehicle moves safely, steering clear of obstacles while keeping track of a lane.

3. Simulation Model

3.1 Vehicle Body

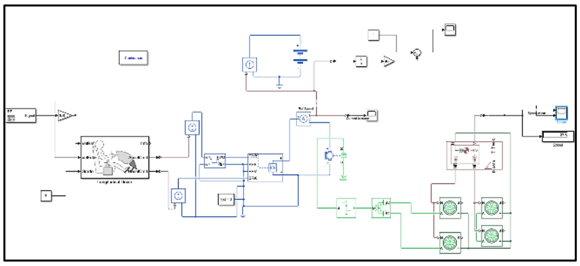

Figure 3. Simulink Model

The Figure 3 is a longitudinal vehicle control system with speed regulation in the motor dynamics. Blocks of the system have been integrated to simulate the working of a real-time automobile model. The upper side of the model represents an equivalent circuit of a DC motor from which the torque to steer the vehicle is drawn. It solves the dynamic equations of the motor by an electrical subsystem.

The equation of motion for the motor is,

$$J_m\left(\frac{d\omega_m}{dt}\right)+B_m\omega_m=T_m-T_l \tag{1}$$

where,

J_m = Moment of inertia

B_m = Damping coefficient

T_l =Load Torque

T_m=Motor Torque

The lower portion of the system holds a longitudinal driver model, whose inputs are desired velocity (v) and feedback velocity, together with the road gradient as its input signals. These inputs are processed by the driver block to compute acceleration and braking commands. All these commands are processed to give a reference signal to the motor controller. Vehicle Longitudinal dynamics depend upon traction force (F_t), rolling resistance force (F_r) and gravitational force (F_g) due to road gradient.

$$m\left(\frac{dv}{dt}\right)=F_t-F_r-F_g \tag{2}$$

The driver model processes the desired velocity (v_{ref}) and feedback velocity to generate acceleration and braking commands.

$$a_{cmd}=K_p(v_{ref}-v)+K_i\int((v_{ref}-v)\,dt+K_d(d(v_{ref}-v)/dt) \tag{3}$$

where,

a_{cmd} = Acceleration Command

K_p, K_i, K_d are the PID controller gains

Simulink motor control based on PWM is similar to how the Arduino produces PWM signals for controlling car speed. Moreover, the H-Bridge inverter in the proposed Simulink model mimics the motor drive circuit in the research for forward, reverse, and braking movements. Similarly, the sensor control system in the model is identical to the interaction between Raspberry Pi and Arduino. The longitudinal driver takes desired speed, actual velocity, and road grade as inputs to supply acceleration and brake commands. Likewise, Raspberry Pi, is the master controller for the lane and object detection information that is fed through the Pi Camera. It then passes control signals through serial communication to Arduino, which varies the motor speeds accordingly.

The Simulink model mimics road slopes to verify how the car dynamically compensates speed based on its compensation. In the research, IR sensors assist in detection and lane alignment, such that the car remains on the given path. After the car begins drifting, Arduino compensates motor movement based on IR sensor signals, and the Simulink model compensates velocity to compensate based on road conditions.

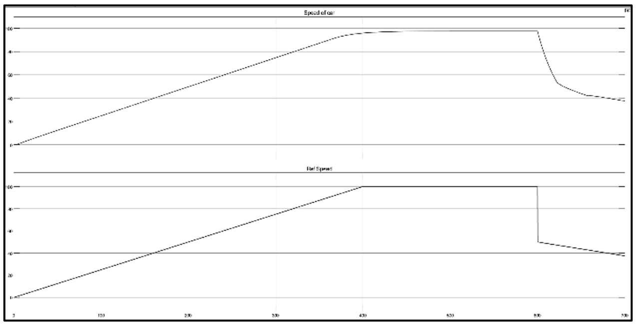

Figure 4. Reference vs Actual Speed

The simulation results in the Figure.4 exhibits the dynamic response of the vehicle to the reference speed input. The Reference Speed curve is the curve of the desired speed, and the Speed of Car curve is that of the actual speed from the system. In this case, the speed of the vehicle closely follows the desired speed initially, showing that good performance by the control system exists as far as acceleration is smooth. At this point of stabilization of the reference speed, the vehicle speed also stabilizes to have a minimal lag. But at the same time, when it lowers the reference speed at a sharp rate, the system begins to brake and lowers the actual vehicle speed gradually. In this way, it reveals its lag during deceleration. This lag during deceleration also indicates that the system is transient and can be optimized to show even faster response.

3.2 Estimation of Road Friction Coefficient

The estimation of the tire-road friction coefficient is one of the important aspects of vehicle dynamics since it affects stability, braking, and traction control. The friction coefficient depends on road surface conditions and is generally modelled using mathematical equations that describe the relationship between slip and friction. The friction coefficient is defined by an exponential function, taking into account the maximum friction value and the slip rate change.

This section discusses the friction estimation model using a predefined function, in which the parameters, determine the shape of the friction curve as a function of slip. The function allows to analyse the behaviour of friction across different road surfaces and optimize vehicle control strategies accordingly as shown in Table 1.

ISSN: 2582-3051

Table 1. Friction Estimation

Surface condition	C1	C2	C3
Dry asphalt	1.2801	23.99	0.52
Dry concrete	1.1973	25.16	0.5373
Wet asphalt	0.857	33.82	0.347
Snow	1.946	94.12	0.0646
Ice	0.05	306.3	0

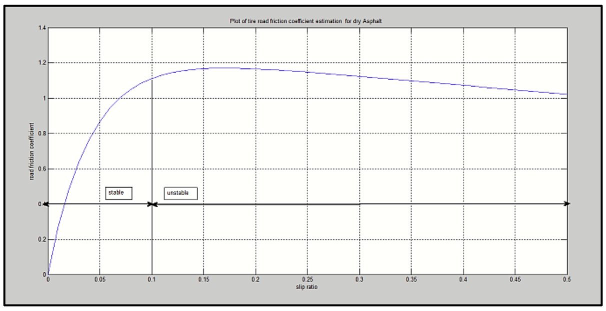

Figure 5. Output graph

High values in Figure 5 stands for high peak friction, which is typical for dry roads such asphalt and concrete. Higher values mean friction increases sharper at low slip; utmost importance should be given to responsive braking behaviour. Higher values are obtained for a greater decrease in friction at high slip, which is the case of wet or icy roads.

4. Hardware Implementation

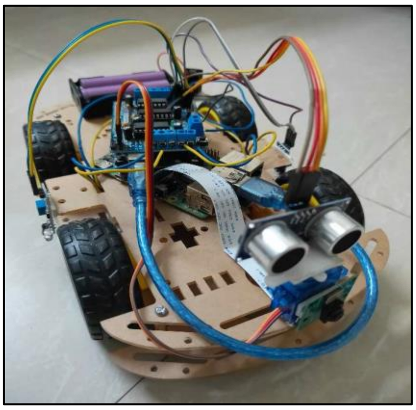

Figure 6. Prototype Model

4.1 Master-slave Integration

The hardware model as shown in Figure.6 uses a master-slave architecture, where Raspberry Pi 4 acts as the master and Arduino UNO as the slave. Master (Raspberry Pi) is used for high-level processing, consisting of lane detection and obstacle identification with a Pi camera module. The processed data is then sent to the Arduino through serial communication. The slave (Arduino) will do all the low-level motor control, so the signals received make it very responsive in real time. This separation of the computational tasks shall maximize the performance by using the Raspberry Pi for vision-based tasks and the Arduino for motor actuation and for some additional sensor-based safety mechanisms.

4.2 Data Gathering and Transfer

The process of data collection is triggered by the Raspberry Pi 4 and it keeps recording real-time video streams through the Pi camera module. Advanced image processing methods like edge detection and colour segmentation are used to detect lane markings and identify obstacles in the path of the vehicle. The Raspberry Pi processes this visual information to find the position of the vehicle in the lane.

After processing the obstacle data, the Raspberry Pi converts the lane position and obstacle detection data into a reduced command format. The data is then sent to the Arduino UNO through UART-based serial communication.

4.3 Data Reception and Processing

After the Arduino UNO receives processed data from the Raspberry Pi 4 through serial communication, it interprets the commands to make suitable motor responses. The received data normally comprises lane position data and obstacle detection signals, which are used to control the movement and guidance of the vehicle. Depending on this data, the Arduino modifies the motor speeds and directions with an L298 motor driver for smooth lane tracking and obstacle evasion.

Apart from data from the Raspberry Pi, the Arduino constantly receives real-time inputs from onboard sensors such as an ultrasonic sensor on a servo motor and two IR sensors. The rotating ultrasonic sensor, which can detect obstacles at different angles, estimates the distance to the obstacles and serves as a secondary safety to the vehicle. The IR sensors help in detecting lane markings and near objects. If an imminent obstacle is sensed, the Arduino initiates autonomous correctional actions, such as a slowdown, halt, or slight steering correction. Through the integration of information from multiple sources, the system provides secured decision-making, which improves the accuracy, reliability, and safety of the autonomous vehicle in dynamic scenarios.

4.4 Mathematical Model for Obstacle Detection

To enhance the obstacle avoidance mechanism, the approach uses the Time of Flight (ToF) principle for the ultrasonic sensor:

$$d = v * \frac{t}{2} \tag{4}$$

where,

d = Distance to obstacle from vehicle

v = Speed of sound in air (~343 m/s)

t = Time taken for the ultrasonic pulse to return

Additionally, the steering angle θ for avoiding obstacles is computed using:

$$\theta = \tan^{-1}\left(\frac{d_{obstacle}}{d_{vehicle}}\right) \tag{5}$$

where,

$d_{obstacle}$ = Lateral distance of the obstacle

$d_{vehicle}$ = Distance from the vehicle's centre to the detected object

These equations help dynamically adjust the steering and speed of the vehicle, ensuring collision avoidance while maintaining lane integrity.

5. Future Perspectives

The autonomous driving domain is changing at a pace that is really quite fast. The research represents a big step forward in demonstrating the feasibility of a low-cost, affordable autonomous system. Future generations of the system can make use of more advanced sensors and AI-based algorithms, so that the vehicle will be able to sense and respond better to the world around it. The current version of the system uses simple sensing technologies, such as ultrasonic sensors and IR sensors, but potential future systems are able to adopt more advanced technology, such as LiDAR sensors, radar-based sensors, and computer vision algorithms powered by machine learning techniques for sensing and decision-making.

Further technological advancements in communications protocols and the use of the cloud will even allow for shared and processed real-time data which can lead to more intelligent methods of navigation, safety, etc. This research goes forward into the future with a lot of potential implications for improving the autonomous driving performance in controlled domains and for extensions to practical applications in smart cities, autonomous public transportation, and personal mobility solutions, which would significantly contribute to safety, traffic flow, and energy efficiency.

6. Conclusion

This research brings out the importance of cost effective in autonomous vehicles. It combines the immense potential of both hardware and software to create intelligent systems. It also highlights the advantages of a hybrid processing approach, demonstrating that high-performance autonomous navigation can be achieved without reliance on expensive computing resources. It provides an important milestone for all future innovations in the area of autonomous vehicle technology, with a focus on scalability, adaptability, and real-world applications. This research is made to be a good contribution toward advancements of autonomous transportation systems with the continuation of development and integration into advanced technologies in intelligent transportation systems, automated delivery robots, and self-navigating platforms in structured and unstructured environments, thus paving the way for safer, smarter, and more efficient mobility solutions.

References

[1] Jin, Ping-Fan, Jung-Hyun Kim, and Jung-Ha Kim. "Design of unmanned vehicle advanced braking system using smart motor." In 2015 12th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), IEEE, 2015. 167-171.

[2] Blyth, Pascale-L., Milos N. Mladenovic, Bonnie A. Nardi, Norman M. Su, and Hamid R. Ekbia. "Driving the self-driving vehicle: Expanding the technological design Horizon." In 2015 IEEE International Symposium on Technology and Society (ISTAS), IEEE, 2015. 1-6.

[3] Rehman, Abbad Ur, Zohaib Mushtaq, and Muhammad Attique Qamar. "Fuzzy logic based automatic vehicle collision prevention system." In 2015 IEEE Conference on Systems, Process and Control (ICSPC), IEEE, 2015. 55-60.

[4] Fenesan, Andrei, Teodor Pana, Daniel Szöcs, and Wen-Hua Chen. "Building an electric model vehicle and implementing an obstacle avoidance algorithm." In 2012 International Conference and Exposition on Electrical and Power Engineering, IEEE, 2012. 49-53.

[5] Freschi, Fabio, Massimo Mitolo, and Riccardo Tommasini. "Electrical safety of electric vehicles." In 2017 IEEE/IAS 53rd industrial and commercial power systems technical conference (I&CPS), IEEE, 2017. 1-5.

[6] Kristan, Matej, Vildana Sulić Kenk, Stanislav Kovačič, and Janez Perš. "Fast image-based obstacle detection from unmanned surface vehicles." IEEE transactions on cybernetics 46, no. 3 (2015): 641-654.

[7] Enisz, Krisztian, Denes Fodor, Istvan Szalay, and Gabor Kohlrusz. "Improvement of active safety systems by the extended Kalman filter based estimation of tire-road friction coefficient." In 2014 IEEE International Electric Vehicle Conference (IEVC), IEEE, 2014. 1-5.

[8] Zhao, Linhui, and Zhiyuan Liu. "Vehicle state estimation with friction adaptation for four-wheel independent drive electric vehicle." In Proceeding of the 11th World Congress on intelligent Control and Automation, IEEE, 2014. 4527-4531.

[9] Ariyanto, Mochammad, Gunawan D. Haryadi, Munadi Munadi, Rifky Ismail, and Zulfa Hendra. "Development of low-cost autonomous emergency braking system (AEBS) for an electric car." In 2018 5th International Conference on Electric Vehicular Technology (ICEVT), IEEE, 2018. 167-171.

[10] Mariaraja, P., T. Manigandan, S. Arun, RM Meenakshi Sundaram, and K. Dhenesh. "Design and implementation of advanced automatic driving system using raspberry pi." In 2020 5th International Conference on Communication and Electronics Systems (ICCES), IEEE, 2020. 1362-1367

[11] AC, Ramachandra. "Advanced Driving Assistance System for Cars Using Raspberry Pi." Indian Journal of Science and Technology 15, no. 40 (2022): 2056-2065.

[12] McCann, Roy A., and Anh T. Le. "A new multisensor network for collision avoidance and jackknife prevention of articulated vehicles using Lebesgue sampling." In 2005 IEEE Vehicle Power and Propulsion Conference, IEEE, 2005.

[13] Zhiwei, Song, Huang Weiwei, Wu Ning, Wu Xiaojun, Wong Chern Yuen Anthony, Vincensius Billy Saputra, Benjamin Chia Hon Quan et al. "Map free lane following based on low-cost laser scanner for near future autonomous service vehicle." In 2015 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2015. 706-711.

[14] Madhavan, T. R., and M. Adharsh. "Obstacle detection and obstacle avoidance algorithm based on 2-D RPLiDAR." In 2019 International Conference on Computer Communication and Informatics (ICCCI), IEEE, 2019. 1-4.

[15] Zhang, Ying, Zhaoyang Ai, Jinchao Chen, Tao You, Chenglie Du, and Lei Deng. "Energy-saving optimization and control of autonomous electric vehicles with considering multiconstraints." IEEE Transactions on Cybernetics 52, no. 10 (2021): 10869-10881.

[16] Yim, Young Uk, and Se-Young Oh. "Three-feature based automatic lane detection algorithm (TFALDA) for autonomous driving." IEEE Transactions on Intelligent Transportation Systems 4, no. 4 (2003): 219-225.