Low-Cost Robot Construction Focused on Educational Environments

Douglas Favaretto[1][a], Vitor de Assis[1][b], Dieisson Martinelli[2][c], Andre Schneider De Oliveira[2][d] and Vivian Cremer Kalempa[1][e]

[1]*Department of Information Systems, Universidade do Estado de Santa Catarina, São Bento do Sul, Brazil*

[2]*Graduate Program in Electrical and Computer Engineering, Universidade Tecnológica Federal do Paraná, Curitiba, Brazil*

{*douglas.correia, vitor.monteiro9000*}*@edu.udesc.br, dmartinelli@alunos.utfpr.edu.br,*

Keywords: Low-Cost Robotics, ROS, Multi-Robot Systems.

Abstract: Designed for educational environments and motivated by the demand for affordable solutions in robotics. Cost serves as a limiting factor for the implementation of robotics projects, especially in educational environments with limited financial resources. By creating a robot composed of simple electronic components, this project aims to make robotic more accessible, enabling educational institutions to incorporate robotics into their educational programs, regardless of their constrained budgets. The robot's proposed features include the ability to navigate obstacles and teleoperation, utilizing the ESP8266 for Wi-Fi connectivity, ensuring its operational versatility. Furthermore, by introducing the robot into an educational environment and interviewing students, the platform demonstrated the robot as an effective, functional educational tool. Direct evaluations from students, contribute to changes and improvements in the platform. It fulfills its purpose of facilitating the learning of basic concepts in robotics.

1 INTRODUCTION

Mobile robotics is a field that has grown rapidly in recent years, and many activities and research focus on the autonomy and capability of robots to perform different tasks (Arvin et al., 2019). Nevertheless, as this technology advances, there arises a need to make it on hand to a broader target audience (Chronis and Varlamis, 2022). Low-cost robots thus become an important part of hands-on learning of robotic (Magrin et al., 2022).

In order to provide researchers with the capabilities and ideas of mobile robotics, this article proposes the development of a low-cost robot that can be used as an educational tool This robot is designed to work with other robots, in order to explore the basic concepts of multi-robot tasks (Feng et al., 2020). Using this robot, students will gain hands-on experience in operating and troubleshooting robotic systems.

The proposed robot is equipped with sensors and actuators, which can be programmed to meet their specific needs. Robots are designed to perform a variety of tasks, including navigating obstacles (Liu et al., 2019) and working in collaboration with other robots to achieve a common goal (Sherwani et al., 2020).The low-cost nature of the robot make it accessible to a larger number of educational institutions, allowing more educational institutions, allowing more students to take advantage of the educational opportunities offered by robotics.

This paper is organized as follows. In Section II, the construction of the robot and the use of Robot Operating System (ROS) in communication between components and the controller are explained. Section III discusses the use of a simulator for development as well as price comparison with other robotic platforms. Section IV covers the implementation of the low-cost robot in classrooms and the use of a questionnaire to assess students' interest, as well as the results of the robot implementation Section V covers plans for expanding the robot linup. Finally, Section VI presents the final considerations.

[a] https://orcid.org/0009-0003-3961-1247
[b] https://orcid.org/0009-0007-7022-6135
[c] https://orcid.org/0000-0001-7589-1942
[d] https://orcid.org/0000-0002-8295-366X
[e] https://orcid.org/0000-0001-9733-7352

Favaretto, D., de Assis, V., Martinelli, D., Schneider De Oliveira, A. and Kalempa, V.
Low-Cost Robot Construction Focused on Educational Environments.
DOI: 10.5220/0012987500003822

In *Proceedings of the 21st International Conference on Informatics in Control, Automation and Robotics (ICINCO 2024) - Volume 2*, pages 66-72
ISBN: 978-989-758-717-7; ISSN: 2184-2809

2 METHODOLOGY

This section describes in detail the design and construction of the robot as well as a control system using ROS (Quigley et al., 2009).

This section reviews the main features of the proposed robot and integrates ROS. Divided into three main parts: controller, mobile base and perception source, it explores how these components interact to achieve efficient communication, navigation and environmental awareness.

2.1 Controller

In the scope of robotics, efficient communication among different components is crucial for the coherent and smart working of the system. The ROS Noetic, running on Ubuntu 20.04, is the primary tool used in this project, being a key and widely recognized software framework for robotics (Blubaugh et al., 2022).

ROS provides a modular framework for robotics systems development by providing tools, libraries, and conventions to simplify programming for robots. The connection via ROS plays a crucial role. In the specific context of this project, ROS enables efficient communication between the controller and the mobile base through the TCP/IP protocol over a WiFi network.

When initializing the parameters of the mobile base and specifying the Wi-Fi network to connect to, along with the IP of the machine running the ROS, a robust bridge for information exchange is established. By creating a common environment for communication enables robot control, transmission of commands from the controller, and reception of data collected by the mobile base.

The script provided in Figure 1 is a shell script, a set of commands in a bash script, used to configure environment variables related to the connection between the controller and the mobile base. In this context, the bash script sets up two specific environment variables. ROS_MASTER_URI is responsible for specifying the URL of the ROS Master. ROS Master is a core component that manages coordination and communication among the various nodes, which are the processes that constitute the ROS system. The value allocate to ROS_MASTER_URI is "http://xxx:11311/", specifying that the ROS Master is find on a system with the IP address "xxx" and on port 11311.

Other variable is ROS_IP, used to specify the IP address that the ROS system should employ to communicate with other systems. This is particularly relevant in network scenarios where the system has multiple network interfaces. In the case of this script, ROS_IP is set to the same IP address as the ROS_MASTER_URI.

```
conecta_lena(){
export ROS_MASTER_URI=http://xxx.xxx.xxx.xxx:11311/
export ROS_IP=xxx.xxx.xxx.xxx
}
```

Figure 1: Shell script to connect multiple machines.

These configurations are crucial to ensure effective communication among the nodes. By explicitly setting the ROS master and the IP address to be used, the Figure 1 enables the system to operate properly on a specific network. It is sufficient for the operator, upon opening a new terminal, to call the "conecta_lena" function to establish the connection with the mobile base, facilitating interaction with it.

2.2 Mobile Base

The mobile base is equipped with the WeMos D1 prototyping board based on the ESP8266 chip, which features a 32-bit microcontroller. This microcontroller, specifically the Xtensa LX106, operates at a clock speed of 80 MHz, offering sufficient processing power for various IoT applications. Additionally, it includes built-in Wi-Fi capabilities, making it ideal for projects that require wireless communication (Macheso and Meela, 2021).

The microcontroller has 4 MB of Flash memory, 64 KB of instruction RAM, and 96 KB of data RAM, providing adequate memory for running complex tasks. Designed to be a low-cost, user-friendly Internet of Things (IoT) development board, the ESP8266 is known for its ability to connect to Wi-Fi networks and its low power consumption, making it a versatile component for a wide array of embedded systems (Budiharto et al., 2021).

The WeMos D1 board has a central role in the project, acting as the main controller for the mobile base. Its compatibility with the Arduino platform simplifies programming using the Arduino IDE, leveraging features and support from the existing community (Peña, 2020).

To control the robot's behavior, code was developed in the Arduino IDE. This code enables the definition of which parts of the system will be responsible for sending and receiving messages. Specifically, it configures communication with the infrared sensor, the mobile base motors, and the controller.

In the context of the motors, a callback function named "odometry_cb" was implemented. This function handles messages related to the robot's velocity, calculating and controlling the motor speeds based on the received instructions. This approach enables the robot to respond appropriately to motion commands by converting odometry information into signals understandable by the motors.

To better integrate these components, the WeMos D1 board is equipped with a shield, reducing the need to connect a large number of wires between them. Additionally, the robot's power is supplied by a 10000mAh power bank, connected to the board.

Additionally, the robot's was designed using SketchUp Online, a free tool, requiring minimal effort for learning (Chiodi and de Figueiredo, 2021). The fully assembled robot is shown in Figure 2.

Figure 2: Mobile Base.

Regarding the fabrication of the parts, a GT-Max3D 3D printer, model A2V2, was utilized, employing ABS material in green.

2.3 Source of Perception

The Sharp GP2Y0A41SK0F distance sensor was incorporated as the perception source, as can be seen in Figure 3. This sensor employs an infrared light measurement principle, emitting a beam of light towards the object (Siyal et al., 2021).

This sensor is constituted by the combination of a Position Sensitive Detector (PSD), a device that measures the position of a point of light or incident radiation on its surface. It typically consists of an array of photodiodes, the PSD provides information about the position of the incident light, allowing its application in tracking, positioning, and control of optical devices.

An Infrared Emitting Diode (IR-LED) is a type of semiconductor diode designed to emit infrared light. They are commonly utilized in remote controls devices. They operate by emitting infrared light when an electric current passes through them.

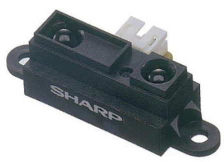

Figure 3: Source of Perception.

The sensor can provide information about the distance between itself and the target object. The IR-LED emits infrared light, which reaches the object and is reflected, captured by the PSD, as depicted in Figure 4. The resulting signal is then processed by a circuit that converts the distance into a voltage variation.

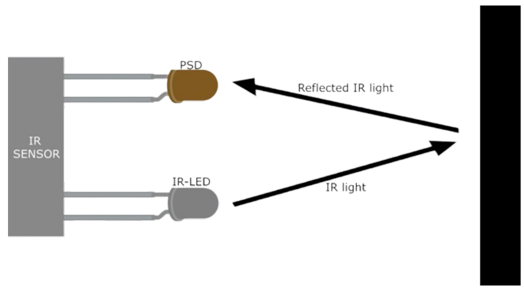

Figure 4: Infrared Sensor functioning.

To increase the sensor's perception range, it was mounted on top of the robot's structure, connected to a servo motor. The servo motor used is an MG90S, ideal for robotics projects or radio-controlled models. With the addition of the servo, it became possible to expand the robot's spatial perception from a single position to a total of 180 positions.

In the code that establishes the relationship between the Sharp Infrared (IR) sensor and the servo motor, was included the "SharpIR" library, which provides specific functionalities for working with Sharp sensors. The creation of an object called "SharpSensor" to represent the sensor. The "SharpSensor" object is configured for the "GP2Y0A41SK0F" model and is connected to analog pin A0.

Additionally, a topic named "topic_sharp" is established to facilitate communication between system components. A publisher is configured to send information from the sensor to this topic, enabling the efficient integration of these data across different parts of the project.

Similar to the infrared sensor, an object named 'servo' is created to represent the control of the servo motor. The variable 'topic_servo' identifies the topic related to servo control in ROS. This topic is associated with a callback function responsible for controlling the servo motor based on the received messages.

The servo callback function is triggered when a message is received on the servo topic. This message contains data representing the desired angle for the servo motor. A function is then employed to adjust the servo's position based on the information contained in the message.

This strategy provides a solution to emulate functionalities similar to those of a LIDAR sensor, enabling the robot's controller to make informed decisions based on distance readings (Li and Ibanez-Guzman, 2020). The sensor/servo combination makes a solution for acquiring data related to the robot's spatial perception, allowing more adaptive interaction with the surroundings.

3 COST ANALYSIS AND SIMULATION

The development prioritized affordability, by utilizing components available in the Brazilian market. The costs incurred can be observed in Table1.

Table 1: Components Cost.

Description	Prices
Micro Servo Motor	$ 4,70
Sensor IR Sharp Gp2y0a21	$ 7,43
Power Bank	$ 12,40
Jumper Cables	$ 2,22
H Bridge	$ 3,14
3D printer Filament	$ 3,06
WeMos D1 Board	$ 10,91
Shield	$ 2,68
4 DC Motors Kit	$ 11,09
Total	$ 57,64

The total cost, as shown in Table 2, demonstrates a savings of approximately 55% compared to Mona and 66.85% compared to FOSSBot.

Table 2: Price Comparison with Other Robotic Platforms.

Robot	**Price**
Developed Robot	$ 57,64
Mona	$ 128,00
FOSSBot	$ 200,00

The cost-benefit analysis was based on the comparison of prices with other robotic platforms such as FOSSBot (Chronis and Varlamis, 2022) and Mona (Arvin et al., 2019), both focus on simplified construction and reduced costs for the widespread promotion of robotics studies, these comparison reveals a significant advantage for the developed robot.

The cost analysis provides a quantitative assessment, but it is also important to consider the performance provided by the robot, emphasizing its efficiency and viability.

Simultaneously, the application of the CoppeliaSim environment, formerly known as V-REP (Rohmer et al., 2013), a widely used platform for robotic simulation, enabling modeling and simulation of robot behavior in virtual environments.

For example, (Montenegro et al., 2022) developed a spherical robot that has only a single point of contact with the ground. The robot moves using joints, axes, counterweights, and a pendulum. The CoppeliaSim simulator was used to conduct a series of simulations regarding the robot's behavior.

Similarly, (Chakraborty and Aithal, 2021) explored the potential of CoppeliaSim to develop a robotic arm with the purpose of facilitating robotics research. The authors emphasize two main problems: the high cost of real robots and the potential risks involved in testing real robots due to bugs or abnormal activities that could result in damage to property, risks to human life, or harm to the robot itself.

Additionally, a simulated version was developed in the CoppeliaSim environment, as shown in Figure 5. The configuration and operation of the robot in the simulator mirror those of the real robot, incorporating the same methods of information exchange between the mobile base and the controller.

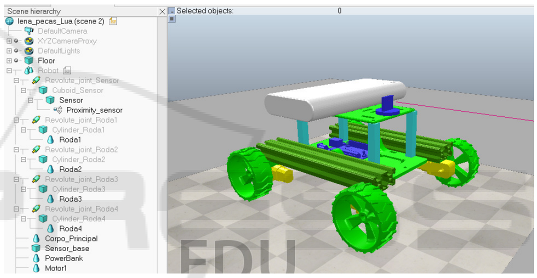

Figure 5: Simulated robot in CoppeliaSim.

This approach, as evidenced by previous studies, enables the simulation of techniques before implementation on the physical robot, providing a safe and efficient space for virtual testing. This approach not only facilitates the understanding of the robot's capabilities but also underscores the versatility of CoppeliaSim as an essential tool in the development and enhancement of various robotic applications.

4 CLASSROOM APPLICATION

The low-cost robot was incorporated into two courses in the Bachelor of Information Systems program at the Universidade do Estado de Santa Catarina in the Centro de Educação do Planalto Norte (UDESC CEPLAN): Special Topics I and Special Topics II. In these courses, students used their machines to connect to the robot, as demonstrated earlier.

The proposed activity consisted of two main stages. In the first stage, the students developed a teleoperation system that allowed remote control of the robot through manual commands. This phase was

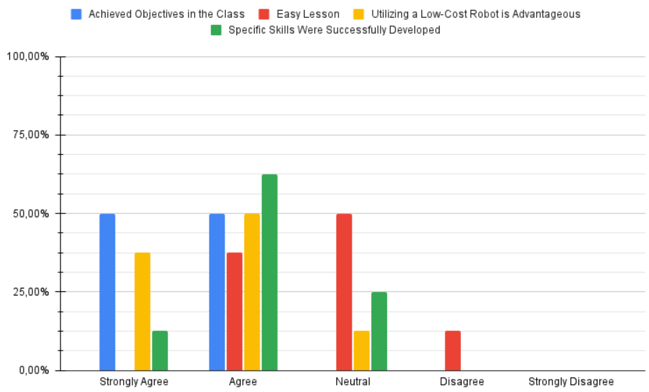

Figure 6: Questionnaire Results.

used to help students understand the basic principles of robot movement and control.

In the second stage, the focus shifted to the implementation of an obstacle avoidance system. The students were challenged to develop an algorithm capable of autonomously detecting and avoiding obstacles using the Sharp sensor to identify obstacles and adjust the robot's trajectory as needed. This phase aimed to introduce advanced robotics concepts, such as environmental perception and decision-making.

With the aim of evaluating the level of interest of the students in working with the robot, a questionnaire consisting of 9 questions was manege after the classes. Five questions were open-ended, seeking the students' opinions, while four were based on the Likert Scale.

The Likert scale is commonly used in quantitative visualization. According to (Bertram, 2007), Likert scales are a measurement method that does not involve comparisons and focuses on a single dimension. Participants has to express their degree of agreement with a statement. These scales are designed to assess the intensity of opinion or attitude regarding a single topic.

The implementation of the robot in the classroom, followed by the questionnaire administered, revealed a variety of perceptions, as shown in Figure 6. The graph displays a scale from 0 to 100, depicting the students' opinions regarding the formulated questions. It was possible to gather feedback from the students and understand their opinions and the benefits that the use of the robot brings to professional development.

The Likert scale questions made it possible to evaluate how well the lesson achieved the objective of using the robot, the difficulties encountered, the advantages of using the robot, and the development of specific skills.

In terms of achieving the objectives, the students expressed that the results were attained, as exemplified in Figure 6, with predominantly positive responses. The difficulty of the class was mostly considered moderate or easy, indicating a relative accessibility of the technology.

Regarding perceived benefits include the convenience of transportation and the chance to engage with an actual robot for experimentation without any associated risks. The relevance of low cost in technology dissemination was acknowledged, emphasizing the potential for inclusion across different social classes.

Some challenges were identified, such as adapting values between simulator and real robot, the robot's adherence to the floor, and the lack of documentation to maximize the robot's possibilities. These comparative study highlights the effectiveness of affordable robotics for use in education, identifying accessibility, benefits and challenges to be addressed.

5 FUTURE PROJECTS

For future projects, this initiative opens up the possibility of expanding the robot lineup by introducing specialized variants and more advanced models. Exploring the integration of 3D cameras and LIDAR and more powerful actuators, enhancing the robot's ability execute specific tasks.

Furthermore, artificial intelligence and machine learning technologies will offer new possibilities for the autonomy and interaction of robots. These advances will provide access to cutting-edge technologies.

6 CONCLUSION

This work introduced the conception and construction of a low-cost robot designed for educational use in classrooms and research, focusing on learning and applying essential concepts in mobile robotics on a simple platform. The robot, composed of accessible components, provides an economical alternative for institutions, enabling the exploration of various tasks and scenarios in the field of robotics.

The methodological approach adopted involved a detailed explanation of the three main parts and their components: the Controller, the Perception Source, and the Mobile Base. The use of ROS facilitated communication between these components, while the CoppeliaSim expanded possibilities before application to the real robot.

In practical application in university courses, the robot demonstrated its feasibility in aiding the teaching and understanding of robotics concepts. The cost evaluation revealed its financial advantage for institutions with limited budgets compared to other platforms. The creation of a simulated version of the robot complemented the hands-on experience, providing a safe environment before implementation on the physical robot.

Moreover, the implementation of the robot in university courses not only confirmed its utility as an educational tool but also highlighted its ability to adapt to different levels of complexity and areas of study. This includes basic introduction to robot programming to more complex tasks.

In conclusion, the development of the low-cost robot contributes significantly to robotics by providing an affordable and practical solution. Its application in the classroom, along with simulation in CoppeliaSim, demonstrates its utility and versatility. The combination of affordable materials and the potential to expand knowledge in robotics emphasizes the importance of this innovation for both educational and scientific applications

ACKNOWLEDGEMENTS

The project is supported by the National Council for Scientific and Technological Development (CNPq) (process CNPq 407984/2022-4); the Fund for Scientific and Technological Development (FNDCT); the Ministry of Science, Technology and Innovations (MCTI) of Brazil; the Araucaria Foundation; and the General Superintendence of Science, Technology and Higher Education (SETI).

REFERENCES

Arvin, F., Espinosa, J., Bird, B., West, A., Watson, S., and Lennox, B. (2019). Mona: an affordable open-source mobile robot for education and research. *Journal of Intelligent & Robotic Systems*, 94:761–775.

Bertram, D. (2007). Likert scales. retrieved november, 2, 2013.

Blubaugh, D. A., Harbour, S. D., Sears, B., and Findler, M. J. (2022). Noetic ros further examined and explained. In *Intelligent Autonomous Drones with Cognitive Deep Learning: Build AI-Enabled Land Drones with the Raspberry Pi 4*, pages 417–436. Springer.

Budiharto, W., Andreas, V., Irwansyah, E., Suroso, J. S., and Gunawan, A. A. S. (2021). Implementation of wemos d1 for wi-fi based controller tank-based military robot. *ICIC Express Lett. Part B Appl*, 12(4):377–382.

Chakraborty, S. and Aithal, P. (2021). A custom robotic arm in coppeliasim. *Chakraborty, Sudip, & Aithal, PS,(2021). A Custom Robotic ARM in CoppeliaSim. International Journal of Applied Engineering and Management Letters (IJAEML)*, 5(1):38–50.

Chiodi, J. R. and de Figueiredo, S. A. (2021). O ensino de geometria plana em aulas online sob a perspectiva de vygotsky por meio do software sketchup. *Anais do Seminário Sul-Mato-Grossense de Pesquisa em Educação Matemática*, 15(1):1–12.

Chronis, C. and Varlamis, I. (2022). Fossbot: An open source and open design educational robot. *Electronics*, 11(16):2606.

Feng, Z., Hu, G., Sun, Y., and Soon, J. (2020). An overview of collaborative robotic manipulation in multi-robot systems. *Annual Reviews in Control*, 49:113–127.

Li, Y. and Ibanez-Guzman, J. (2020). Lidar for autonomous driving: The principles, challenges, and trends for automotive lidar and perception systems. *IEEE Signal Processing Magazine*, 37(4):50–61.

Liu, Z., Lü, Z., Zheng, W., Zhang, W., and Cheng, X. (2019). Design of obstacle avoidance controller for agricultural tractor based on ros. *International Journal of Agricultural and Biological Engineering*, 12(6):58–65.

Macheso, P. S. B. and Meela, A. G. (2021). Iot based patient health monitoring using esp8266 and arduino. *International journal of computer communication and informatics*, 3(2):75–83.

Magrin, C. E., Ribas, I. C. L., Ribas, T. A., Vitola, M. S., Treaquin, P., Surdi, J. J., Magrin, C. S., and Todt, E. (2022). Promovendo a aprendizagem da robótica nas escolas com metodologias ativas e o desenvolvimento de um robô móvel acessível para redução das desigualdades sociais. *Anais do Computer on the Beach*, 13:212–219.

Montenegro, G., Chacón, R., Fabregas, E., Garcia, G., Schröder, K., Marroquín, A., Dormido-Canto, S., and Farias, G. (2022). Modeling and control of a spherical robot in the coppeliasim simulator. *Sensors*, 22(16):6020.

Peña, C. (2020). *Arduino IDE: Domina la programación y controla la placa*. RedUsers.

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., Ng, A. Y., et al. (2009). Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, Japan.

Rohmer, E., Singh, S. P., and Freese, M. (2013). V-rep: A versatile and scalable robot simulation framework. In *2013 IEEE/RSJ international conference on intelligent robots and systems*, pages 1321–1326. IEEE.

Sherwani, F., Asad, M. M., and Ibrahim, B. S. K. K. (2020). Collaborative robots and industrial revolution 4.0 (ir 4.0). In *2020 International Conference on Emerging Trends in Smart Technologies (ICETST)*, pages 1–5. IEEE.

Siyal, A. et al. (2021). Optical distance sensors. *Acta Mechanica Slovaca*, 25(3):28–32.