sciendo

VOLUME 18, N° 1 2024
Journal of Automation, Mobile Robotics and Intelligent Systems

Low-Cost Small-Scale Autonomous Vehicle

Submitted: 11th July 2023; accepted: 10th October 2023

Ismail Bogrekci, Pinar Demircioglu, Mustafa Yasir Goren

DOI: 10.14313/JAMRIS/1-2024/3

Abstract:
A low-cost small-scale autonomous vehicle refers to a self-driving vehicle that is designed to be affordable and suitable for smaller applications or specific purposes. In this study, the firefly algorithm was utilized to address obstacle avoidance challenges in the presence of dynamic or statically positioned uncertain obstacles. The autonomous vehicle successfully reached the intended destination, demonstrating a satisfactory level of accuracy. Regardless of the starting point, the vehicle arrived at the predetermined position within an area measuring 5 meters in diameter. The achievement of such results can be attributed to the cost-effective selection of sensors, utilization of a simple algorithm, and the implementation of a moderately powered processor and circuit components.

Keywords: *Autonomous drive, Unmanned ground vehicle, Sensors, Firefly algorithm*

1. Introduction

The significance of low-cost small-scale autonomous vehicles in various domains and applications is noteworthy. Several key reasons contribute to their value:

Accessibility: Low-cost small-scale autonomous vehicles increase the accessibility of autonomous technology to a wider range of users. This eliminates financial barriers and enables individuals, researchers, hobbyists, and small businesses to explore and experiment with autonomous systems.

Education and Research: Small-scale autonomous vehicles provide a practical and hands-on platform for educational institutions, researchers, and students to engage in learning and conducting experiments in fields such as robotics, artificial intelligence, control systems, and computer vision. They facilitate the study of autonomous vehicle algorithms, behavior, and sensor integration within controlled environments.

Testing and Prototyping: Small-scale autonomous vehicles are well-suited for testing and prototyping new algorithms, software, and hardware components. They enable developers to validate their ideas, perform simulations, and gather real-world data on a smaller and more manageable scale before transitioning to larger and more expensive platforms.

Innovation and Entrepreneurship: Low-cost small-scale autonomous vehicles foster innovation and entrepreneurship by empowering individuals and startups to develop new applications and services based on autonomous technology. They serve as a foundation for building proofs-of-concept and minimum-viable products in industries such as delivery services, agriculture, surveillance, and environmental monitoring.

Skill Development: Engaging with low-cost small-scale autonomous vehicles presents an opportunity for individuals to develop skills in areas such as programming, robotics, sensor integration, and system integration. This facilitates the growth of a talent pool comprising autonomous system developers and professionals who contribute to the advancement of the field.

Safety and Testing Grounds: Small-scale autonomous vehicles can serve as testing grounds for evaluating and refining autonomous systems and safety protocols before real-world deployment. They provide controlled environments for identifying and addressing potential risks and challenges without compromising safety.

Technological Advancement: The development and adoption of low-cost small-scale autonomous vehicles drive technological advancements in sensor technology, artificial intelligence, machine learning, and computer vision. This fosters innovation and pushes the boundaries of autonomous systems, resulting in improved efficiency, reliability, and performance.

In summary, low-cost small-scale autonomous vehicles play a crucial role in democratizing autonomous technology, promoting education and research, facilitating innovation and entrepreneurship, and advancing the field of autonomous systems as a whole. They serve as stepping stones for individuals and organizations to explore, experiment, and contribute to the growing ecosystem of autonomous vehicles and related applications.

2. Literature Review

Rapid advances in autonomous vehicle (AV) technology are expected to bring about a transformation in transportation habits. Despite their limited presence on the road, public preferences, acceptance, and adoption intentions related to AVs have been the subject of investigation by a growing body of research [1]. Autonomous vehicle literature reviews provide insights on tech, control, sensors, human

Open Access. © 2024 Ismail Bogrekci et al., published by Sciendo. BY-NC-ND This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License

factors, security, and privacy, informing research's current state and future directions.

The acceptability of different autonomous vehicle behaviors in conflicts depends on various factors like societal norms, legal constraints, and ethical frameworks. Understanding these influential factors is crucial for creating effective guidelines and policies. Future research can explore specific aspects like ethics, safety algorithms, real-time decision-making, and human-machine interfaces in more depth [2].

The long-term effects of autonomous vehicles on the built environment have gained significant attention due to the potential transformative impact of this technology. Developing conceptual frameworks to study the long-term effects of autonomous vehicles on the built environment requires an interdisciplinary approach. Incorporating elements from urban planning, transportation engineering, environmental science, social sciences, and public policy can provide a comprehensive understanding of the complex interactions and potential consequences [3].

Designing and developing the software stack of an autonomous vehicle using the Robot Operating System (ROS) in conjunction with hardware modules responsible for controlling the car requires careful integration between software and hardware components. Throughout the development process, it is essential to consider safety, reliability, and system redundancy. Implement mechanisms to handle sensor failures, communication errors, and emergency situations. Adhere to safety guidelines and regulatory requirements to ensure the autonomous vehicle operates safely and complies with applicable laws. Additionally, consider leveraging existing ROS packages, libraries, and tools that provide functionalities for sensor integration, actuator control, and planning and control algorithms. The ROS ecosystem offers numerous resources that can accelerate development and provide a solid foundation for autonomous vehicle software stacks [4].

As AV technology evolves, there is a possibility that traffic lanes and on-street parking spots could be downsized or reconfigured to accommodate the efficiency and safety features of AVs. This downsizing could result in the availability of additional spare road space in future urban streets. It is essential for urban planners, policymakers, and communities to proactively consider the potential repurposing of spare road space as AV technology advances. Through careful planning and collaboration, cities can leverage this opportunity to create more livable, sustainable, and people-centric urban environments [5].

Research on path planning for autonomous vehicles based on the Frenet system has gained significant attention in recent years, providing a mathematical framework for describing the motion of a particle along a curve in three-dimensional space. It is particularly useful for path planning in autonomous vehicles as it allows for efficient trajectory generation and control. The road behavior of a car was simulated using a five-fold polynomial algorithm model, which allows for the generation of path trajectories that mimic different driving behaviors.

By analyzing the rate of change of lateral and vertical velocity, as well as lateral and vertical acceleration under various behaviors, it became possible to estimate the prediction time for the car [6].

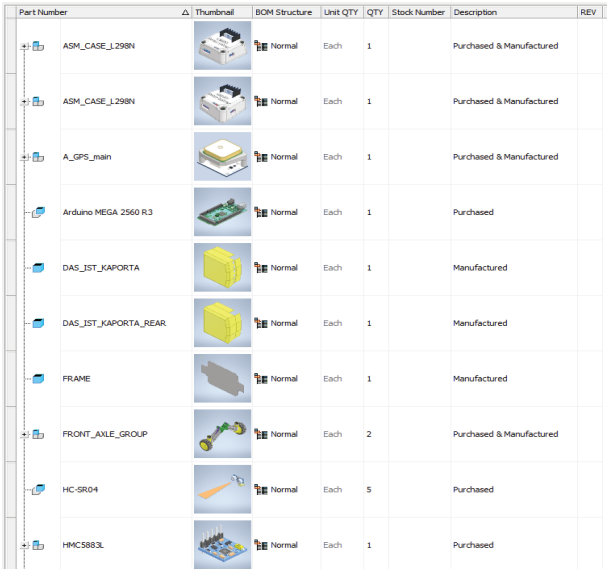
Autonomous vehicles rely on a combination of sensors to perceive their surroundings and make informed decisions. In this review, a list of sensors like LiDAR (Light Detection and Ranging), Radar (Radio Detection and Ranging), and Cameras (RGB, monocular, stereo, or multi-camera setups) commonly used in autonomous vehicles are explained in detail [7].

Offline mapping for autonomous vehicles with low-cost sensors gave a feasible approach, especially when high-precision mapping data was not a strict requirement [8]. While low-cost sensors may not offer the same level of accuracy as high-end sensors, they can still provide valuable data for basic mapping purposes.

Vision-based navigation and guidance systems offer numerous benefits in agricultural applications, including increased efficiency, reduced labor requirements, improved accuracy, and optimized resource utilization. Ongoing advancements in computer vision, machine learning, and robotics continue to enhance the capabilities and reliability of these systems in the agricultural sector [9].

Map-based localization methods using 3D-LiDAR (Light Detection and Ranging) sensors have proven to be effective in providing accurate and robust localization for autonomous vehicles. By leveraging the rich spatial information captured by 3D-LiDAR sensors, these methods enable vehicles to determine their position within a pre-built map. Curb-map-based localization leverages the unique characteristics of curbside features, which are relatively stable and distinguishable in urban environments. By focusing on curbs and associated features, this approach can provide precise and reliable localization, even in challenging scenarios with limited GPS availability or complex road layouts. It is important to note that curb-map-based localization may be used in combination with other sensor inputs, such as GPS, IMU, or camera data, to enhance the overall localization accuracy and robustness [10].

Nonetheless, the full-scale deployment of autonomous vehicles continues to face significant obstacles concerning safety concerns. These concerns stem from a range of issues within the vehicles themselves and external factors in their operational environments. Addressing these safety challenges is imperative. Sensor data plays a critical role in this endeavor by providing valuable insights into the current operational status of autonomous vehicle systems and the influence of external environmental factors. Such data helps in monitoring and mitigating risks, contributing to the overall safety and reliability of autonomous vehicle technology [11, 12].



Part Number	Thumbnail	BOM Structure	Unit QTY	QTY	Stock Number	Description	REV
ASM_CASE_L298N		Normal	Each	1		Purchased & Manufactured	
ASM_CASE_L298N		Normal	Each	1		Purchased & Manufactured	
A_GPS_main		Normal	Each	1		Purchased & Manufactured	
Arduino MEGA 2560 R3		Normal	Each	1		Purchased	
DAS_IST_KAPORTA		Normal	Each	1		Manufactured	
DAS_IST_KAPORTA_REAR		Normal	Each	1		Manufactured	
FRAME		Normal	Each	1		Manufactured	
FRONT_AXLE_GROUP		Normal	Each	2		Purchased & Manufactured	
HC-SR04		Normal	Each	5		Purchased	
HMC5883L		Normal	Each	1		Purchased	

Figure 1. Bill of materials (main assemblies and parts)

3. Materials and Methods

In the Materials and Methods section, the study examines each stage and result separately, encompassing equipment, algorithms, and methods.

Firstly, the bill of materials is provided in Figure 1. When selecting components, considerations are given to factors such as cost, functionality, compatibility, and ease of access to resources, aiming for successful implementation in the experimental research. The materials are described with general information, including visuals of the components, technical specifications, and brief comments based on the conducted experiments and studies.

Furthermore, in the method section, the design stages of the vehicle are explained, including calculations and the software algorithm for autonomous driving.

3.1. Materials

The L298N Motor Driver Unit is considered an optimal motor driver module for driving DC and Stepper Motors. It is composed of a 78M05 5V regulator and L298 motor driver. With the L298N Module, up to 4 DC motors or 2 DC motors with direction and speed control can be operated. In the experimental study, two of them were used as directional and speed controllers, resulting in the control of two wheels by 1 L298N motor driver.

The U-Blox GY-NEO6MV2 GPS unit is a GPS module that integrates a U-Blox NEO-6M GPS receiver with an external antenna.

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 microcontroller, responsible for managing all the components of the entire circuit and the functions expected from the autonomous drive. The selection of the Arduino Mega 2560 was driven by the ease of accessing code sources, the cost of the microcontroller unit, and its compatibility with other circuit members.

The HC-SR04 is an ultrasonic distance sensor commonly utilized in various applications, including robotics, automation, and proximity sensing. The ultrasonic measuring module HC-SR04 technically is able to measure a range of 20mm to 4000mm due to its technical data sheet provided by the manufacturer, with a range accuracy that may reach up to 3mm. The modules consist of ultrasonic sound receivers, transmitters, and controlling circuits. The essential working principle involves using the IO trigger to send a high-level signal for approximately 10us. The sensor emits an 8, 40kHz ultrasonic pulse and detects the return signal pulse. The duration of the high output IO time, when the signal returns, corresponds to the time taken from ultrasonic transmission to reception.

The HMC5883L is a magnetometer sensor designed to measure magnetic field strength and direction. It is commonly employed in applications such as navigation, robotics, and magnetometer calibration.

The 6V DC brushed motor with a reducer and wheel represents a standard configuration utilized in this experimental study.

3.2. Methods

Firstly, when determining the basic dimensions of the vehicle, careful consideration was given to elucidate how the decisions were made. One of the main factors taken into account was the manufacturability of the parts, which led to the realization that 3D printing was the most suitable option for implementing the experimental car. Consequently, this set a constraint on the design to keep the dimensions as compact as possible. Additionally, the functionality of the sensors and electromechanical parts posed further considerations that necessitated longer dimensions. Ultimately, considering these various conditions, the dimensions of the autonomous car were determined.

In Figure 2, it can be observed that the wheels have been positioned at a 5-degree camber angle. The reason for the selection of a positive 5-degree camber angle is to ensure that the tire remains connected to its reducer without the need for fastening or gluing. Furthermore, when the shock absorbers are compressed, the camber angle changes to a negative value if it was initially set at zero degrees. However, if the camber angle is set to a positive value, even when loaded at full capacity, the camber angle will remain positive or zero, thereby maintaining vehicle dynamics and drive stability.

Path planning algorithms play a critical role in autonomous vehicles' ability to navigate safely, efficiently, and adaptively in dynamic environments.

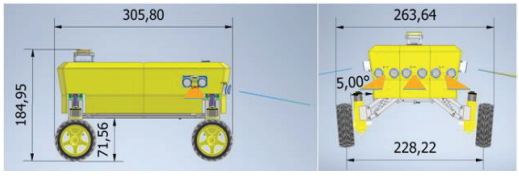


Figure 2. Basic dimensions of the autonomous vehicle (dimensions in mm)

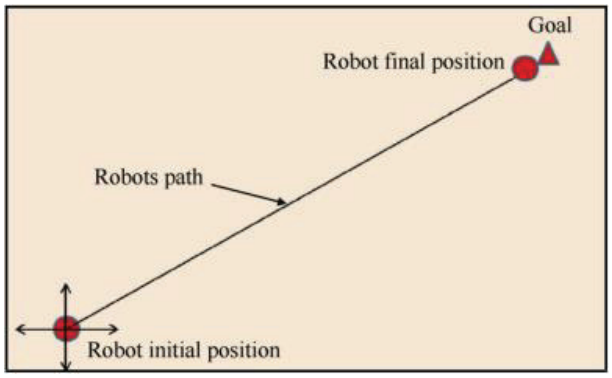


Figure 3. Shortest line between initial and final position of autonomous vehicle (AV)

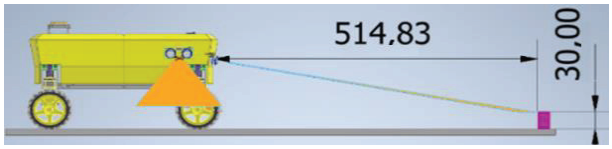


Figure 4. Autonomous vehicle faces with an obstacle. (dimensions in mm)

These algorithms optimize trajectories, ensure collision avoidance, handle complex maneuvers, consider human interaction, and contribute to overall efficiency and reliability. The path planning algorithm functions by determining the shortest line between the starting point and destination point, as illustrated in the figure below. Sensors 2, 3, and four serve the same purpose as shown in Figure 8, but the placement of three sensors extends the range of obstacle detection.

By the application of the Firefly Algorithm (FA) to path planning, the search space of possible paths can be explored by the autonomous vehicle. These paths are evaluated based on their fitness values and iteratively refined to find optimal or near-optimal solutions, considering the defined objective function.

It should be emphasized that the performance and suitability of the Firefly Algorithm as an optimization technique for path planning depend on the specific problem, environment, and objectives. To assess and adapt the Firefly Algorithm for optimal results in a given application scenario, comparisons with other path planning algorithms and careful parameter tuning are necessary.

During the calculation of positions and bearing degrees, the shortest route is generated to navigate toward the destination. In the event of an obstacle detected by the sensors at a distance shorter than 520mm, the vehicle will steer in another direction to bypass the obstacle. After placing sensors at 10degree angle between ground and sensors normal axis, ideal distance had been calculated as 514.83mm to detect 30mm obstacle. That 514.83mm distance had been rounded to 520mm by self-decision due to make the calculations easier and gaining additional safer distance 5.17mm. At this point, the Firefly Algorithm (FA) loop is initiated to navigate around obstacles within the shortest distance. In Figure 4, the vehicle's

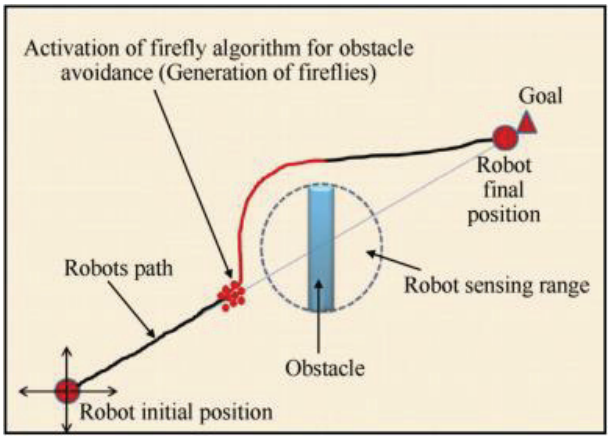


Figure 5. Activation of firefly algorithm

obstacle definition is illustrated, which is applicable to objects with a height greater than 30mm, considering their dynamic capabilities.

The condition for one of the sensors is set at a safe distance of 520mm (which is an optimized value) to ensure that obstacles are not approached. In situations where this condition is not met, the AV breaks its heading loop and initiates the FA algorithm loop to generate fireflies in the vicinity of the obstacles, as demonstrated in Figure 5.

Several random fireflies are produced and positioned near the obstacles, and the brighter fireflies are selected from this group. Brighter fireflies refer to new starting points that provide the maximum safe distance between the obstacle and the firefly. The AV maintains this generation and selection process of the brightest firefly until it successfully avoids obstacles that can be detected by the ultrasonic sensor distances.

The flowchart of the Firefly Algorithm is explained in Figure 6, and the pseudo code of the Firefly Algorithm is provided in Figure 7.

Before the application of the Firefly Algorithm (FA) formulation to the software of the autonomous vehicle (AV), the AV's response to encountering an obstacle can be summarized in the following steps:

Initialization: The AV and the target position are initialized.

Calculation of Heading Degree: The heading degree between the AV's current GPS position and the target position is calculated.

Obstacle Detection: During the heading phase, if an obstacle is detected by the AV, the heading loop is interrupted, and the FA loop is activated.

Firefly Population Generation: A population of fireflies is generated by the AV in the vicinity of the detected obstacle.

Brightest Firefly Selection: The AV employs a fitness equation to select the brightest firefly from the generated population.

Heading Loop Activation: The AV activates the heading loop again to calculate a new route between the brightest firefly's position and the target position.

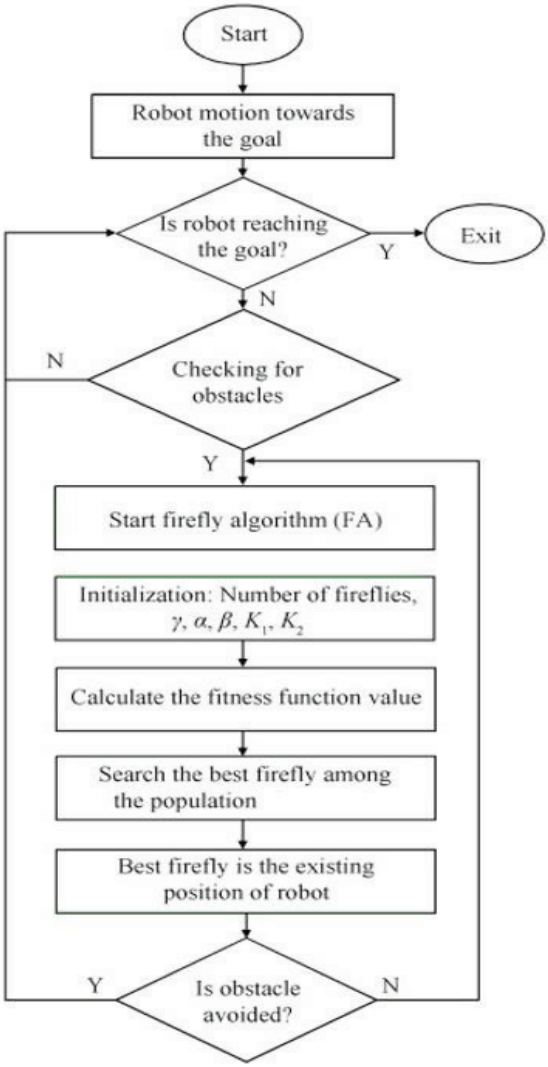


Figure 6. Flowchart of firefly algorithm

Objective function $f(x)$, $x=(x_1, x_2,\ldots\ldots,x_d)^T$
Generate initial population of fireflies, x_i (i=1, 2, 3,……, n)
Light intensity I_i at x_i is determined by $f(x_i)$
Define light absorption coefficient γ
while (t< Max generation)
for i=1: n all n fireflies
  for j=1: n all n fireflies (inner loop)
  if (I_i<I_j), Move firefly i towards j;
  end if
    Vary attractiveness with distance r via exp $[-\gamma r^2]$
    Evaluate new solution and update light intensity
  end for j
end for i
Rank the fireflies and find the current global best
end while

Figure 7. Pseudo code of firefly algorithm

Navigation with Obstacle: The AV proceeds to move toward the target position while encountering an obstacle. If the AV encounters another obstacle during this process, steps 3 to 6 are repeated to ensure obstacle avoidance.

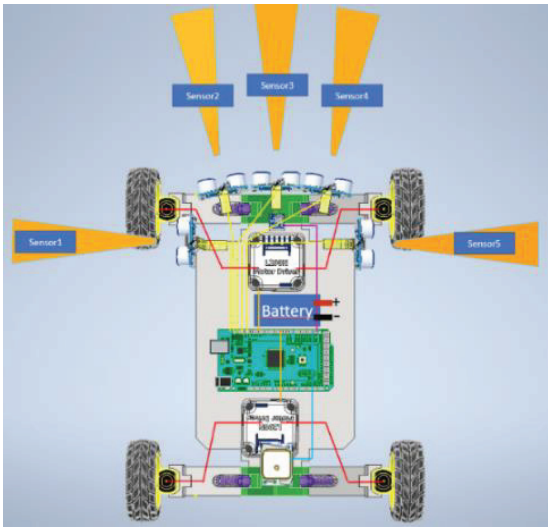


Figure 8. Circuit diagram of AV

By following these steps, the AV dynamically adjusts its path and navigates toward the target while effectively avoiding obstacles encountered along the way.

$$D_{fo} = \sqrt{(x_0 - x_{f_i})^2 + (y_0 - y_{f_i})^2} \quad (1)$$

The Euclidean distance, referred to as Dfo, between the position of a firefly and a nearby obstacle is a crucial parameter in the algorithm. In this study, the Dfo values are obtained through sensor readings, specifically from five sensors (see Fig. 8). These sensor readings serve as the calculated Dfo values, as represented in Equation (1).

$$D_{fg} = \sqrt{(x_g - x_{f_i})^2 + (y_g - y_{f_i})^2} \quad (2)$$

Dfg is the Euclidean distance between firefly and target point shown in Eq. (2).

$$f_i = K_1 \cdot \frac{1}{\min o_n \in o_s \| D_{fo} \|} + K_2 \cdot \| D_{fg} \| \quad (3)$$

The calculation of the firefly's path optimization, denoted as fi, is performed using the formula provided in Equation (3). In this equation, K1 represents a parameter that signifies the safety level of the path, while K2 denotes a parameter defining the maximum and minimum path lengths for routing. After the positions and parameters are computed and set by the user, the microcontroller unit (MCU) initiates the execution of the algorithm described in Equation (3), subsequently determining the optimal firefly.

Finally, as illustrated in Figure 8, the autonomous vehicle (AV) calculates a new route between the initial point (selected firefly by the MCU) and the final point (target set by the user).

The library includes all the necessary function keys for user-defined operations. Additionally, a math library has been incorporated to perform calculations related to positions detected by satellites. After configuring the power and signal pins of the GPS module, a void GPS loop has been implemented, as depicted in Figure 9.

```
void GPS(){
 while(serial_connection.available()){
       gps.encode(serial_connection.read());
     }
       if(gps.location.isUpdated()){
       Serial.println("Satellite Count:");
       Serial.println(gps.satellites.value());
       Serial.println("Lattitude:");
       Serial.println(gps.location.lat(),6);
       Serial.println("Longtitude:");
       Serial.println(gps.location.lng(),6);
        Serial.println("Speed MPH");
        Serial.println(gps.speed.mph());
         Serial.println("Altitude Feet:");
          Serial.println(gps.altitude.feet());
           Serial.println();
     }

 a1 = (gps.location.lat())*PI/180;  // lat radian
 b1 = (gps.location.lng())*PI/180;  // lng radian
 a2 = 37.853706*PI/180 ; //  Hedef lat radian
 b2 = 27.854791*PI/180 ; // Hedef lng radian

R =  6371000;
 m = (sin((a2-a1)/2)*sin((a2-a1)/2))+(cos(a1)*cos(a2)*sin((b2-b1)/2)*sin((b2-b1)/2));
  c = 2*atan2((sqrt(m)),(sqrt(1-m)));
    D = R*c;

 y = sin(b2-b1)*cos(a2);
 x = cos(a1)*sin(a2) - sin(a1)*cos(a2)*cos(b2-b1);
 brng = atan2(y,x)*(180/PI) ;
 int istikamet = brng;
 if(istikamet < 0){
  brng+=360;
 }
}
```

Figure 9. Void GPS loop

```
  // Calculate heading
  float heading = atan2(norm.YAxis, norm.XAxis);

  // Set declination angle on your location and fix heading
  // You can find your declination on: http://magnetic-declination.com/
  // (+) Positive or (-) for negative
  // For Bytom / Poland declination angle is 4'26E (positive)
  // Formula: (deg + (min / 60.0)) / (180 / PI);
  float declinationAngle = (4.0 + (55.0 / 60.0)) / (180 / PI); // İzmir'in Sapma değeri 4derece 55 dakika olarak
hesaplandı.
  heading += declinationAngle;

  // Correct for heading < 0deg and heading > 360deg
  if (heading < 0){
    heading += 2 * PI;
  }

  if (heading > 2 * PI){
    heading -= 2 * PI;
  }

  // Convert to degrees
  float headingDegrees = heading * 180/PI;

  /* Output
  Serial.print(" Heading = ");
  Serial.print(heading);
  Serial.print(" Degrees = ");
  Serial.print(headingDegrees);
  Serial.println();
   delay(100); */
  return( int(headingDegrees));
```

Figure 10. Bearing degree calculation

For the bearing degree calculations of the magnetometer, formulas were written in the software editor program of Arduino. Magnetic declination is also considered while calculating the bearing degree of the vehicle (Fig. 10).

The vehicle operates using two main loops. The first loop is responsible for calculating the heading, bearing degree, and position of the vehicle, allowing it to navigate toward a desired destination. As the vehicle approaches the destination, the heading degree is recalculated based on geometric rules to ensure accuracy.

The second loop is dedicated to obstacle avoidance. The vehicle relies on sensors as its "eyes" to detect obstacles. If the distance between the vehicle and an obstacle falls below the predefined safe distance of 520mm, the microcontroller interrupts the main loop and switches to the obstacle avoidance loop. In this mode, the microcontroller provides directives to steer the vehicle and prevent collisions until a safe distance is maintained. Once all the sensors detect and confirm the absence of obstacles along the vehicle's path, the microcontroller transitions back to the loop that directs and guides the vehicle toward its designated destination.

The experiment was designed to be conducted in outdoor conditions, involving obstacles with five different geometric shapes constructed from cardboard. These specific shapes were chosen to evaluate the vehicle's ability to avoid obstacles. The figures below (Figure 11(a)–11(e)) illustrate the five distinct geometric objects used in the experiment.

The five different shapes were selected based on their level of difficulty for detection by ultrasound sensors. These shapes will be positioned in various orientations during the ten measurements of the vehicle's performance. The crenel-shaped obstacle, in particular, was chosen due to its complex structure, presenting a significant challenge for ultrasound sensors.

The vehicle's rated speed has been measured as 0.5 meters per second. To evaluate its speed capabilities, a speed test was conducted on the street, covering a distance of 2.5 meters within 5 seconds. By applying the formula speed = distance ÷ time, a speed of 0.5 m/s was calculated.

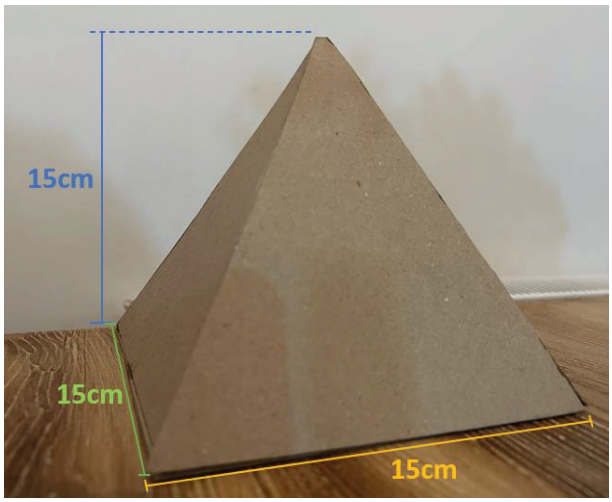


Figure 11(a). Pyramid shaped obstacle

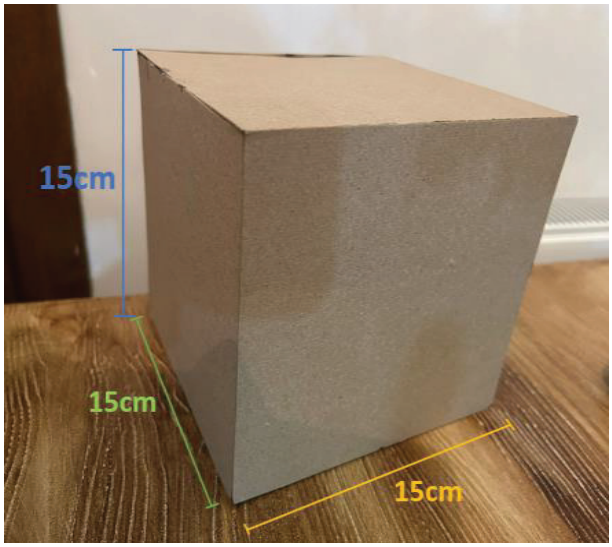


Figure 11(b). Cube shaped obstacle

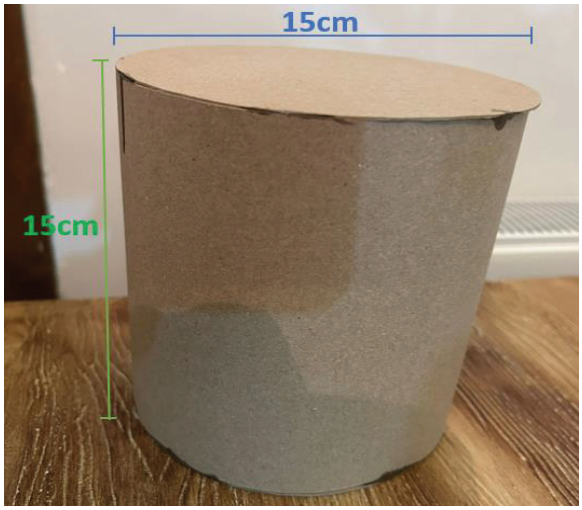


Figure 11(c). Cylindrical shaped obstacle

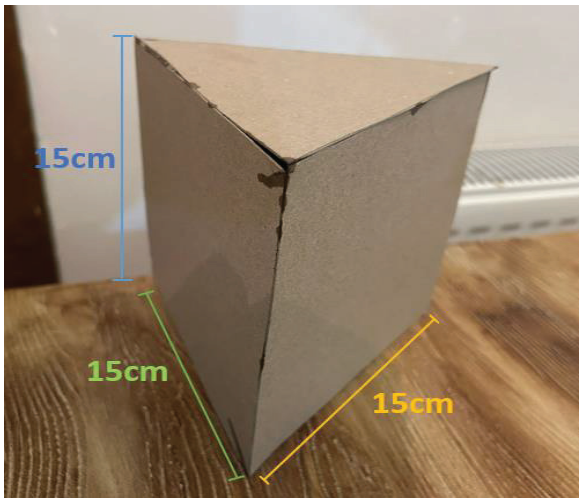


Figure 11(d). Triangular prism shaped obstacle

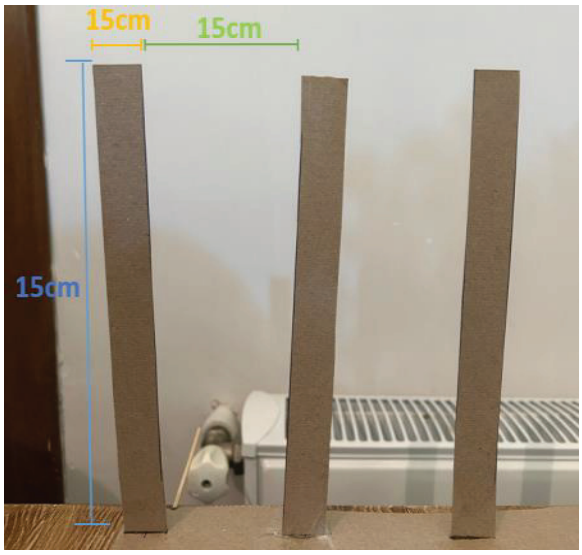


Figure 11(e). Crenel-shaped obstacle

Figure 12. Obstacle avoidance outdoor test

```
Heading (degrees): 336.45
X: -33.55  Y: -35.00  Z: -22.76  uT
Heading (degrees): 336.22
X: -33.36  Y: -34.91  Z: -22.86  uT
Heading (degrees): 336.30
X: -33.36  Y: -35.00  Z: -22.86  uT
Heading (degrees): 336.37
X: -33.18  Y: -35.09  Z: -22.55  uT
Heading (degrees): 336.60
X: -33.36  Y: -34.91  Z: -22.86  uT
Heading (degrees): 336.30
X: -33.55  Y: -35.09  Z: -22.65  uT
Heading (degrees): 336.29
X: -33.36  Y: -35.00  Z: -22.76  uT
Heading (degrees): 336.37
```

Figure 13. Heading degree experimental sensor readings

While performing the small-scale autonomous vehicles outdoor test, the temperature of the sunny day was 24 degrees Celsius, the humidity was %52, and the wind speed was 12 km/h, 1018 hPa, and during the test, the values were almost exactly the same due to the short test durations. In addition to that, air currents are important for measurement accuracy; however, air currents must be at serious levels, such as in stormy weather, which has speeds over 60 km/h. During test day, the wind speed was 12 km/h, which is a fair level of air current that would be neglected for accuracy.

Based on the information provided in Figure 12, the initial position of the autonomous vehicle (AV) was determined using GPS readings as 38.516534, 27.044097. The target position for the AV was selected as 38.516808, 27.043947. Using the bearing degree calculation formula, the microcontroller unit of the AV determined the bearing degree to be −23.18 degrees. To obtain the actual bearing degree, 360 degrees were added to the negative value, resulting in a final bearing degree of 336.81 degrees. Additionally, the travel distance was calculated to be 32.94 meters.

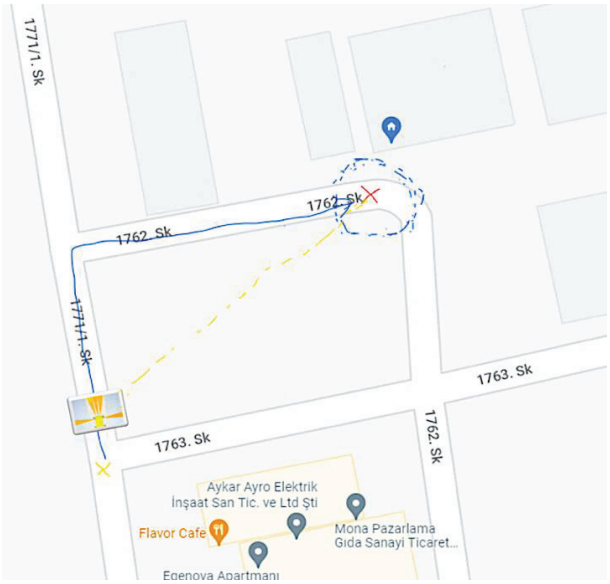


Figure 14(a). Experimental drive route

When the AV was positioned to the target locations, sensor readings were presented above in Figure 13. With these sensor readings, the percentage of error would be calculated. Firstly, the average value of the eight experimental readings will be calculated as:

$$\frac{\begin{array}{l}336.45 + 336.22 + 336.30 + 336.37 \\ +336.60 + 336.30 + 336.29 \\ +336.37\end{array}}{8} = 336.36 \quad (4)$$

$$\% \text{ Error} = \frac{|\text{Experimental Value} - \text{Theoretical Value}|}{\text{Theoretical Value}} \times 100 \quad (5)$$

Then following Eq. (5) was used to calculate percentage of error for magnetometer sensor:

$$\% \text{ Error} = \frac{|336.36 - 336.81|}{336.81} \times 100 = \%0.13 \text{ Error} \quad (6)$$

The navigation process exhibited a remarkably low level of error, which is highly favorable. Additionally, the continuous measurement of sensors while the autonomous vehicle is progressing toward the target destination significantly minimizes any potential errors, especially as the distance traveled becomes shorter.

Another routing test was conducted at a different location, as depicted in Figure 14(a). The initial position was determined as 38.447223, 27.226783, and the target position was set as 38.447524, 27.227266. The calculated bearing degree for this particular route was 51.49 degrees.

During the route between the two positions, five different obstacles with varying geometric shapes were randomly placed, as depicted in Figure 14(b). The autonomous vehicle (AV) operated at a rated speed of 0.5 m/s. A total of ten measurements were conducted, recording the time taken and the path traveled by the AV during each measurement.

Figure 14(b). Experimental drive route with five different geometrical shaped obstacles

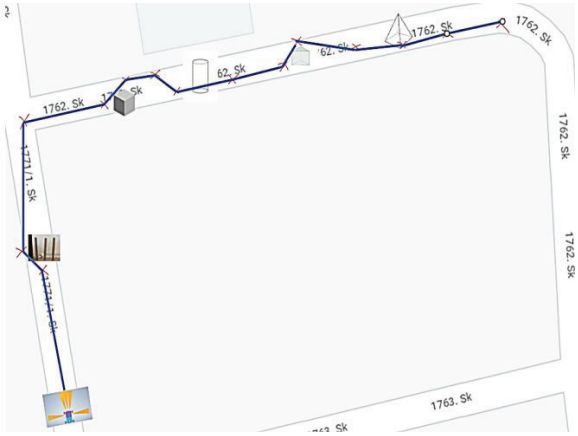


Figure 15(a). Traveled points and distance

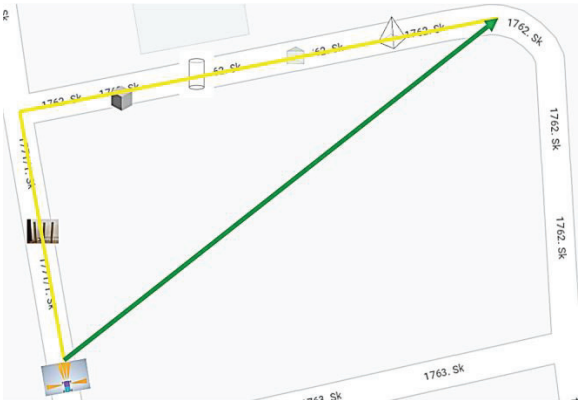


Figure 15(b). Ideal route and Euclidean Distance between two positions

The time measurements were recorded using a stopwatch on a mobile phone. This process was repeated ten times for each case. The paths traveled by the vehicle were manually marked on a sketchbook. The points were measured using Google Maps, as depicted in Figure 15(a).

The ideal path, which is the shortest line between the two positions, was calculated to be 98.25 meters. This ideal path is represented by the yellow line in Figure 15(b). On the other hand, the Euclidean distance between the two positions is 53.52 meters.

4. Results

During the experimental drive, a simple route was chosen to conduct tests. The initial position was at coordinates 38.447223, 27.226783, and the target position was at coordinates 38.447524, 27.227266. The vehicle calculated the bearing degree as 51.49 degrees.

As the vehicle started to move, it determined the shortest path, represented by the yellow hatched line. However, due to the presence of sidewalks, the vehicle followed the path indicated by the blue continuous line. The vehicle detected the sidewalks as obstacles, so instead of driving over them, it chose to stay on the road. The experimental drive was repeated ten times, as depicted in Figure 16.

Upon completing the experimental measurements, the 9th trial drew attention due to its travel distance of 120 meters, which was 21.75 meters longer than the ideal path length of 98.25 meters. This disparity may be attributed to errors in obstacle avoidance and GPS final position estimation. The remaining values appeared to be within an acceptable range. In order to calculate the error, it is necessary to determine the average distance traveled by the autonomous vehicle (AV).

$$\frac{\begin{array}{l}113.4 + 108.1 + 106.7 + 105.5 + 110.8 \\ +108.6 + 115.1 + 110.3 + 120.4 \\ +105.1\end{array}}{10} = 110.4 \tag{7}$$

By using Eq. (6):

$$\%\ \text{Error} = \frac{|110.4 - 98.25|}{98.25} \times 100 = \%12.3\ \text{Error} \tag{8}$$

The error value of 12.3% needs to be evaluated due to its effect on travel time. It is evident that this 12.3% error will lead to a 12.3% increase in travel time.

The vehicle's arrival was completed within a circle with a diameter of 5 meters, as depicted by the blue hatched lines in Figure 14(a). Based on observations, an accuracy of 2.5 meters in radius is considered acceptable.

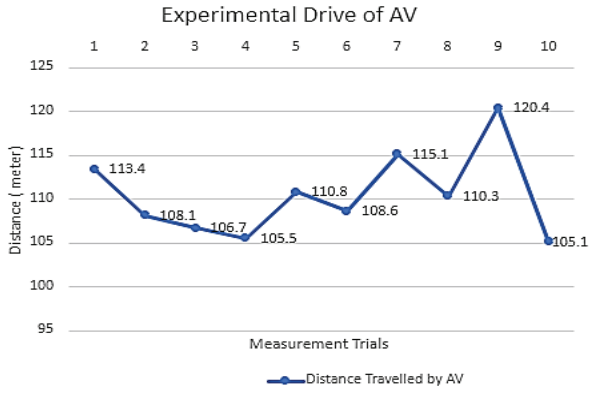


Figure 16. Experimental drive of AV [13]

The Firefly Algorithm, known for its effectiveness in complex and crowded environments, has demonstrated its capability to handle both linear and nonlinear problems. It exhibits a high convergence speed while not requiring a high-performance MCU (Microcontroller Unit) and has shown fast responses during the AV's mission.

Tests conducted without utilizing the Firefly Algorithm resulted in confusion during several trials. It appeared that the AV encountered difficulties in solving obstacle avoidance problems, leading to a state of confusion.

Ultimately, the advantage of this algorithm lies in its simple structure, which does not necessitate a complex and expensive controlling unit. The cost advantages associated with it further enhance its applicability.

5. Conclusion

The primary objective of this study was to design and manufacture an autonomous vehicle (AV). During the design phase, careful consideration was given to potential components such as cameras, ultrasonic sensors, and LiDARs, taking into account factors like cost, compatibility, and algorithms. After thorough research and evaluation, it was determined that ultrasonic sensors were capable of detecting obstacles effectively in outdoor conditions while also offering a cost advantage and a straightforward working principle.

The algorithm was intentionally designed to be simple in order to avoid encountering complex bugs during simulated delivery operations. The GPS position errors remained within a manageable range, with a diameter not exceeding 5 meters, which was deemed sufficient for the intended operations. However, the magnetometer was occasionally affected by environmental conditions that could not be precisely identified. This interference may have been caused by factors such as cell phone signals or other electronic components within the vehicle. Despite these occasional challenges, the AV successfully reached the target after deviating momentarily, which can be considered negligible since it was the initial trial on the roads during the adaptation phase.

The actual implementation and feasibility of such low-cost small-scale autonomous vehicles will depend on various factors, including technological advancements, regulatory frameworks, and market demand.

In conclusion, the following suggestions and recommendations can be made:

- For applications other than mail and package delivery, advanced positioning sensors may be required to ensure optimal performance based on specific operational needs.
- Creating low-cost autonomous robots can collect data on air quality, temperature, humidity, and other parameters, helping researchers and environmental agencies gather valuable information.
- The next stage of the study would involve incorporating mobile app-controlled positions and real-time vehicle tracking on online maps. This would necessitate wireless communication and present a new and challenging task in configuring the entire system.

AUTHORS

Ismail Bogrekci – Dept. of Mechanical Engineering, Faculty of Engineering, Aydin Adnan Menderes University, Efeler, Aydin, Turkey, e-mail: ibogrekci@adu.edu.tr.

Pinar Demircioglu* – Institute of Materials Science, TUM School of Engineering and Design, Technical University of Munich (TUM), Garching, Munich, 85748, Germany/ Dept. of Mechanical Eng, Faculty of Eng, Aydin Adnan Menderes University, Aydin, Turkey, e-mail: pinar.demircioglu@tum.de; pinar.demircioglu@adu.edu.tr.

Mustafa Yasir Goren – Schneider Electric Inc., Mechanical Design Engineer, Izmir, Turkey, e-mail: m.yasirgoren@hotmail.com.

*Corresponding author

References

[1] Q. Zhang, T. Zhang, and L. Ma, "Human Acceptance Of Autonomous Vehicles: Research Status And Prospects," *International Journal of Industrial Ergonomics*, vol. 95, 2023, 1–15. doi: 10.1016/j.ergon.2023.103458.

[2] G. Nativel-Fontaine, V. Lespinet-Najib, R. Cazes, C. Dupetit, C. De Gasquet, M. Chevrie, F. Aïoun, and L. Ojeda, "Exploration Of The Acceptability Of Different Behaviors Of An Autonomous Vehicle In So-Called Conflict Situations," *Accident Analysis & Prevention*, vol. 186, 2023, pp. 1–11. doi: 10.1016/j.aap.2023.107041.

[3] A.R. Pimenta, M. Kamruzzaman, and G. Currie, "Long-Term Effects Of Autonomous Vehicles On The Built Environment: A Systematic Scoping Review Towards Conceptual Frameworks," *Transport Reviews*, 2023, pp. 1–35. doi: 10.1080/01441647.2023.2189325.

[4] A.O. Prasad, P. Mishra, U. Jain, A. Pandey, A. Sinha, A.S. Yadav, R. Kumar, A. Sharma, G. Kumar, K.H. Salem, A. Sharma, and A.K. Dixit, "Design And Development Of Software Stack Of An Autonomous Vehicle Using Robot Operating System," *Robotics and Autonomous Systems*, vol. 161, 2023, pp. 1–12. doi: 10.1016/j.robot.2022.104340.

[5] Y. Joo, S. Kim, B. Kim, G.H. Cho, and J. Kim, "Autonomous Vehicles And Street Design: Exploring The Role Of Medians In Enhancing Pedestrian Street Crossing Safety Using A Virtual Reality Experiment," *Accident Analysis & Prevention*, vol. 188, 2023, pp. 1–12. doi: 10.1016/j.aap.2023.107092.

[6] Y. Wang and Z. Lin, "Research On Path Planning For Autonomous Vehicle Based On Frenet System", *Journal of Engineering Research*, 2023, pp. 1–6. doi: 10.1016/j.jer.2023.100080.

[7] H.A. Ignatious, H. Sayed, and M. Khan, "An Overview Of Sensors In Autonomous Vehicles," *Procedia Computer Science*, vol. 198, 2022, pp. 736–741, doi: 10.1016/j.procs.2021.12.315.

[8] Z. Wang, X. Zhao, and Z. Xu, "Offline Mapping For Autonomous Vehicles With Low-Cost Sensors," *Computers & Electrical Engineering*, vol. 82, 2020, pp. 1–11. doi: 10.1016/j.compeleceng.2020.106552.

[9] Y. Bai, B. Zhang, N. Xu, J. Zhou, J. Shi, and Z. Diao, "Vision-Based Navigation And Guidance For Agricultural Autonomous Vehicles And Robots: A Review," *Computers and Electronics in Agriculture*, vol. 205, 2023, pp. 1–20. doi: 10.1016/j.compag.2022.107584.

[10] L. Wang, Y. Zhang, and J. Wang, "Map-Based Localization Method for Autonomous Vehicles Using 3D-LIDAR", *IFAC-PapersOnLine*, vol. 50, no. 1, 2017, pp. 276–281. doi: 10.1016/j.ifacol.2017.08.046.

[11] S. Kitajima, H. Chouchane, J. Antona-Makoshi, N. Uchida, and J. Tajima, "A Nationwide Impact Assessment of Automated Driving Systems on Traffic Safety Using Multiagent Traffic Simulations", *IEEE Open Journal of Intelligent Transportation Systems*, vol. 3, 2022, pp. 302–312. doi: 10.1109/Ojits.2022.3165769.

[12] H. T. Zheng, C. Y. Chen, S. Li, F. Zheng, S. E. Li, Q. Xu, and J. Q. Wang, "Learning-Based Safe Control for Robot and Autonomous Vehicle Using Efficient Safety Certificate," *IEEE Open Journal of Intelligent Transportation Systems*, vol. 4, 2023, pp. 419–430. doi: 10.1109/Ojits.2023.3280573.

[13] M.Y. Goren, "Designing and Manufacturing Small Scale Autonomous Vehicle," Unpublished M.Sc. Thesis, 2023-M.Sc.-043, Aydin Adnan Menderes University, Turkey, 2023.