International Journal on Science and Technology (IJSAT)

E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

Design and Implementation of a Lane Detection-Based Steering Control System for an Autonomous Mobile Robot

Helmi Jamaludin [1], Najmi Hafizi Zabawi [2], Amrul Akil Ahmad Hashim[3]

[1,3] Electrical Engineering Department, Sultan Haji Ahmad Shah Polytechnic, Malaysia
[2] Lifelong Learning Department, Kuantan Community Collage, Malaysia

Abstract

The development of autonomous mobile robots (AMRs) has advanced rapidly, driven by increasing demands for intelligent and reliable navigation systems. This paper presents the design and implementation of a lane detection-based steering control system to enable an AMR to follow lanes autonomously. The system employs RoboRealm software for real-time image processing, allowing lane markers to be detected from camera input and translated into steering commands for the robot car. A modular vision-processing architecture, integrating a camera, Mini PC, microcontroller, servo motor, and DC motors, is utilized to simplify development while maintaining sufficient accuracy and responsiveness for controlled environments. Lane detection is achieved through image pre-processing techniques such as color thresholding, blob filtering, and Hough Transform, with steering adjustments dynamically based on the computed Center of Gravity (COG) between lane boundaries. Experimental results demonstrate that the system successfully maintains lane adherence during both straight and turning maneuvers with minimal deviation, validating the effectiveness of the RoboRealm-based approach for small-scale autonomous navigation. Future work will focus on enhancing system robustness under complex lighting conditions and dynamic environments through adaptive or learning-based vision techniques.

Keywords: Autonomous Mobile Robot, Lane Detection, RoboRealm

1. Introduction

Autonomous navigation is a fundamental capability for mobile robots in a wide range of applications, from industrial automation to self-driving vehicles. A key component of autonomous navigation is the ability to detect and follow lanes or paths. Traditional lane-following algorithms often involve complex image processing pipelines and demand significant computational resources. To address these challenges, this research focuses on using RoboRealm, a flexible and user-friendly vision-processing platform, to implement a lane detection and steering control system.

The primary objective of this project is to enable a robot car to detect lanes and adjust its steering accordingly in real-time. Using RoboRealm offers several advantages, such as modular programming, rapid prototyping, and the ability to integrate easily with various hardware platforms. This study aims to demonstrate that a lightweight, accessible solution can achieve reliable lane-following performance suitable for educational, research, and prototyping purposes. The system is evaluated based on its lane-tracking accuracy and responsiveness under different lighting and track conditions.

2. Literature Review

Lane detection and tracking have been actively researched in the field of autonomous navigation. Various techniques have been proposed, ranging from classical image processing methods to modern deep learning approaches. This section reviews several relevant methods used in lane-following systems.

2.1 Classical Image Processing Methods

Early lane detection systems commonly used color thresholding and edge detection techniques to identify lane markings. The Bottom-Up Line Extraction method proposed by Tusor et al. utilizes a bottom-up approach, identifying small line segments (3 pixels long) from the edge map and grouping them based on characteristics. This method is efficient due to its parallel computing capabilities, allowing for rapid line extraction (Tusor et al., 2018). Another method introduce by Rasiq et al., a fast parallel processing algorithm that detects straight lines directly from binary images without prior noise removal. This method employs multiple processing elements to efficiently scan and evaluate line segments (Rasiq et al., 2019).

2.2 Machine Learning-Based Lane Detection

More recent approaches have applied machine learning techniques, such as Support Vector Machines (SVM) and Convolutional Neural Networks (CNN), for lane detection tasks. Deep learning models, like SCNN (Spatial CNN) (Patel et al., 2024), can learn complex features from large datasets and handle variations in road conditions, lighting, and occlusions. Although highly accurate, these methods require significant computational resources, large training datasets, and are often unsuitable for small mobile robots due to processing limitations (Lin et al., 2024).

2.3 Modular Vision Processing Using RoboRealm

An alternative lightweight approach is to use modular vision design. The system comprises distinct modules for object recognition, manipulation, and interaction, facilitating focused improvements in each area (Flores-Rodriguez et al., 2017). By using modular vision-processing software such as RoboRealm, the tasks can be simplified. RoboRealm provides a collection of pre-built modules for tasks like color segmentation, line following, and centroid tracking (RoboRealm Documentation, 2025). These modules allow rapid prototyping without deep programming knowledge, making it ideal for educational purposes and smaller research projects. While not as robust as deep learning methods, RoboRealm-based systems offer a good balance between development simplicity and functional performance in controlled environments.

3. Methodology

3.1 Block Diagram

This section details the design of the self-driving car robot system, which is aimed at achieving precise and responsive control over both steering and propulsion. The system is divided into two primary subsystems: the steering system and the rear-wheel drive system. Each subsystem operates independently yet in coordination to provide smooth navigation and movement.

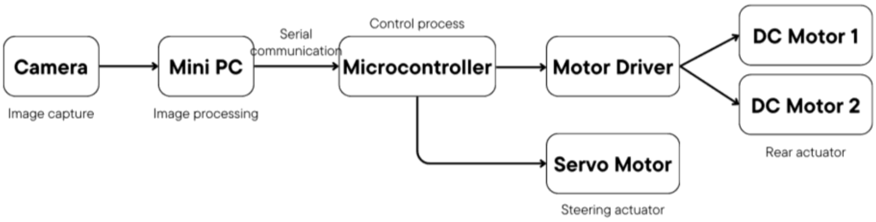

Figure 1: Block diagram

3.1.1 Steering System

The steering system is responsible for controlling the angle of the front wheels to navigate the vehicle in the desired direction. This is achieved through a servo motor, which receives input from both a potentiometer (for manual fine-tuning) and commands from the control unit (sent via serial communication). The servo motor adjusts the front wheels' direction based on the steering angle input, allowing for precise and responsive directional control.

3.1.2 Rear-Wheel Drive System

The rear-wheel drive system is powered by two DC motors, which are controlled by a motor driver. The motors receive throttle commands (also via serial communication), which dictate the speed of the vehicle. The system ensures that both motors work in sync, providing stable and smooth propulsion for the robot. The ability to modulate the motor speed allows the robot to accelerate or decelerate as needed, ensuring safe operation.

3.2 Hardware Architecture

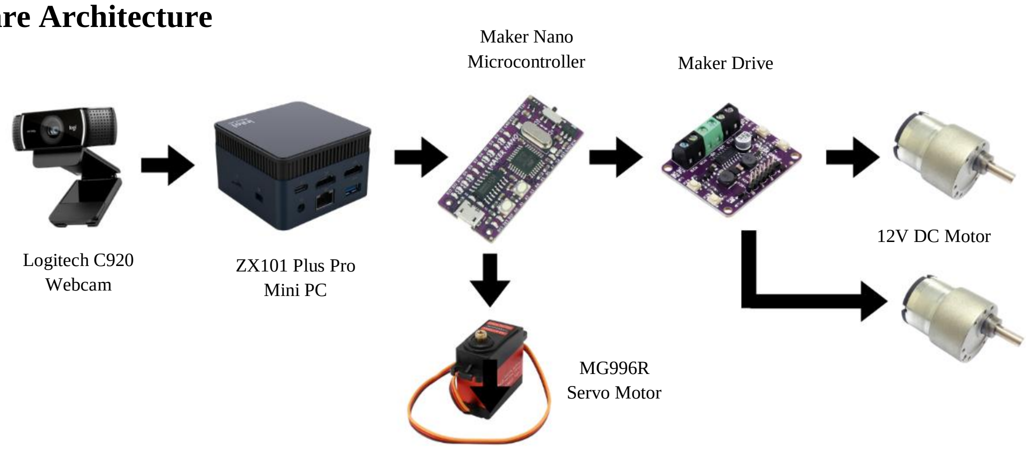

Figure 2: Hardware architecture

Figure 2 shows the hardware architecture of a self-driving robot system. The system starts with a Logitech C920 webcam that captures real-time images of the environment. These images are processed by the ZX101 Plus Pro Mini PC, which functions as the main processing unit. Inside the Mini PC, RoboRealm software is used for image processing tasks such as object recognition, lane detection, and path planning, while Arduino IDE manages communication and control commands for interfacing with the microcontroller. After analysis, the processed data is transmitted to the Maker Nano microcontroller, which acts as an intermediary to control the physical hardware. The microcontroller sends signals to a

MG996R servo motor for steering adjustments and to a Maker Drive motor driver that controls two 12V DC motors for rear-wheel propulsion. The system is powered by two dedicated power sources to ensure stable operation. A 12.6VDC 9800mAh Polymer Lithium-Ion battery supplies power to the Mini PC, while a separate 12VDC lithium battery powers the DC motors through the Maker Drive. This dual power setup helps to isolate the high current requirements of the motors from the sensitive computing components, improving overall system reliability. With the integration of the camera, Mini PC, microcontroller, servo motor, DC motors, and stable power supplies, the robot is capable of autonomous navigation with real-time speed and direction adjustments based on visual data.

3.3 Prototype Designing

Figure 3: Waveshare JetRacer AI mobile vehicle

Figure 3 shows a four-wheeled Waveshare JetRacer AI mobile vehicle, designed for autonomous navigation or remote-controlled applications. The structure consists of a sturdy green metal frame that supports various electronic and mechanical components. At the front, a MG996R servo motor with a torque of 9kg/cm is connected to a steering linkage, allowing precise control of the front wheels for turning. The rear section houses two 37-520 DC motors with a reduction rate of 1:10 and idle speed of 740 RPM, which drive the rear wheels and provide propulsion (Podbucki & Marciniak, 2022). These motors are wired for external control, through a motor driver and a microcontroller. The four rubber wheels offer good traction, with the front wheels dedicated to steering and the rear wheels handling movement.

The central processing unit responsible for controlling the steering and drive systems, as well as processing data from the camera, is the ZX01 Plus Pro Mini PC. Its key technical specifications are presented in Table 1. The ZX01 Plus Mini PC utilized in this project is powered by an Intel Processor N100, featuring an integrated Intel UHD Graphics GPU with 24 Execution Units (EUs) operating up to 750 MHz. It is equipped with 12GB of DDR4 RAM and runs on the Windows 11 operating system, providing a stable platform for executing RoboRealm software, facilitating real-time image acquisition, vision processing, and steering control tasks.

Table 1: ZX101 Plus Pro Mini specification

GPU	Intel UHD Graphics
CPU	Intel N100 3.4GHz
RAM	12GB DDR4
Operating Sytem	Windows 11

Figure 4 shows the physical prototype of the self-driving car robot. The robot integrates a webcam, Mini PC, microcontroller, servo motor, DC motors, and two separate battery supplies. The compact design measures 26.5 cm in length, 19.5 cm in width, and 14 cm in height, with a total weight of approximately 1 kg, making it suitable for indoor navigation and experimental testing.

Figure 4: Waveshare JetRacer AI mobile vehicle

3.4 Lane Detection using RoboRealm

In this project, lane detection is implemented using RoboRealm, a vision processing software designed for real-time robotic applications. The software offers a comprehensive set of image processing algorithms through a user-friendly Graphical User Interface (GUI), enabling rapid development of vision-based control systems (Yusof et al., 2021; Sirisha & Patnaik, 2012). The detection process begins with capturing real-time video frames from the forward-facing camera mounted on the robot. These frames undergo pre-processing steps such as color thresholding, blob filtering, edge detection and Hough Transform to isolate two critical visual features: the left solid boundary line and the dashed centerline marking (Figure 5). Following the extraction of these lane boundaries, RoboRealm computes the Center of Gravity (COG) between the left solid line and the dashed centerline to determine the robot's lateral position within the designated left lane segment. Maintaining the COG centrally between these two references is crucial for stable and accurate lane following. Figure 6 illustrates the final processed output from RoboRealm, where the detected lane boundaries are marked by red and blue points, and the computed COG is represented by a green triangle. The corresponding servo motor angle, calculated based on the COG position, is displayed in real time at the top of the interface. When the COG shifts left or right relative to the ideal center, RoboRealm calculates a corrective steering value and transmits it via serial communication to the Maker Nano microcontroller. The microcontroller interprets this data and adjusts the servo motor angle accordingly to realign the robot towards the lane center.

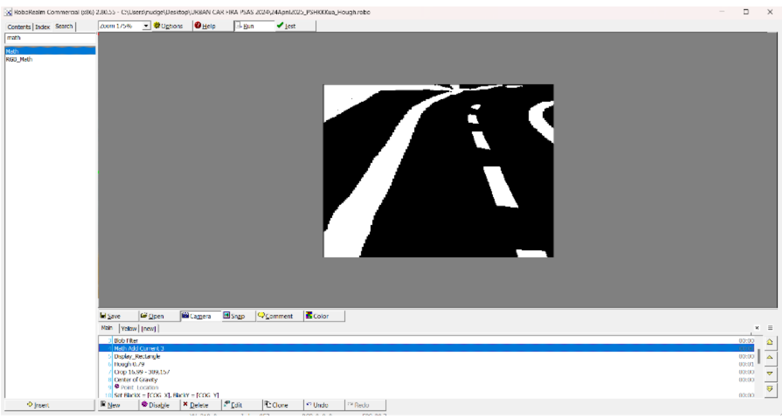

Figure 5: A binary masked image generated by RoboRealm isolates the left solid lane line and the dashed centerline for COG computation.

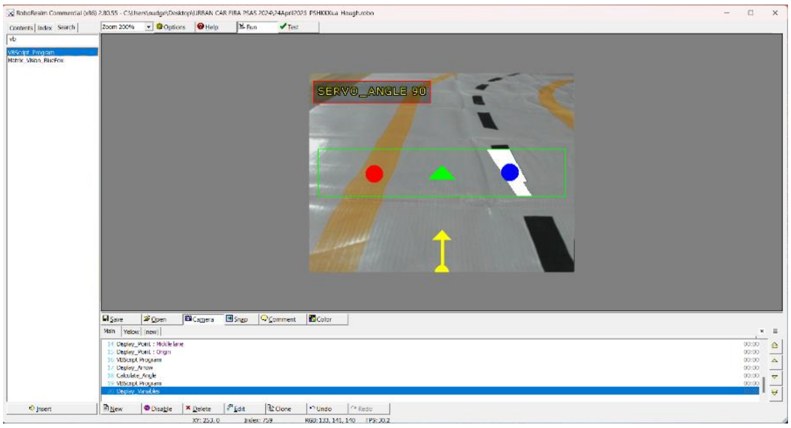

Figure 6: Final processed output in RoboRealm showing lane detection, COG computation between the left lane boundary and centerline, and servo angle adjustment.

3.5 Motor Controller

The Mini PC, running RoboRealm, processes real-time video input from the camera to extract the center of gravity (COG) of the lane path and determine the corresponding throttle and steering angle values. These control values are transmitted to the Maker Nano microcontroller via serial communication. Upon receiving the data, the microcontroller parses the throttle (T) and steering (S) values, and subsequently executes the motion function to control both the steering servo motor and the rear drive motors. The servo motor, managed through the servo library, adjusts the front wheel angle based on the Ackerman steering principle. The control logic is such that when the servo is positioned at 90°, the front wheels are aligned straight, allowing the robot to move forward in a straight line. If the servo angle is greater than 90°, the front wheels turn left, causing the robot to steer to the left. Conversely, if the servo angle is less than 90°, the front wheels turn right, causing the robot to steer to the right. This control logic enables dynamic steering adjustments based on real-time lane detection data, ensuring stable and accurate navigation. Additionally, dual DC motors connected to Cytron MD10 motor drivers regulate the forward and reverse movements of the robot, providing smooth propulsion and speed control based on the throttle value.

4. Experiments and Results

The experimental setup consists of an indoor lane track provided by Waveshare, placed on a flat wooden floor surface to ensure stable testing conditions (Figure 7). The track features two concentric orange boundary lines with dashed black lane markings in the center, serving as the reference for lane detection. A four-wheeled autonomous mobile robot is used for the experiments. The robot is initially positioned in the left lane before starting each test, ensuring consistent starting conditions for both clockwise (CW) and counter-clockwise (CCW) movements. Testing is performed in both directions to evaluate the robot's steering control system performance for left and right turning maneuvers, validating the responsiveness and accuracy of the lane-following algorithm.

Figure 7: Experimental setup for lane-following testing using the Waveshare track and autonomous mobile robot.

The experimental results evaluate the servo motor angle behavior during lane-following tasks by focusing on three movement conditions: straight movement, right turn, and left turn. During clockwise (CW) movement, the robot predominantly moved straight and performed right turns at curved sections. When moving straight, the servo motor maintained an angle close to 90°, indicating that the front wheels were aligned parallel to the lane center. As shown in Figure 8, the green triangle representing the center of gravity (COG) was centered between the left and right lane boundaries, and the servo motor reading was recorded at 90°. During right turns, the COG shifted slightly towards the right side of the frame, prompting the system to reduce the servo angle to 73°, as observed in Figure 9. This adjustment enabled the robot to steer right and maintain alignment with the lane. In counter-clockwise (CCW) movement, the robot similarly maintained a servo angle close to 90° during straight segments. However, when executing left turns, the COG shifted towards the left side of the frame, causing the servo motor angle to increase to 117°, as shown in Figure 10. This increase in servo angle allowed the robot to perform a left turn while remaining centered within the lane boundaries. Overall, the results demonstrate that the lane detection and steering control system can accurately interpret the position of the robot within the lane and dynamically adjust the steering angle in real-time. The consistent servo motor responses during straight and turning movements validate the effectiveness of the center of gravity (COG)-based lane-following algorithm implemented in the autonomous mobile robot.

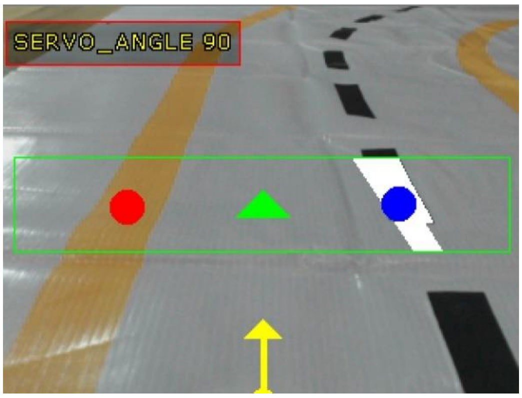

Figure 8: Camera view during straight movement showing lane alignment and servo motor angle at 90°.

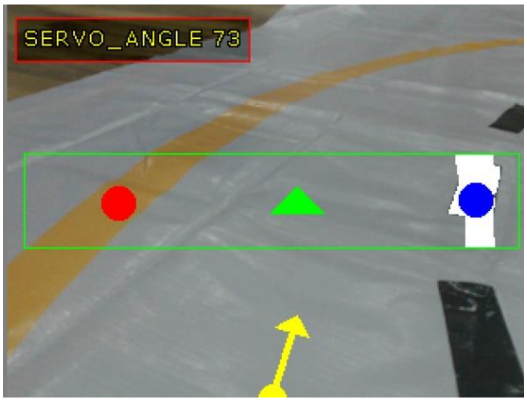

Figure 9: Camera view during right turn showing showing lane alignment and servo motor angle at 73°.

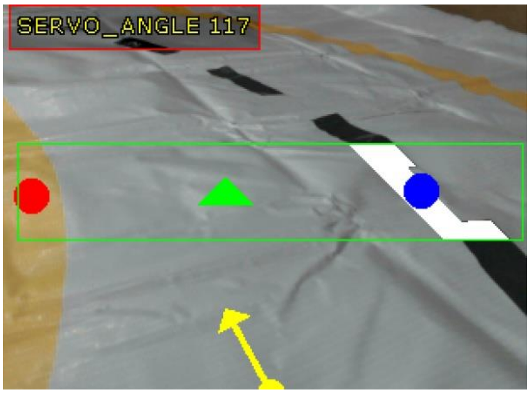

Figure 10: Camera view during left turn showing servo lane alignment and motor angle at 117°.

5. Conclusions

This study successfully designed and implemented a lane detection-based steering control system for an autonomous mobile robot using computer vision and real-time servo motor actuation. The center of gravity (COG) was specifically computed between the left solid boundary and the dashed centerline of the lane, allowing the system to accurately monitor the robot's lateral position within the left lane throughout the test runs. The experimental results confirm that the system is able to maintain the robot's trajectory by dynamically adjusting the steering angle based on real-time COG values derived from the camera input. The robot consistently performed straight movements, right turns (in CW direction), and left turns (in

CCW direction) with accurate servo motor responses corresponding to the detected lane deviations. By achieving stable and responsive steering behavior based on precise lane referencing, the system effectively fulfills the research objectives of demonstrating real-time lane detection, accurate lane-following capability, and efficient servo motor control for autonomous mobile navigation. Future work may explore the integration of adaptive thresholding techniques or machine learning-based lane detection methods to enhance system robustness under varying lighting conditions and more complex track geometries.

References

1. Lin Z., Ye P., Chen T., Tang S., "Multi-dimensional Search with Strip Convolution and R-Squared Loss for Lane Detection", International Joint Conference on Neural Networks (IJCNN), 2024, 1–8.
2. Patel S., Shah P., Patel D., Patel N., Patel J., "Hybrid Lane Detection System Combining SCNN, Canny Edge Detection, and Hough Transform", 8th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), 2024, 327–332.
3. Pires R., de Smet P., Bruyland I., "Line extraction with the use of an automatic gradient threshold technique and the Hough transform", IEEE International Conference on Image Processing (ICIP), 2000, 3, 909–912.
4. Podbucki K., Marciniak. T., "Aspects of autonomous drive control using NVDIA Jetson Nano microcomputer", Proceeding of the 17th Conference on Computer Science and Intelligence System, 2022, 30, 117-120.
5. RoboRealm Documentation, "RoboRealm Vision Software," Accessed: 2025. [Online]. Available: https://www.roborealm.com/
6. Sirisha K., Patnaik R., "Level Control Using Image Recognition Using RoboRealm", International Journal of Emerging Trends In Engineering and Development, May 2012, 4(2), 64-72.
7. Tusor B., Várkonyi-Kóczy A. R., Bukur J., "A fast line extraction method based on basic segment grouping", IEEE International Instrumentation and Measurement Technology Conference (I2MTC), 2018, 1–6.
8. Yusof M.A., Kamarudin N. D., Rahayu S.B., Makhtar S.N., Mohamed H., Tajudin N., "Prototyping Digital Tongue Diagnosis System on Raspberry Pi", International Journal of Integrated Engineering, 2021, 13(5), 146-155.