Mechanical Engineering – Scientific Journal, Vol. **43,** No. 2, pp. 93–100 (2025)
Number of article: 697
Received: October 20, 2025
Accepted: November 2, 2025
In print: ISSN 1857–5293
On line: ISSN 1857–9191
UDC: [629.3.05-027.445:681.586]:712.36
https://doi.org/10.55302/MESJ2543293dm

DESIGN AND PROTOTYPING OF AUTONOMOUS ROBOTIC VEHICLE FOR PATH FOLLOWING AND OBSTACLE AVOIDANCE

Simona Domazetovska Markovska, Katerina Andonovska, Bojana Mačavelovska, Andrej Cvetanovski, Medin Ademi, Petar Janevski, Damjan Pecioski

Faculty of Mechanical Engineering, "Ss. Cyril and Methodius" University in Skopje, P.O.Box 464, MK-1001 Skopje, Republic of North Macedonia
simona.domazetovska@mf.edu.mk

A b s t r a c t: This paper presents the design and implementation of a compact robotic vehicle capable of autonomously following a predefined path through integrated sensor technologies. A Pixy2 camera detects and translates the path into vector data, enabling real-time tracking across straight and curved segments using differential steering. Motor control is achieved by modulating the speed of two DC motors via the L298N driver, guided by line position data. Communication between the camera and Arduino Uno is established through the SPI interface. An ultrasonic sensor enhances navigation by detecting and avoiding obstacles. The system halts safely at designated stop lines by setting motor PWM to zero. This project demonstrates effective hardware-software integration for autonomous navigation, combining sensor fusion, control logic, and real-time processing.

Key words: autonomous car; Pixy2; differential control; sensor integration; PID

ДИЗАЈН И ПРОТОТИПИЗИРАЊЕ НА АВТОНОМНО РОБОТСКО ВОЗИЛО ЗА СЛЕДЕЊЕ ПАТЕКА И ИЗБЕГНУВАЊЕ ПРЕЧКИ

А п с т р а к т: Овој труд се фокусира на развој на мало роботско возило способно да следи дефинирана патека користејќи интеграција на сензори. Камерата Pixy2 ја детектира линијата и ја претставува како вектор, овозможувајќи следење на патеката во реално време. Возилото е дизајнирано да следи патека што се состои од прави делови и кривини, користејќи диференцијално управување. Контролата на двата DC-мотора се постигнува со приспособување на нивните брзини, овозможувајќи прецизно управување врз основа на положбата на линијата. Камерата комуницира со Arduino Uno преку SPI интерфејсот, додека моторите се контролираат преку драјверот на моторот L298N. Дополнително се користи ултразвучен сензор за откривање и избегнување предмети на патеката. Системот е дизајниран да застане пред линијата за запирање со исклучување на моторите (PWM поставен на нула), овозможувајќи возилото безбедно да застане. Проектот ја истакнува синергијата на хардвер-софтвер, докажувајќи автономна навигација преку интеграција на сензори, алгоритми за контрола и обработка во реално време.

Клучни зборови: автономно возило; Pixy2; диференцијално управување; интеграција на сензори; PID

INTRODUCTION

This paper was developed to demonstrate a simple and cost-effective method for implementing autonomous navigation. Line-following robots are widely used in educational and research settings as they offer a practical platform for exploring control systems, sensor integration, and real-time decision-making. Additionally, they serve as foundational models for real-world applications in transportation and logistics. The primary objective of this work is to design a small robotic vehicle capable of following a predefined path using the Pixy2 camera and differential steering. The` research highlights how affordable and readily available components can be integrated to achieve effective autonomous mobility.

To contextualize this study, it is essential to examine prior research on line-following robots and autonomous navigation systems. Recent advancements in mobile robotics have been driven by the rapid development of sensor technologies, microcontrollers, and sophisticated navigation and obstacle avoidance algorithms. The following literature review summarizes various approaches to the design and implementation of mobile robotic platforms with functionalities including line following, autonomous navigation, obstacle detection, and surveillance. These implementations utilize diverse platforms such as Arduino, Raspberry Pi, and an array of specialized sensors and modules. One early design utilizes an Arduino Nano microcontroller with three infrared (IR) sensors to detect a black line on a white surface. The robot employs four DC motors controlled by an L293D motor driver and is programmed to move when a white surface is detected and stop when encountering a black one [1]. In another project, a line-following robot is developed as a mobile surveillance system. This system uses an Arduino Uno for control, NI MyRIO for wireless communication, four pairs of IR sensors for navigation, and a webcam for visual monitoring [2].

A more complex platform combines a Raspberry Pi and Arduino Uno for steering control. The robot features a fixed four-wheel chassis and is equipped with multiple sensors and a robotic arm, enabling capabilities such as mapping, autonomous navigation, and object manipulation for transportation tasks [3]. Another project incorporates two ultrasonic and two IR sensors, enabling the robot to first identify the line to follow and then detect obstacles or edges during movement. The system continues to navigate the path only when no obstacles are detected and the line remains within sensor range [4]. An additional study describes an obstacle-avoidance robot that initially operates in manual mode through a Bluetooth connection with an Android smartphone. The robot transitions to autonomous mode using ultrasonic sensors to detect and avoid obstacles in real time [5]. Meanwhile, robotic vision is explored using the Pixy2 camera, programmed via the PixyMon application to track a dominant object based on color detection [6]. Another approach applies Dijkstra's algorithm to generate an optimal offline path. The robot pauses at each node to assess the environment for obstacles before continuing its movement [7]. This study also provides an overview of twelve positioning methods, categorized into radio-frequency techniques – such as IMU, VLC, IR, ultrasonic, geomagnetic, and LiDAR – and non-radio-frequency methods, including Wi-Fi, Bluetooth, ZigBee, and RFID [8].

Further development is seen in an autonomous vehicle designed to detect and avoid obstacles dynamically. This vehicle adjusts its speed based on the presence of obstructions, optimizing its path to reach a target destination more efficiently [9]. Finally, another project presents a multi-functional robot incorporating a control module, ultrasonic sensor, line sensor, IR sensor, and Bluetooth module. This robot is capable of both autonomous line following and remote control via infrared or Bluetooth communication with a mobile phone [10].

METHODOLOGY

The methodology describes the systematic process followed in the development of the robotic vehicle. The design was approached using a black box model as shown on Figure 1 to define the overall function of the system, which was then decomposed into subfunctions shown on Figure 2 representing sensing, control, actuation, and power supply. A morphological matrix shown on Table 1 was created to evaluate alternative solutions for each subfunction, allowing a structured comparison of possible design choices. Based on this analysis, the most suitable components and methods were selected to form the final solution.

Fig. 1. Black box for the autonomous vehicle

The black box model shows the robotic vehicle as a system where energy and sensor data are inputs, and the controlled movement and stopping actions are the outputs.

To better understand the internal structure of the system, the main functions of the robotic vehicle are broken down into distinct subfunctions. Each subfunction addresses a key aspect of operation, from sensing and control to actuation and movement. Figure 2 shows the main subfunctions of the robotic vehicle and their interactions.

A morphological matrix shown on Table 1 was created to compare alternative solutions for each subfunction, helping select the most suitable components for the system.

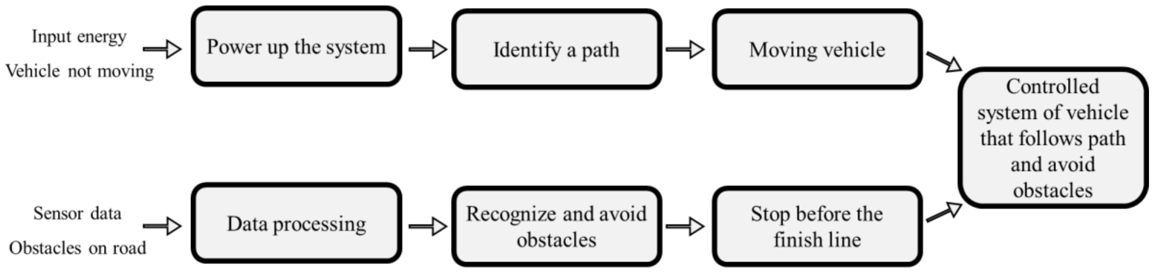

Fig. 2. Subfuntions of the system

Table 1

Morphological matrix for choosing components

Subfunctions	Executors	
1. Power up the system	1.1. Lithium batteries 	1.2. Alkaline batteries 
2. Identify a path	2.1. IR sensor 	2.2. Camera 
3. The vehicle is moving	3.1. DC motor 	3.2. Stepper motor 
4. Data processing	4.1. Arduino 	4.2. Esp32 
5. Avoid obstacles	5.1. Camera 	5.2. Ultrasonic sensor 
6. Stops before the end line	6.1. Ultrasonic sensor 	6.2. Camera 

The final solution has chosen the lithium batteries (1.1) for powering the system, camera (2.2) for identifying a path, DC motor (3.1) for movement, Arduino (4.1) microcontroller, Ultrasonic sensor (5.2) for avoiding obstacles and camera (6.2) for stopping the vehicle. In this system, the vehicle operates by moving between the white lines using sensors to detect the line position. The vehicle is powered by lithium batteries and controlled via an Arduino microcontroller, using DC motors driven through a motor driver. The Pixy2 camera detects the black line in the center and guides the vehicle along the path. The Pixy2 detects the line by representing it as vectors defined by a start point (x_0, y_0) and an end point (x_1, y_1) within the camera's field of view. These vectors are extracted from the image and represent the direction and position of the line on the path. In the code, the robot uses the horizontal positions x_0 and x_1 of the first detected vector to calculate the central position of the line using the formula $(x_0 + x_1) / 2$. This value is then compared to the ideal center of the camera's field of view, which is 39 on the horizontal axis (since the Pixy2 has an effective horizontal resolution of 78 pixels). The difference between the detected line position and the center is treated as a tracking error. This error is fed into a PID controller, which calculates a correction value to adjust the PWM signals to the left and right motors, allowing the robot to accurately follow the line by steering toward the center of the path. The system also uses an ultrasonic sensor to detect and avoid obstacles along the path. When the Pixy2 no longer detects the central line, the Arduino interprets this as the stop line being near and stops the motors.

PROTOTYPING

The prototyping phase focuses on the practical implementation of the robotic vehicle, combining all hardware components into a functional system. This chapter presents the main components used, their assembly, and the wiring diagram, including assembly drawings and renders, to illustrate how the sensors, actuators, and control system are integrated to achieve autonomous line-following and obstacle avoidance.

Components and assembly

The main components used for prototyping of the vehicles are:

– **Arduino Uno** – central microcontroller for processing sensor data and controlling motor outputs;

– **Pixy2 camera** – detects and tracks the line on the path using vector representation;

– **HC-SR04 Ultrasonic sensor** – detects obstacles to enable avoidance;

– **DC motors** – provide driving force in a differential drive configuration;

– **L298N motor driver** – controls motor speed and direction via PWM signals from the Arduino;

– **Lithium battery** – supplies stable power to all components;

– **Chassis and wheels** – provide mechanical support and mobility.

Assembly drawing and CAD models were created to visualize the placement of components on the chassis and ensure correct mechanical assembly (Figure 3). Renders of the assembled vehicle show the final design and layout (Figure 4).

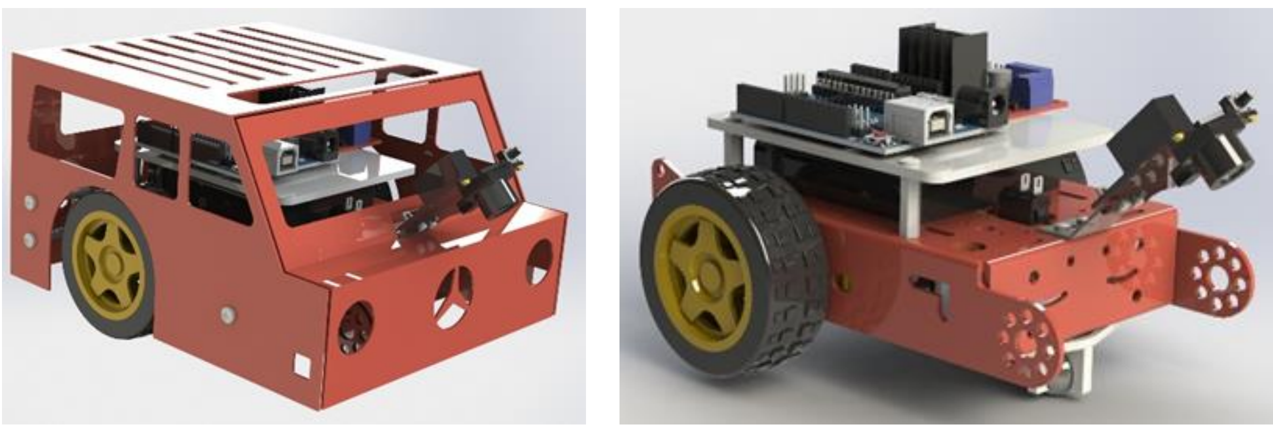

Fig. 3. Rendered photo of the vehicle design

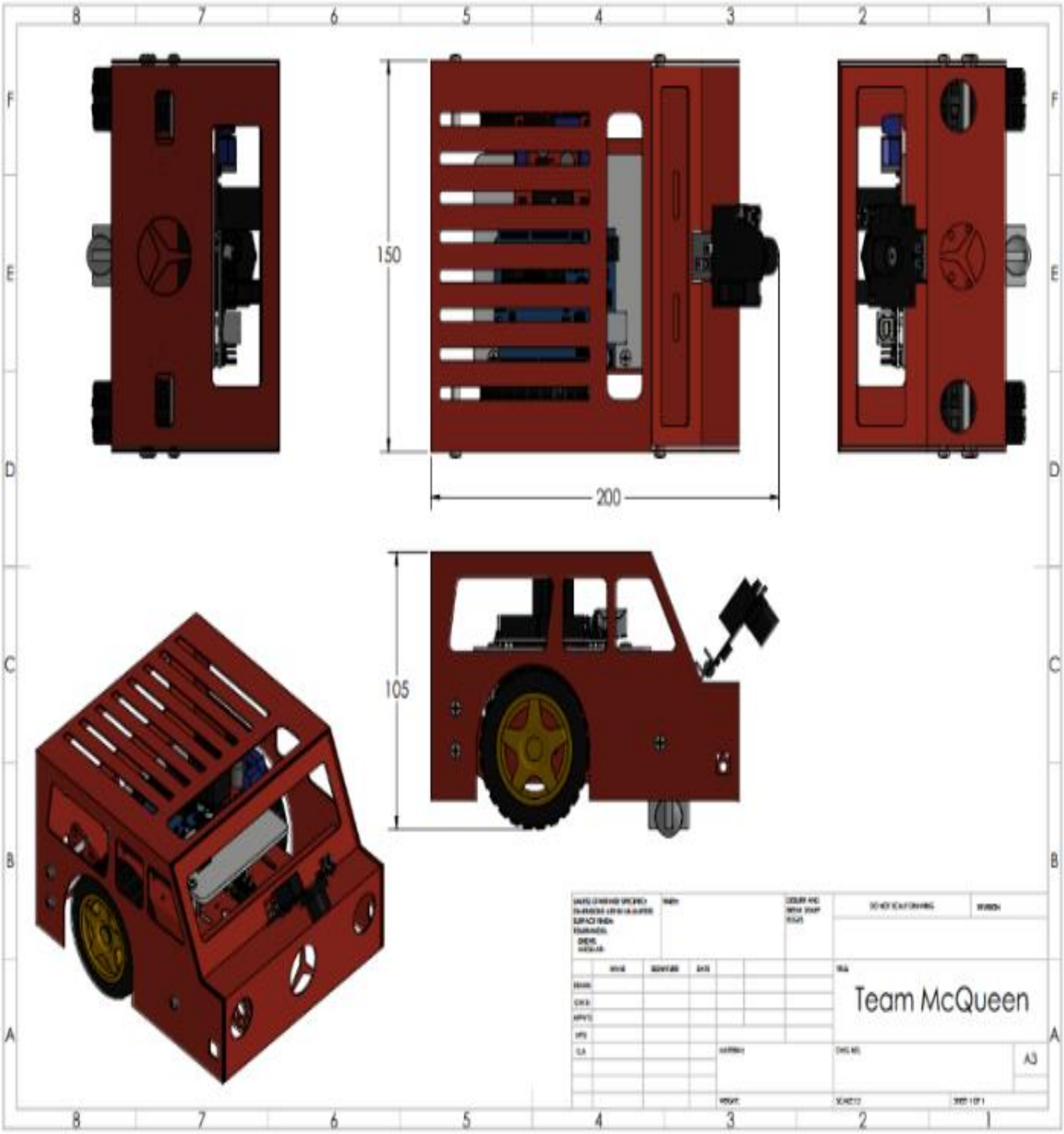

Fig. 4. Assembly drawing of the vehicle

Figure 5 illustrates the wiring diagram of the system. The Arduino Uno microcontroller serves as the central unit, coordinating signals between the sensors and the actuators. It receives input from the Pixy2 camera and the ultrasonic sensor, processes the data through the implemented code, and generates control signals for the motor driver. The motor driver regulates the power supplied from the lithium batteries to the DC motors, enabling smooth and precise motion control. This wiring setup ensures reliable communication between all components and provides the foundation for the system's programmed behaviour.

Figure 6 shows the prototype of the autonomous vehicle. The images present the overall design and assembly of the system, including the chassis, mounted sensors, wiring, and power supply. These visuals provide a clearer understanding of how the individual components are integrated into a functional unit.

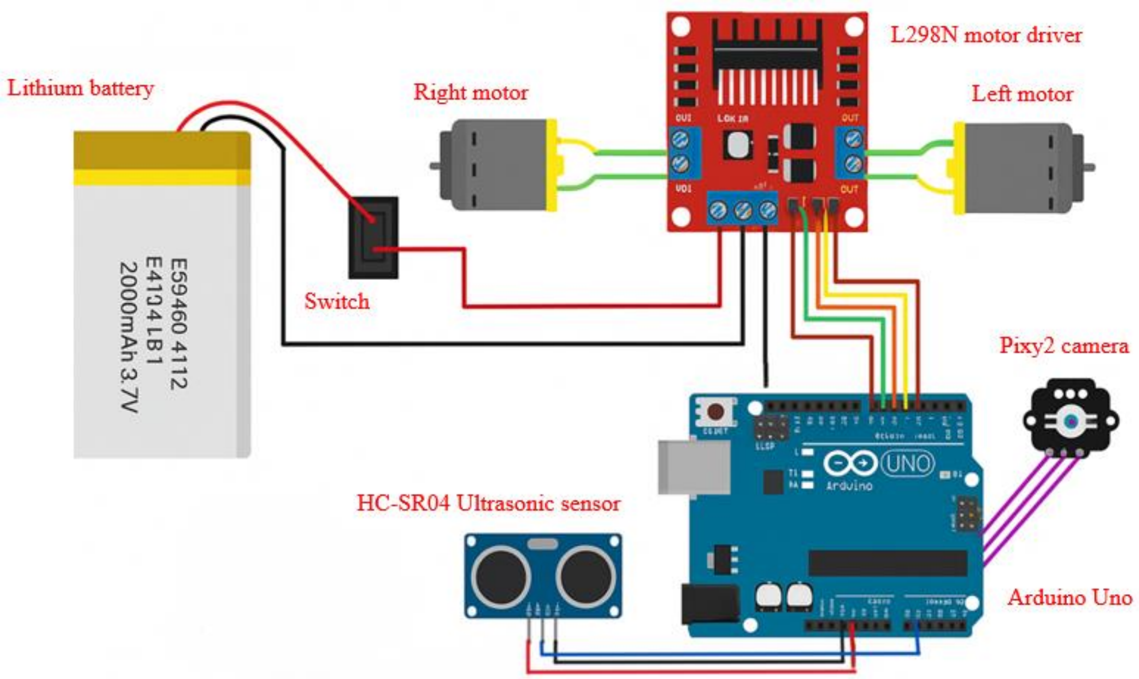

Fig. 5. System connection of the components

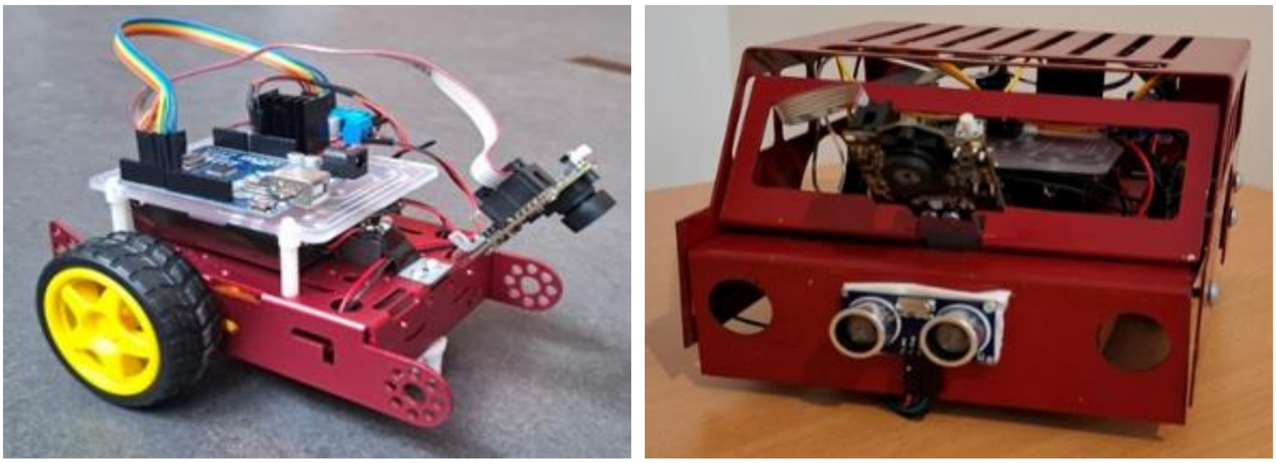

Fig. 6. Prototype of the vehicle

RESULTS

The expected results focus on assessing the robotic vehicle's functionality and stability under real-world conditions. It is anticipated that the vehicle will successfully follow a predefined line using the Pixy2 camera, which detects the black line in real time and represents it as a vector. The camera determines the vector's position and compares it to the center of its field of view (value 39). The deviation from the center generates an error value, which is used by a PID (Proportional–Integral–Derivative) controller to calculate a correction. This correction dynamically adjusts the speeds of the left and right DC motors, allowing the robot to maintain accurate and stable line tracking.

The PID controller uses three components – proportional, integral, and derivative – to respond to changes in the error signal. The proportional term responds to the current error, the integral accounts for accumulated past errors, and the derivative anticipates future trends based on the rate of error change. The controller is configured with the following gains: the proportional gain is 2, the integral gain is 0, while the derivative gain is 1.

Based on the correction value, the motor speeds are adjusted as follows: when turning left, the left motor slows down, and the right motor speeds up, while when turning right, the right motor slows down, and the left motor speeds up.

To regulate motor speed, the system uses Pulse Width Modulation (PWM), a technique that controls the effective power delivered to the motors by rapidly switching the voltage on and off. Arduino implements PWM with 8-bit resolution, meaning the PWM value can range from 0 to 255. This range represents the duty cycle, where a value of 0 equates to a 0% duty cycle (no power) and 255 equates to 100% (maximum power). For forward motion, both motors are set to a base PWM value of 150, corresponding to a duty cycle of approximately 58.82%.

The vehicle's turning and movement are achieved through **differential steering**. Depending on the correction value from the PID controller, one motor's speed increases while the other decreases, enabling smooth and precise directional changes. To ensure correct operation, PWM values are constrained within the valid range from 0 to 255.

In addition to line tracking, the system includes **obstacle detection** using an HC-SR04 ultrasonic sensor. This sensor emits ultrasonic pulses and measures the time it takes for the echo to return after hitting an object. The distance to an object is determined by measuring the time it takes for an ultrasonic pulse to travel to the object and reflect back to the sensor. Using the known speed of sound (approximately 343 meters per second), the system calculates how far the object is based on the duration of this round-trip. Based on the measured distance, the vehicle makes decisions accordingly:

If the object is more than 20 centimeters away (considered a safe distance), the vehicle continues following the line using the Pixy2 camera and PID control.

If the object is 20 centimeters away or closer, the vehicle stops line following and activates obstacle avoidance mode.

In obstacle avoidance mode, the vehicle first comes to a brief stop to ensure safety. It then performs a turning maneuver to change its direction. After turning, the ultrasonic sensor scans the new path to check for any obstacles. If the path is clear, the vehicle continues moving forward.

The observed results show that the vehicle successfully follows the predefined path with high precision, using real-time data from the Pixy2 camera and dynamic adjustments from the PID controller. When the line is centered, both motors receive equal PWM signals, and the robot moves straight. When the line deviates from center, the PID controller modifies the motor speeds, accordingly, allowing for smooth and accurate turns. The system handles both straight and curved segments effectively, without noticeable oscillations or delays.

When the Pixy2 camera can no longer detect the lines such as at the end of the track the system shuts off both motors by setting their PWM values to zero, ensuring a safe and controlled stop. Additionally, the HC-SR04 sensor reliably detects obstacles within 20 cm and enables the robot to avoid them, regardless of the object's shape, color, or transparency. This robustness makes the system suitable even in challenging environmental conditions like dust or fog.

Overall, the results validate the integration of the Pixy2 camera, PID-based motor control, and ultrasonic sensing as a reliable and stable solution for autonomous navigation. The combination of accurate line following and responsive obstacle avoidance demonstrates the system's successful both in terms of mechanical design and software functionality.

CONCLUSION

This project successfully demonstrated the development of a compact autonomous robotic vehicle capable of following a predefined path while avoiding obstacles. The integration of the Pixy2 camera with a PID control algorithm enabled precise, real-time line tracking, allowing smooth navigation through both straight and curved sections. The ultrasonic sensor provided reliable obstacle detection and avoidance, ensuring safe operation under dynamic conditions. Testing confirmed that the system responds quickly to changes in the line position, maintains stable operation, and stops accurately at the end of the path. Overall, the results highlight the effectiveness of combining sensor integration, control algorithms, and real-time data processing to achieve robust and precise autonomous navigation in a compact robotic platform.

REFERENCES

[1] Saw, K. K., Mon, L. Y. (2019): Design and construction of line following robot using Arduino. *International Journal of Trend in Scientific Research and Development*, 3(4), pp. 939–941.

[2] Saad, W. H. M., Karim, S. A. A., Azhar, N., Manap, Z., Soon, Y. Y., Ibrahim, M. M. (2018): Line follower mobile robot for surveillance camera monitoring system. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, **10** (2–7), pp. 1–5.

[3] Oltean, S. E. (2019): Mobile robot platform with Arduino Uno and Raspberry Pi for autonomous navigation. *Procedia Manufacturing*, **32**, pp. 572–577.

[4] Barua, V., Neloy, M. A. I., Rahat, S. U., Das, M., Joy, M. S. I., Nahar, N., Pathak, A. (2020): An ultrasonic line follower robot to detect obstacles and edges for industrial and rescue operations. *International Journal of Computer Applications*, **176** (31), p. 31.

[5] Vaishnavi, M. S. (2023): Obstacle Avoiding Bluetooth Robot Car using the Internet of things. *Indian Journal of Data Communication and Networking (IJDCN)*, **3** (2), pp. 11–14.

[6] Wahrini, R. (2023): Development of color-based object follower robot using Pixy 2 camera and Arduino to support robotics practice learning. *Jurnal Edukasi Elektro*, **7** (2), 161–168.

[7] Alshammrei, S., Boubaker, S., Kolsi, L. (2022): Improved Dijkstra algorithm for mobile robot path planning and obstacle avoidance. *Comput. Mater. Contin*, **72** (3), 5939–5954.

[8] Huang, J., Junginger, S., Liu, H., Thurow, K. (2023): Indoor positioning systems of mobile robots: A review. *Robotics*, **12** (2), 47.

[9] Patil, N., Badole, Y., Sharma, P., Shrivas, P., Kumar, R., Nema, N. K. (2024): Using an Arduino Nano, an autonomous vehicle system for speed control. *International Research Journal of Journal of Modernization in Engineering Technology and Science* **6** (3), 3532–3536. DOI:10.56726/IRJMETS50979

[10] Idhom, M., Budijanto, A., Mufti, N., Alamsyah, M. R., Kristiawan, K. Y., Arinda, U. Y. (2022): Making a mobile educational robot with a practical approach using Arduino. *Nusantara Science and Technology Proceedings*, 253–262.