OPEN ACCESS
The Journal of Engineering and Exact Sciences – jCEC, Vol. 09 N. 11 (2023)
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075
ISSN: 2446-9416

A prototype of a low-cost autonomous mini-vehicle

Protótipo de baixo custo de um mini-veículo autônomo

Article Info:
Article history: Received 2023-11-19 / Accepted 2023-12-20 / Available online 2023-12-30
doi: 10.18540/jcecvl9iss11pp18149

Gabriel Arthur Vicentini
ORCID: https://orcid.org/0009-0000-1867-8774
Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais - Campus Ibirité, Brasil
E-mail: gabriel.arth30@gmail.com
Luan Gabriel Vidal da Silva
ORCID: https://orcid.org/0009-0002-7361-4233
Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais - Campus Ibirité, Brasil
E-mail: luangabrielvidal0000000@gmail.com
Carlos Dias da Silva Junior
ORCID: https://orcid.org/0009-0008-9564-2706
Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais - Campus Ibirité, Brasil
E-mail: carlos.dias@ifmg.edu.br
Elias José de Rezende Freitas
ORCID: https://orcid.org/0000-0002-6986-0602
Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais - Campus Ibirité, Brasil
E-mail: elias.freitas@ifmg.edu.br

Resumo
Este trabalho apresenta a primeira fase do desenvolvimento de um protótipo de um miniveículo autónomo em escala 1:24. A principal contribuição é a viabilidade e a execução do projeto utilizando apenas componentes simples, facilmente disponíveis e de baixo custo. O foco principal foi a construção de um chassi 3D funcional, capaz de alojar todas as peças necessárias, desenvolvendo componentes mecânicos 3D à escala real. Além disso, é apresentado um servidor-cliente inicial, usando a plataforma ROS (Robot Operating System), que permite a comunicação entre um computador central e o microcontrolador ESP32. De maneira que todos os dados que suportam este artigo estão disponíveis gratuitamente.
Palavras-chave: veículo autônomo. robôs móveis. design 3D.

Abstract
This work presents the first phase of developing a prototype of an autonomous mini-vehicle on a scale of 1:24. The main contribution is the feasibility and executability of the project using only simple, readily available, and low-cost components. The primary focus was to construct a functional 3D chassis capable of housing all the necessary parts, creating full-scale 3D mechanical components. In addition, we show the first server-client programming using the ROS (Robot Operating System) platform, which allows communication between a central computer and the ESP32 microcontroller. All data that supports this paper are freely available.
Keywords: autonomous vehicle. mobile robots. 3D design.

This article is an extended English version by (VICENTINI *et al*., 2023), presented in the IV CoBICET (IV Congresso Brasileiro Interdisciplinar em Ciência e Tecnologia).

1. Introduction

A mobile robot can be defined as a programmable mechanism with autonomy, capable of moving around an environment. An example of a mobile robot is the autonomous vehicle, which has a high degree of autonomy, e.g., a driverless vehicle capable of carrying out tasks related to driving without external human interference (FAISAL *et al*., 2019).

This technology will be increasingly present in society; it is estimated that by 2050, 50% of cars sold will be autonomous (CAMPOS, 2017). However, even though current technology already allows for a satisfactory level of autonomy and safety in experimental tests, there are still many gaps to be filled in developing these cars, e.g., the development of collaborative vehicle control and movement planning strategies.

In this sense, Hyldmar *et al*. (2019) point to a need for small-scale, low-cost systems to carry out navigation strategy experiments for multiple robots, also given the high cost of developing a single full-scale autonomous vehicle. However, many of these systems, even on a reduced scale, are expensive for Brazilian institutions to acquire. For example, Karaman *et al*. (2017) present a pedagogical program for elementary/middle school students, with positive results in project-based learning, using a 1/10 scale minicar platform for $1,060.00. According to (WILLIAMS *et al*., 2016), tests were conducted on a 1:5 scale platform to study control techniques for navigating these vehicles. The study presented a predictive control technique with an estimated cost of $9,210.00. Our team also developed one 1/10 autonomous vehicle to participate in the F1Tenth competition (SOUZA *et al*., 2021), with an estimated cost of $3,500.00.

In addition, integrating the Robot Operating System (ROS) is an essential key element in the quest for low-cost autonomous mini-vehicles (QUIGLEY *et al*., 2009). ROS is an open-source framework that has gained prominence in Robotics due to its flexibility, adaptability, and extensive community support. It provides a robust infrastructure for developing, controlling, and coordinating autonomous robots.

Considering this context, this article covers the early stages of developing a low-cost autonomous mini vehicle, including the electronic aspects, mechanical design, and firmware implemented in a microcontroller.

2. Methodology

The methodology used in this paper can be divided into steps: (i) Design and mechanical simulation: checking measurements, calculations, fitting the components, developing the functional and structural components of the prototype, and (ii) Computing and Control: programming the microcontrollers to control the system and communicate between the electronic and mechanical components.

The design and mechanical simulation step aimed to develop and test a solution for the structure and movement of the autonomous vehicle. The project was carried out using a scale of 1:24, where the mathematical relationship between the actual reference vehicle and its reproduction was reduced by 24 times.

The platform extension developed at this scale has approximate dimensions of 200x70x60mm. The scale was used to obtain a low-cost, reduced model compatible with the available components; however, this choice posed challenges in the implementation and arrangement of the components, which were solved via software.

Thus, the entire structure and functional components were created using CAD (ComputerAided Design) software, which, according to Groover and Zimmers (1983), can be defined as using computer systems to assist in creating, modifying, analyzing, and optimizing a design. Specifically, the approach used was parametric modeling, which allows for the development of easily adaptable designs. This is possible thanks to the constraints on associations that this approach uses to generate the model's geometries, which allows for greater precision in the dimensions and fittings developed.

The software chosen was Onshape, a cloud-based CAD tool that allows 2D and 3D modeling of complex systems directly from the browser, not requiring local installation. This advantage allows for greater progress at this step since the project can be accessed from any device and can also be modified in teams. The Onshape interface is shown in Figure 1.

In addition, Onshape has a free version for students and educators, through which the necessary components and functionalities for the mechanical design. It also provides a complete solution for assembling and evaluating various characteristics of the model, such as weight, dimensions, operation, and fittings.

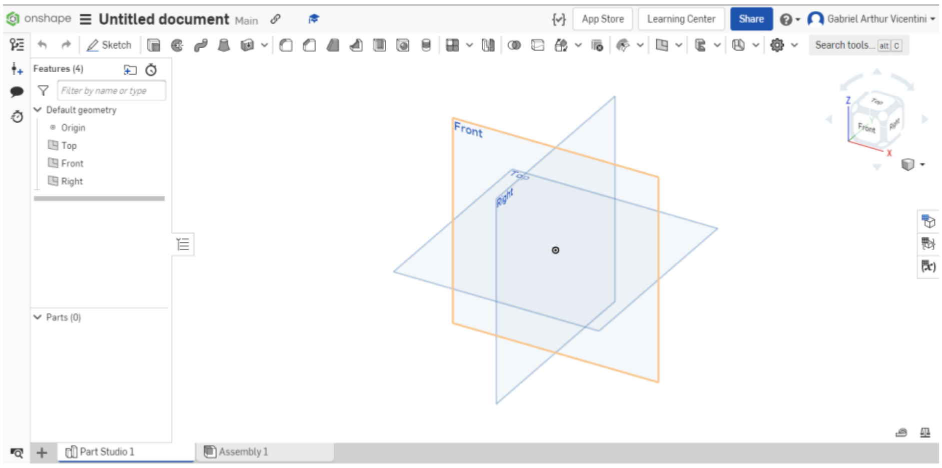

Figure 1 – Onshape software web interface.

For the development of the autonomous mini-vehicle platform, we selected the components listed in Table 1. This selection was based on the balance between quality and cost. The mini-vehicle has an onboard camera (ESP32-CAM) and an MPU6050 inertial sensor to sense the environment. The actuators are a servo motor and a DC motor powered by two 3.7V batteries. The ESP32 microcontroller platform is responsible for processing and communicating with the central node. This mini-vehicle and its tires are designed to be printed in PLA and TPU filaments, respectively.

Table 1 - List of components and approximate costs.

Components and approximate costs in 2023	
Component	**Costs (R$)**
Servo SG92R	25,00
MPU6050 Sensor	20,00
Voltage regulator	5,00
DC motor 6V N20 300rpm	40,00
Mini H-bridge L298N	10,00
16340 3,7V battery	20,00 (2 units)
ESP32	50,00
ESP32-CAM	80,00
Screws M2.5 6cm	20,00 (100 units)
Bearing 623zz	3,00
TPU Filament	80,00 (500g)
PLA Filament	80,00 (1 kg)

The firmware/software architecture was developed using ROS (Robot Operating System), enabling development in C++ and Python. We used a central computer with the Ubuntu 20.04

operating system, which is responsible for general processing, sending, and receiving data from the user and the platform.

This central node and the embedded microcontroller communicate via a Wi-Fi connection. The ROS communication is achieved through a publish-subscribe messaging paradigm, where nodes exchange data by publishing messages on specific topics and subscribing to the topics of interest. For example, the message defined in the geometry_msgs/Twist.h package makes it possible to send movement commands to the mini-vehicle (linear and angular velocity).

From the server-side perspective, we used the rospy library for accessing topics, services, and parameters, facilitating data exchange between the PC and microcontroller. On the microcontroller's client-side, the code was developed in C++ through Visual Studio Code (https://code.visualstudio.com/), employing the PlatformIO extension for direct ESP32 programming and handling WiFi communication, including the configuration of the network credentials.

3. Results

This section shows some results and details about the mechanical design. Figure 2 shows the exterior view of the complete 3D model proposed in this work. All the parts and pieces of the platform have been made publicly available in the cloud of Onshape (https://tinyurl.com/bdf2syx9). In this way, the virtual assembly allows for visualization of the prototype's functionalities. In addition, these mechanical files and the initial software developed are disponible in (Vicentini *et al.*, 2023b).

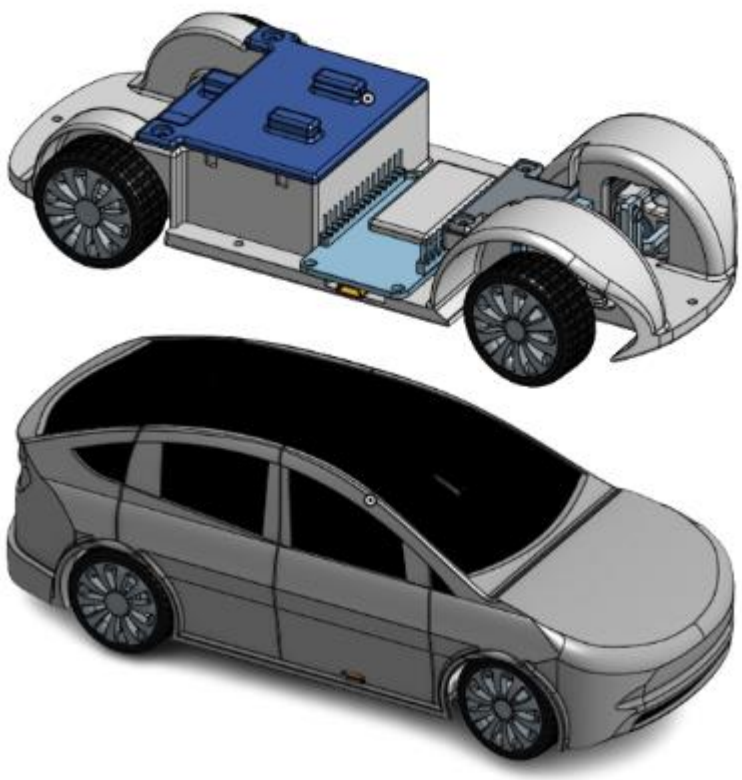

Figure 2 - Interior and exterior view of the complete 3D model developed.

A custom design was used to position the ESP32 components, L298 H-bridge and MPU6050 sensor, allowing them to be mounted without screws or other elements. In this way, spaces were allocated to contain the movement of the parts, which, dimensioned to have a small clearance, keep these components inert to the vehicle's movement, preventing damage and poor contact. This should make the vehicle easier to assemble and maintain, and reduce the cost of fasteners, as shown in Figures 3 and 4, using the components shown in Figure 5.

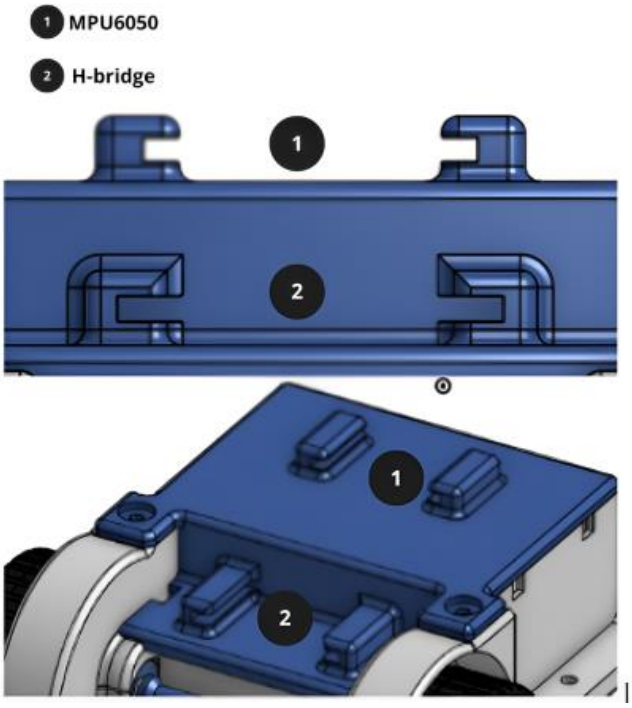

Figure 3 - Fitting for H-bridge and MPU6050 sensor.

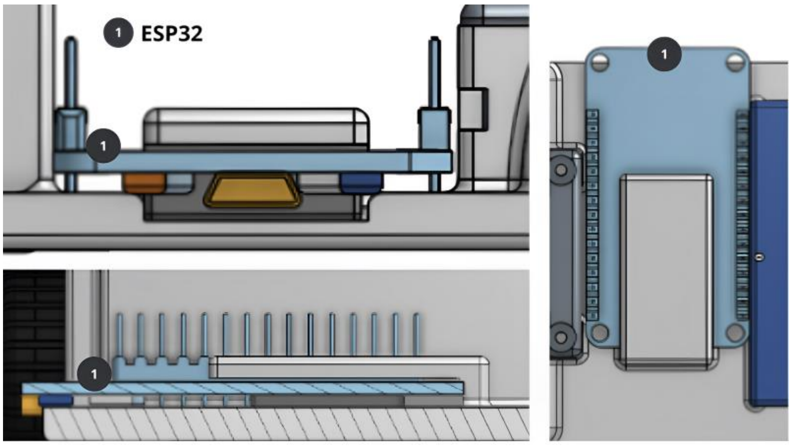

Figure 4 – Fitting designed for the ESP32.

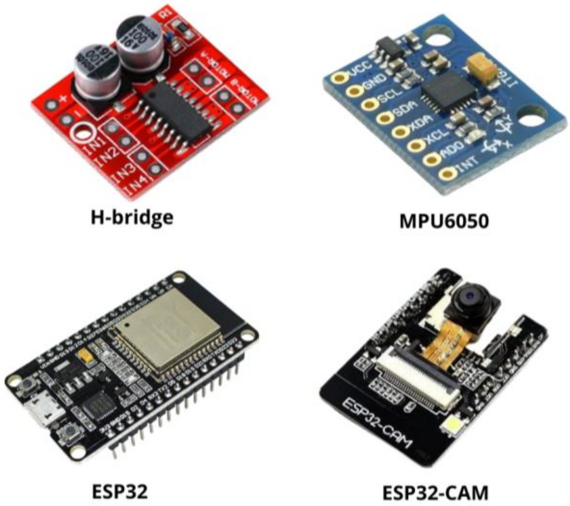

Figure 5 – The components used in the project: H-bridge, MPU6050, ESP32 and ESP32CAM.

The positioning of the other components followed the same pattern, but in a way that was more specific to their functionality and included screws and other parts. For the ESP32-CAM, it was necessary to create a specific relief and opening for the camera to capture a first-person view of the vehicle, which resembles the driver's view. This detail is shown in Figure 6.

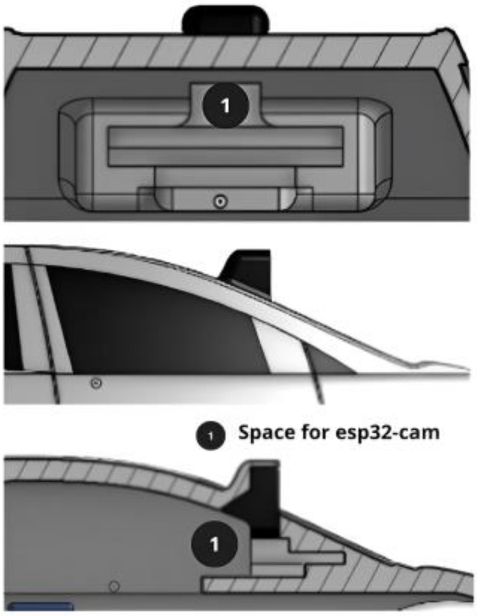

Figure 6 – Detail of the fitting for the onboard camera (ESP32-CAM).

To use the batteries, a commercial case was adapted, as shown in Figure 7, to facilitate removal for recharging and ensure the whole operation. The contacts and switch were used, connecting the output directly to the voltage regulator, and the set was placed in a different structure to the original one, already integrated into the vehicle structure to be compatible with the selected battery model, as shown in Figure 8.

Figure 7 - Commercial battery case used as a reference for the model.

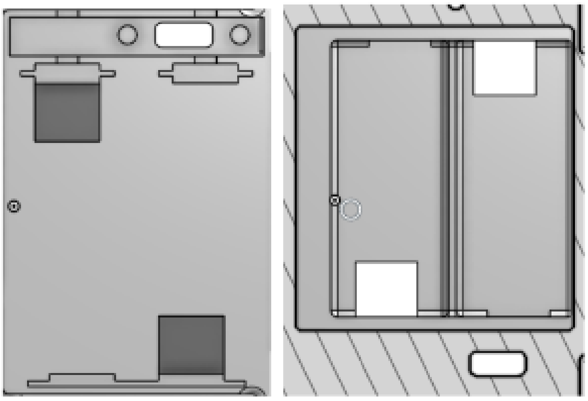

Figure 8 – Details of the space for fitting the batteries.

The traction system is based on a motor, axle and wheel assembly that transmits power by direct contact through a pair of 1:1 ratio gear. This design choice maintains a consistent, non-slip gear ratio and ensures that both components have identical dimensions. As a result, the force and speed generated by the engine are seamlessly transferred in parallel to the rear axle, with minimal variation, as detailed in the study by Flores and Gomes (2014). To optimize this system, the 623ZZ bearing is incorporated into the axle, which provides compactness and effectively reduces friction with the support on the base, as shown in Figure 9. In this context, the gear transmission is a robust and reliable solution for this specific application.

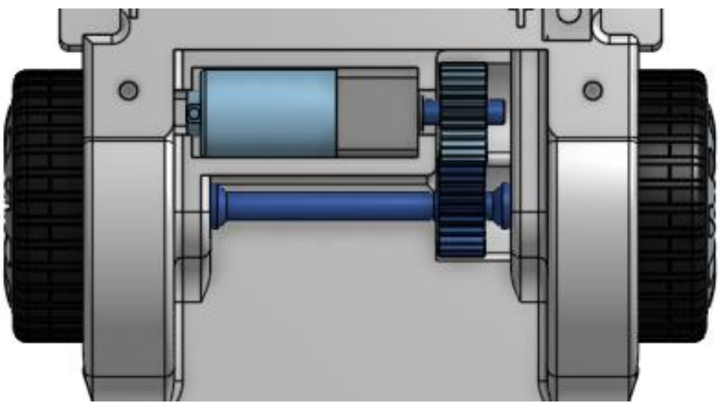

Figure 9 – Details of the traction system developed.

According to Wang and colleagues in 2020, Thermoplastic Polyurethane (TPU) demonstrates exceptional performance as a material for non-pneumatic tires, as shown in Figure 10. The study shows it offers better wear resistance than other rubber materials evaluated. Additionally, TPU can be easily utilized in 3D printing using FDM technology, making it a versatile material for various applications. As a result, we selected this material to print the vehicle's tire, making it easier to replicate the project.

Figure 10 – Wheel and tire developed for the mini-vehicle.

The servo motor controls the steering, where the radial movement is transformed into linear movement by means of a bar that allows the servo arm to move freely vertically but accompanies the horizontal movement. This movement is transferred to the component that supports the front wheels, through a channel that follows a radius from the axis that promotes the inclination of the wheels, again transforming the linear movement into radial. An example of this movement is presented in Figure 11.

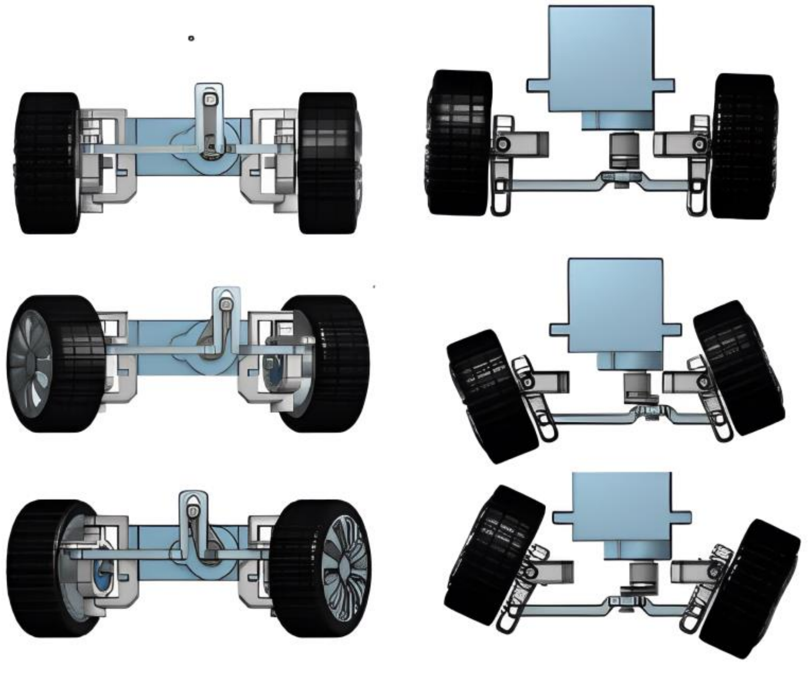

Figure 11 – Details of the steering system developed for the platform.

The model used is similar to the parallel steering model (Figure 12), which is easy to apply and control because it is based on mathematical relationships and a simplified structure compared to other geometries such as the Ackerman geometry, and yet promotes movement similar to real vehicles, unlike differential steering systems.

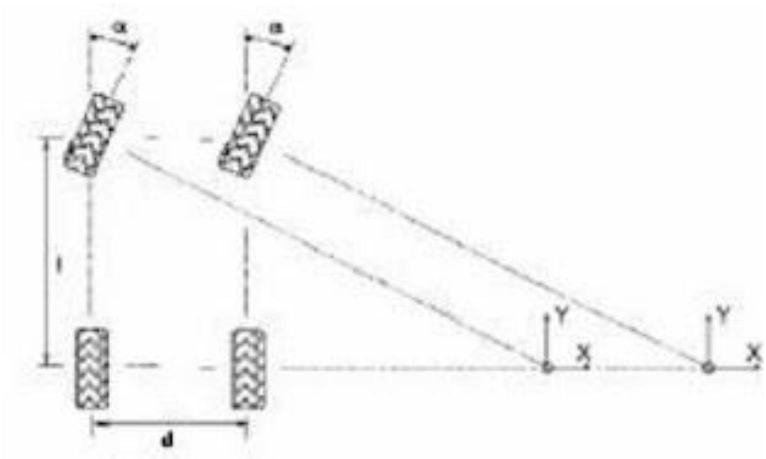

Figure 12 – Parallel geometry of the vehicle.

We used a protective "bubble" enclosure to protect the vehicle's internal components. This enclosure serves a dual purpose by not only shielding the internal components, but also acting as an integrated support structure for the ESP32-CAM, as illustrated in Figure 13. The design of this enclosure was inspired by the aesthetics of the Tesla Model X vehicle.

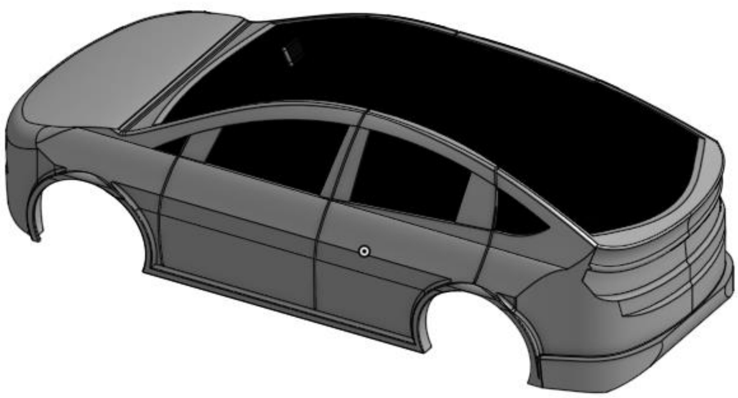

Figure 13 – Platform protection “bubble”.

In addition to 3D design, the assembly and movement simulation was seamlessly executed within the Onshape platform, enabling a comprehensive assessment of the mini-vehicle's functionality, compatibility, and dimensions.

To optimize production efficiency, the models were tailored for 3D printing, an increasingly popular rapid prototyping method. This process involves translating the 3D models into machine operating instructions, with a filament of PLA (polylactic acid) as the recommended material, chosen for its environmentally friendly properties, favorable mechanical characteristics and cost-effectiveness. During production, we utilized 8 grams of TPU (Thermoplastic Polyurethane) material and 345 grams of PLA, with a total print time of 3 days and 18 hours, employing a gyroid configuration and a 50% fill. The estimated unit cost is approximately R$260,00 ($50,00) in 2023.

Furthermore, we successfully tested the communication between the ESP32 and the central node. We verified that the ROS approach resulted in highly efficient communication, facilitating seamless data exchange between the server-side and client-side components of the system.

4. Conclusion

This paper has outlined the developmental and prototyping phases of a 1:24 scale autonomous mini-vehicle. As a result, a parallel steering system was developed that is more similar to the control of commercial vehicles, allowing it to be replicated, with control and communication via ROS. The prototype has been simulated and validated as an alternative for testing autonomous systems and control strategies and has high reliability in mechanical, electronic, and computational systems. Finally, as future work, we intend to develop a control system for such a platform, develop a differential to improve cornering, and create a suspension system. Furthermore, we desire to create a fleet of these mini-vehicles to work on autonomous vehicle coordination projects.

Acknowledgments

This work has been supported by the Brazilian National Council for Scientific and Technological Development (CNPq) and by the Federal Institute of Minas Gerais (IFMG).

References

Campos, E. (n.d.). Carros autônomos serão um mercado de US$ 7 Trilhões. Época Negócios. https://epocanegocios.globo.com/Empresa/noticia/2017/10/carros-autonomos-serao-um-mercado-de-us-7-trilhoes.html

Faisal, A., Kamruzzaman, M., Yigitcanlar, T., & Currie, G. (2019). Understanding autonomous vehicles. Journal of transport and land use, 12(1), 45-72.

Fingas, J. (2018). Waymo launches its first commercial self-driving car service. Sunnyvale, California: Verizon Media, 12.

Fingas, J. (2019, August). Waymo launches its first commercial self-driving car service. Engadget. https://www.engadget.com/2018-12-05-waymo-one-launches.html

Flores, P., & Gomes, J. (2014). Cinemática e dinâmica de engrenagens: 1-aspetos gerais sobre engrenagens.

Groover, M., & Zimmers, E. W. J. R. (1983). CAD/CAM: computer-aided design and manufacturing. Pearson Education.

Junior, F. E. F. (2020, March). Uma Introdução AO robot operating system (ROS). Embarcados. https://embarcados.com.br/uma-introducao-ao-robot-operating-system-ros/

Karaman, S., Anders, A., Boulet, M., Connor, J., Gregson, K., Guerra, W., ... & Vivilecchia, J. (2017, March). Project-based, collaborative, algorithmic robotics for high school students: Programming self-driving race cars at MIT. In 2017 IEEE integrated STEM education conference (ISEC) (pp. 195-203). IEEE.

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., ... & Ng, A. Y. (2009, May). ROS: an open-source Robot Operating System. In ICRA workshop on open source software (Vol. 3, No. 3.2, p. 5).

Souza, L. H. V.; Ribeiro, C. A. ; Freitas, E. J. R. . Robô móvel autônomo para corrida F1TENTH. In: Mostra Nacional de Robótica - 2021, 2021, (virtual). Mostra Nacional de Robótica - 2021, 2021. p. 1-6.

Vicentini, Gabriel A., da Silva, Luan G. V., Junior, Carlos D. da S. & Freitas, Elias J. R. (2023, September). PROTÓTIPO DE UM MINIVEÍCULO AUTÔNOMO DE BAIXO CUSTO. IV Congresso Brasileiro Interdisciplinar em Ciência e Tecnologia (IV CoBICET), Diamantina, MG, Brasil. doo: https://doi.org/10.5281/zenodo.8395120

Vicentini, G. A., Vidal da Silva, L. G., Dias da Silva Junior, C., & Freitas, E. J. de R. (2023). A prototype of a low-cost autonomous mini-vehicle (3D files). Zenodo. https://doi.org/10.5281/zenodo.10034155

Wang, J., Yang, B., Lin, X., Gao, L., Liu, T., Lu, Y., & Wang, R. (2020). Research of TPU materials for 3D printing aiming at non-pneumatic tires by FDM method. Polymers, 12(11), 2492.

Williams, G., Drews, P., Goldfain, B., Rehg, J. M., & Theodorou, E. A. (2016, May). Aggressive driving with model predictive path integral control. In 2016 IEEE International Conference on Robotics and Automation (ICRA) (pp. 1433-1440). IEEE.