Open source and open hardware mobile robot for developing applications in education and research

D. Betancur-Vásquez, M. Mejia-Herrera, J.S. Botero-Valencia *

Grupo de Automática, Electrónica y Ciencias Computacionales, Instituto Tecnológico Metropolitano, Medellín, Colombia

**A R T I C L E   I N F O**

**Keywords:**
Mobile robotics
Omnidirectional movement
Sensorics
Navigation

**A B S T R A C T**

Nowadays, additive manufacturing, rapid prototyping and assembly modules represent a market that has invaded the entire world, especially in developing countries where traditional manufacturing is more restricted. In robotics, it is pertinent to think that modular construction is essential, due to the complexity of geometry in each of the pieces and their manufacture. Taking into account the globalization of information and the worldwide reproduction of databases, facilitating access to CAD files to be reproduced in 3D printing promotes the easy construction of archived mechanical designs. A robotic architecture becomes a complex assembly by having multiple operating systems. The sensorics, mechanics, electronics and programming that it requires for navigation, collaboration, development, operation and even industrial manufacturing means that more and more elaborate embedded systems are used. In this work, a mobile robotics architecture was developed with a sensory system that allows free movement and navigation in closed loop inverse kinematics. This kind of robot uses navigation algorithms to take a trajectory in collaborative closed environments, that is, closed industrial environments where obstacles are normally immovable and corridors to move narrow, in addition to having mobile obstacles like humans.

© 2021 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

**Specifications table:**

| Hardware name | Omnidirectional shock-absorbed Robot |
|---------------|-------------------------------------|
| Subject area  | Robotics, Omnidirectional moving, Sensors |
| Hardware type | Autonomous navigation, Automatic Control, Mechanical and Structural development |
| Open source license | Creative Commons Attribution-ShareAlike license |
| Cost of hardware | $ 6–00 USD |
| Source file repository | https://doi.org/10.17605/OSF.IO/KQ3EW |

* Corresponding author.

E-mail address: juanbotero@itm.edu.co (J.S. Botero-Valencia).

https://doi.org/10.1016/j.ohx.2021.e00217

2468-0672/© 2021 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).
1. Hardware in context

The development of mobile robotic systems has become an area of interest to the industry due to their potentials to transport, assemble, and perform tasks that increase the performance and profitability of a company and reduce healthy risks for human beings [1]. Many of the robotic systems that exist today use QR codes or color marks to control and coordinate their movements in space, which is a problem in a dynamic environment where such tags can be obstructed by the presence of obstacles [2–4].

Although it is possible to find a wide variety of unmanned robotic platforms [5–9], many of them are not suitable for operation in enclosed environments with human presence. And some of them are focused on aquatic environment operation as shown in Table 1, which is not often presented in industrial applications.

Other electronic platforms have been developed to help robotic systems, integrating sensors for imaging systems or environmental variables analysis, such as [10–14]. These platforms facilitate robotics systems navigation in different environments, as well as other research tasks like data acquisition. Given the risk posed by machines moving between humans in a closed environment, terrestrial collaborative mobile robotic platforms have been developed. These systems consider an individual’s location, movement, and personal space to calculate trajectories and evade potential obstacles [15,16], which made them one of the most important technologies for Industry 4.0 and its applications.

While there are several ways to build a ground vehicle, the alternative that provides greater freedom of movement on a plane is the omnidirectional wheels. This type of wheel facilitates movement in any direction including 90 degree turns, which is an advantage for both system design and path planning due to the decreasing of the number of moving parts, and the complexity of the routing system. The omnidirectional wheels also allow path planning using straight lines, which is more natural than calculating trajectories using turning radius or other [17] motion architectures. The developed platform can perform path planning with an accuracy of up to 0.06020 m by running on a flat surface. Nevertheless, the system can also perform path planning with some corrections when riding over a non-flat or irregular Surface. Such ability is highly desirable when doing exploration or inspection missions. In addition, the modular construction of the designed device allows fulfilling the needs of a variety of problems, such as gathering information in some hazardous or human unfriendly environments, reach difficult places, or assist different tasks like delivery, among other cooperative processes. As well as reducing the budget through the Fused Deposition Material technology and the use of Mecanum wheels, which are easily replaceable, and are broadly available at lower prices than similar traction systems. Navigation in indoor spaces has problems in the narrow corridors available to travel. The trajectories, therefore, must be better planned and the robot strategically designed to be able to overcome any present obstacle. Even counting with dynamic obstacles as humans, the detection of these and the sensory assembly that must be had is key to get from a starting point to a final one [18].

2. Hardware description

2.1. Mechanical Hardware

The assembled hardware is a mobile robotics platform, which can move omnidirectionally using Mecanum’s wheels. The robot was mainly built with 3D printing materials through FDM (Fused deposition modeling) made of Polyactic acid (PLA), making its manufacture, construction, and assembly not only cheap but easy to implement. Additionally, some of the parts were made of aluminum to improve resistance to bending, and compression in critical areas of the chassis. And some other pieces were laser cut in acrylic sheets to achieve greater accessibility of manufacture/materials and reliable resistance at a low cost. It is imperative to note that the damping mechanism is necessary when using Mecanum wheels. These wheels produce vibration to the structure by not having a continuous touch of the rollers with the contact surface [19]. These vibrations cause a mismatch in the assembly hardware and noise in the sensors used by the robot. In addition, having a damping system allows each wheel to overpass small obstacles on the road without losing orthogonal contact of any of the four wheels with the floor surface, preserving omnidirectional movement effect and therefore, the correct operation of the kinematics of the robot. For these reasons, the damping mechanism is designed in a way that preserves continuous orthogonality with the ground when flanking obstacles or non-flat surfaces [20].

Table 1
Brief capabilities review of some of the developed Robotic platforms.

| Platform | Three Degrees of Freedom | Obstacle Avoidance | Absorb Shock System | Ground Use |
|----------|--------------------------|--------------------|---------------------|------------|
| Own      | Y                        | Y                  | Y                   | Y          |
| [5]      | N                        | Y                  | N                   | Y          |
| [6]      | Y                        | N                  | N                   | Y          |
| [7]      | N                        | N                  | N                   | N          |
| [8]      | N                        | Y                  | N                   | N          |
| [9]      | N                        | N                  | N                   | Y          |
2.2. Electronics Hardware

The platform integrates distance sensors to monitor environmental activity. The Electronics system consists of three main electronic sections: electro-mechanical power unit, sensory intercommunicated system and controllers.

2.2.1. Electro-mechanical power unit

The electromechanical motor system has four DC Faulhaber 12V motors, assembled with Mecanum wheels. One H-bridge powerboard regulates the speed rotation and direction of each motor, and therefore the omnidirectional movement of the robot, by using a PWM pulse calculated using the inverse kinematics of the robot and sent by the Teensy 3.2.

2.2.2. Sensory Intercommunicated System

In robotic mobile platforms, the distance sensors not only aid to locate the robot in space but warn of static and dynamic obstacles in the coplanar space of the robot and its surroundings. The selection of sensors is based on the cost/benefit balance. The electronic sensors used provide the highest efficiency in terms of performance while preserving the main idea of the development of the robot to keep low cost. The sensory modules implemented data fusion precisely to improve the efficiency of low-cost sensing, giving economical alternatives to make mobile robots more accessible for small companies. The platform uses two distance sensors with different principles of measurement, ultrasonic and time-of-flight to detect distance to objects or obstacles of different shapes and colors. These two sensors are the MB1010 LV-MaxSonar-EZ1 ultrasonic sensor and the VL53L0XV2 time-of-flight (ToF) laser distance sensor. The information delivered by these two sensors is merged and processed in an Arduino Nano that is also responsible for receiving the data by serial I2C communication and sending it to Teensy3.6 at a higher plant level. Also, this controller has to take the information from a PMW3901 optical flow spatial location sensor, which provides the necessary information for inverse kinematics closed loop control. After this, the acquired data is sent to the central control system (Jetson Nano), which also connects via USB3.0 protocol to an Intel Realsense D435 depth camera. The camera gathers depth image information to know the robot surroundings and perform the route calculation parallel with the displacement, achieving a VSLAM (Visual Simultaneous Localization and Mapping) control. The Realsense also delivers an Red-Green–Blue image to the central system, which can be used for other automatic navigation algorithms. Sensorial Scheme can be visualized in Fig. 1.

2.2.3. Controllers

Each mechanical module that supports the distance sensors has an Arduino Nano development board that receives the data by serial I2C communication, as mentioned above, to then process it and send it to the next level of communication control. At this point, the Teensy 3.6 receives information from the optical flow sensor and each of the low-level microcontrollers to establish a safe movement protocol with the central controller (Jetson Nano). Such data corresponds to distance measurements from each of the low-level sensory modules, as well as the concerning robot speed movement information. Finally, the complete system is assembled together with electronic cards, sensors, batteries and power electrical circuits, which are better explained in the 5 section and is shown in Fig. 2.

---

**Fig. 1.** Arduino Nano assembled with sensor module. SDA (Serial Data) and SCL (Serial Clock) pins are the transmission and receptions lines used for the I2C communication protocol.
3. Design files

3.1. Design Files Summary

| Design file name                  | File type | Open source license | Location of the file |
|----------------------------------|-----------|---------------------|----------------------|
| F1_exploded jpg                  | jpg       | GNU GPL v3.         | https://osf.io/f3nrp/ |
| F2_exploded jpg                  | jpg       | GNU GPL v3.         | https://osf.io/rth5a/ |
| F3_exploded jpg                  | jpg       | GNU GPL v3.         | https://osf.io/kwn7y/ |
| F4_exploded jpg                  | jpg       | GNU GPL v3.         | https://osf.io/d2qx6/ |
| F101SensorSupportLeft x2 stl     | stl        | GNU GPL v3.         | https://osf.io/nmhfb/ |
| F101SensorSupportRight x2 stl    | stl        | GNU GPL v3.         | https://osf.io/qs54u/ |
| F102SensorBase x2 dxf            | dxf       | GNU GPL v3.         | https://osf.io/peyr9/ |
| F102SensorBase x2 stl            | stl        | GNU GPL v3.         | https://osf.io/46pjd/ |
| F102TripleSensorBase x4 stl      | stl        | GNU GPL v3.         | https://osf.io/5m49p/ |
| F103MainBase stl                 | stl        | GNU GPL v3.         | https://osf.io/3nx4t/ |
| F201StructuralLink stl           | stl        | GNU GPL v3.         | https://osf.io/pn84s/ |
| F202AngleBeam x4 stl             | stl        | GNU GPL v3.         | https://osf.io/xs2ev/ |
| F203AssemblyAngle x4 stl         | stl        | GNU GPL v3.         | https://osf.io/sbuaj/ |
| F204Bushing x20 stl              | stl        | GNU GPL v3.         | https://osf.io/ew4jz/ |
| F205StructuralBeam x6 stl        | stl        | GNU GPL v3.         | https://osf.io/9rta7/ |
| F301PyramidBase stl              | stl        | GNU GPL v3.         | https://osf.io/xzmyyp/|
| F302RailBase x4 stl              | stl        | GNU GPL v3.         | https://osf.io/3euf3/ |
| F303StructuralSupport stl        | stl        | GNU GPL v3.         | https://osf.io/y95uf/ |
| F304GeneralSupport stl           | stl        | GNU GPL v3.         | https://osf.io/x8v5j/ |
| F305BatterySupport stl           | stl        | GNU GPL v3.         | https://osf.io/h7xjt/ |
| F306DinRail x4 stl               | stl        | GNU GPL v3.         | https://osf.io/yeu8s/ |
| F402MetallicSupport x4 stl       | stl        | GNU GPL v3.         | https://osf.io/p3myi/ |
| F403DampingArm x4 stl             | stl        | GNU GPL v3.         | https://osf.io/f46jx/ |
| F404ShockAbsorber stl             | stl        | GNU GPL v3.         | https://osf.io/n8v6b/ |
| F405LowerTrim x4 stl              | stl        | GNU GPL v3.         | https://osf.io/szczw/ |
| GeneralSchematic png              | png        | GNU GPL v3.         | https://osf.io/hmgur/ |
| NanoSchematic png                 | png        | GNU GPL v3.         | https://osf.io/hq2bp/ |
| InverseK_Teensy ino               | ino        | GNU GPL v3.         | https://osf.io/c2my9/ |
| Sensor_ArduinoNano ino            | ino        | GNU GPL v3.         | https://osf.io/k9xb/  |

![Fig. 2. Controllers scheme. Similar to SDA and SCL, TX (Transmission) and RX (Reception) are used for Serial Communication protocol.](image-url)
4. Bill of materials

Table 2
Bill of hardware and electronic components and suppliers web page.

| Design File Name                  | Qty | Price  | Source of material                   |
|----------------------------------|-----|--------|--------------------------------------|
| Jetson NANO                      | 1   | $89.99 | http://t.ly/K4pu                     |
| VLS3L0XV2 range laser sensor     | 4   | $14.95 | http://t.ly/hMYP                     |
| Depth Camera D435                | 1   | $179.00| http://t.ly/tvWxz                    |
| Maxsonar Ultrasonic range sensor | 4   | $25.95 | http://t.ly/Znzt                     |
| 12 V 7.8Ah Rechargeable battery  | 1   | $23.00 | http://t.ly/l0BF                     |
| H-Bridge module driver           | 4   | $4.50  | http://t.ly/Ml4                      |
| Faulhaber 12 V Motor             | 4   | $4.50  | http://t.ly/Whzl                     |
| Arduino Nano                     | 4   | $20.00 | http://t.ly/sWJb                     |
| PJRC Teensy 3.6                  | 1   | $30.00 | http://t.ly/1Mu0                     |
| Stainless steel bristol screws   | 520 | $24.99 | http://t.ly/sDOg                     |
| and nuts box [M3, M4, M5, and M6] |     |        |                                      |

5. Build instructions

The mechanical structure was designed to be a modular assembly, so each part can be manufactured individually, allowing easy parts construction and replacement when needed. The assembly procedure is not sequential and can be done in any desired order, using bristol head stainless steel screws. The type of thread for screws (millimeter or Whitworth) is not very relevant as long as the external diameter is a little bit smaller than the element holes. This is due to the fact that none of the robot's holes depend on threads made in the elements, all the holes are smooth and therefore, all the screws are fastened with nuts. This characteristic is an advantage in the assembly because it does not depend on a specific type of thread, nor on how much a thread yields in a polymeric or metallic material over time. To approach stainless steel Bristol screws with a flat head, with millimeter thread M2, M3, M4, M5, and M6 were used. All additional mechanical elements fit into these diameters such as bearings, nuts and bushings. Finally, it is important to indicate that all these elements are shown in Table 2.

Initially, it is a good start to assemble the main structure, which comprises the 3 bases of acrylic material on which the battery and other electronic devices are supported, including the general supports of the Jetson Nano controller and the depth camera. It is necessary to clarify that the easiest way to assemble the robot is to assemble each main module separately and then put them together. In the Desing Files Summary Table, there are images in JPG and PNG format about the main modules, named as follows: main control structure, plant sensory structure, mechanical support structure and electromechanical motor system. An image of the mechanical support system is shown in Fig. 3.

Dealing with issues of the attached codes, two algorithms will be used: one for the arduinos nano, which is responsible for acquiring, calibrating, conditioning and processing the distance sensors and then sending them by serial communication to the Teensy embedded system, and the code from Teensy board, which manages the information provided by the arduinos and performs the inverse kinematics algorithms with closed loop control (See Figs. 4–6).

In the Arduino Nano code, there is a class called module, which executes initializing, calibrating and merging the sensory data through different modules and functions, and then sending the information through a serial communication.

In the code of the Teensy board, initially different variables and functions are declared, which are specific to the code for its operation, variables that are not relevant when understanding the code roughly. There are functions such as usb coordinates that allow to receive and send position coordinates with another development card if requested. In addition, there are full motion functions to move forward, backward, stop, turn, and more. There are check flags, for when the robot in the inverse kinematics method, confirms when it reached its goal in both X and Y, and uses functions such as readsensors(), to read the sensory data sent by the arduinos Nano and check that the robot's surroundings are available to navigate. The inverse kinematics function also calculates the mathematics necessary to operate the robot in an omnidirectional way and reach the objective in the shortest possible time. Finally, in the last section of the code there is a function that allows you to control the robot remotely via bluetooth, either from an Android application or with a remote control with specific commands.

Each motor is connected to a 3A H-Bridge power circuit, which is used to regulate its speed and direction of rotation. Each H-Bridge circuit is connected to the 12V 7.8Ah battery, which provides enough current for the entire robot operation, and to the main Teensy 3.6 plant-level controller. To assemble the electronics and camera support structure to the platform, acrylic bases cut with laser manufacturing techniques were used, a large one assembled below as a support and two at the front and back, to support the sensor modules. In the figure, 1 refers to the couplings to the structure, 2 to the acrylic supports of the sensors and 3, to the acrylic base of the battery and chamber structure.

The elements that connect the main assembly structure, is 3D printed with an 80% hexagonal infill, giving it enough resistance for the structure to flex, thus providing stability to movement and sensors as well. In the figure, 1 refers to the main
links of the structure, 2, 3, 4 and 5 are coupling links between structures to give greater stability, making it more achievable and efficient.

The assembly of the camera support structure comprises a rail to adjust the height of the camera, the rail is a standard DIN rail, with Clips to fasten the metal and provide the necessary stability to the camera is stable. In addition, it has a support for the battery, a 7.8 Ah battery that can keep the robot running from 1 to 2 h depending on the area of movement and navigation. In the figure, 1 is the depth chamber support pyramid, 7 is the camera and 2 and 6 are part of the rail structure to raise or lower the height of the pyramidal support. On the other hand, 3, 4 and 5 are the pieces that protect and house the battery and the jetson nano board.

Finally, the shock absorber structure is assembled, placing the Faulhaber motors with the mecanum wheels and the damping arms. Pieces 1 and 4 belong to the parts that are purchased in the industry, which are the motors, the mecanum wheels, and the hydraulic shock absorbers. Part 2 is a metal element to couple the motor to the damping arm, it was designed in this material since in this section it is not possible to make a change in geometry to support the torque of each motor, so it was decided to improve the material. Part 6 finishes coupling to the motor damping arm (Parts 3 and 4), as shown in Fig. 7.

6. Operation instructions

The robotic platform performs two operations modes, this is to bring adaptability to different research and operation scenarios.

6.1. Manual operation

The manual operation mode focuses on operating the robot via blue-tooth with commands in ASCII code from an Android application. It has ten operation buttons: four arrow-shaped buttons to direct the robot omnidirectionally to the north,
south, east and west; four buttons in the shape of a diagonal arrow to execute a diagonal movement at northeast (45°), north-west (135°), south-west (225°) and south-east (315°). And finally, two buttons to rotate the robot clockwise and anti-clockwise direction in its own gravity center. The way in which the wheels move to execute these movements can be consulted in [21,22] (is controlled automatically by pressing the buttons). Additionally, the robot constantly consults the distance sensors to know if, even when operated manually, it is going to collide with an obstacle and in that case, stop its movement to avoid crashes.

6.2. Automatic

The robot executes its automatic movement based on the closed-loop inverse kinematics algorithm. Inverse kinematics, calculates the movement of the wheels with a specific path, to reach a point desired by the user. The user enters a planar Cartesian position (2D) end point of movement for the robot and proceeds to make his way to that point. The feedback of the closed loop is achieved with the optical flow sensor assembled in the robot, which is responsible for informing the controller how much it has moved on the flat surface. The inverse kinematics algorithm uses the advantages of the omnidirectional configuration that the Mecanum wheels present. The inverse kinematics equation has values A and B (Shown in Fig. 8), which come from the measurements of width and length of the robot, taking them from the support points of the robot on the surface, that is, it is measured from the axis of one wheel to another, both from wide as long. The inverse kinematics equation returns the speed at which each of the wheels have to go to reach a specific Cartesian coordinate (Cartesian X and Y input values). The equation is expressed as follows, in Eq. 1, being \( \omega_i \) the angular velocity from each wheel, and \( R \) the wheel radius too:

\[
\begin{bmatrix}
\omega_1 \\
\omega_2 \\
\omega_3 \\
\omega_4
\end{bmatrix} = \frac{1}{R} \begin{bmatrix}
1 & -1 & - (A + B) \\
1 & 1 & (A + B) \\
1 & 1 & - (A + B) \\
1 & 1 & (A + B)
\end{bmatrix} \begin{bmatrix}
x \\
y \\
\omega_r
\end{bmatrix}
\]

The algorithm uses this equation in the motion section of the code, where it is applied to calculate the angle of the robot’s motion and, at the same time, transform the desired motion to the angular velocity of each of the wheels.

Finally, the design, manufacturing of the parts, the connection and the assembly of the robot were developed. A real photo of the assembly is shown in Fig. 9.
7. Validation and characterization

To assess robot motion performance in closed-loop, infrared marks were assembled to the platform to be detected by the Optitrack motion capture system. Eight displacements forming an asterisk on the Z plain were evaluated: north, south, east, west, northeast, northwest, southeast, and southwest (0°, 45°, 90°, 135°, 180°, 225°, 270°, and 315°). Such a route allows omnidirectional movement evaluation of the platform with the control, when moving diagonally, vertically, and horizon-
The robot moves forward and backward at each direction of the asterisk, achieving a total of 16 trajectories (adding those that go and those that come) to evaluate the repeatability too. The results are shown in Fig. 10.

The performance of the robot was also measured on square paths, proving that the mecanum wheels have a slippage between the surface and the polymer of the inclined rollers. Another path is shown in Fig. 11.

The results show that the robot motion has an overall good performance and can correctly follow the desired path with an accuracy up to 0.0081 m. Nevertheless, some wrong or non desired motion was experienced when moving in the 45° direction, as the robot experienced a turn at the end of the northeast forward displacement. Such behavior corresponds to a system response to self-correct his path due to drifting or irregular floor. The accuracy of the system is shown in Table 3.

Finally, to evaluate the function of the damping system, a test was carried out where obstacles of 7, 5, 4, and 1 cm were located in a straight line and the trajectory of the robot. A triaxial accelerometer was used to carry out the observation, but
only the acceleration in the Z-axis was considered. In Fig. 12, the comparison of the acceleration with the shock absorb (red line) and without the shock absorb (black line) is presented, to make the observation without shock-absorbing, an element that would impede its function was fixed. The experiment shows that the vibration on the system is significantly reduced and the amplitude of the acceleration is also reduced.

Although the overall accuracy of the system is acceptable. Environmental factors like mud, liquids, or dirt can cause drifting or slides that might lead to a biased final location.

The robotic platform integrates a set of distance sensors that can be used to avoid obstacles. Nevertheless, due to the modular design of the platform, several sensors can be integrated, this allows to upgrade the system or adjust its function to different specific tasks.

The robotic platform has a damping system to mitigate structural damage, smooth movement and reduce measurement error of sensors and depth chamber.

--

**Table 3**

| Trajectory | Absolute Error[m] |
|------------|-------------------|
| Horizontal | 0.0620            |
| Vertical   | 0.0134            |
| Diagonal   | 0.0081            |

**Fig. 10.** Evaluation of omnidirectional movement of the platform using the asterisk path and Optitrack system. The blue line corresponds to the robot displacement measured by the Optitrack, while the red line belongs to the ideal trajectory.

**Fig. 11.** Square trajectory using omnimove through vertical and horizontal axis. Real trajectory in red, experimental trajectories in blue..
In an embedded development system such as the jetson nano, different advanced navigation algorithms can be implemented, since it has a high processing capacity with respect to its energy consumption (3A Max).

**Human and animal rights**

Although, the tests were conducted under human presence, no human or animal studies were conducted in this work.

**Declaration of Competing Interest**

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

**Acknowledgments**

This study were supported by the Sistemas de Control y Robótica (GSCR) Group COL0123701, at the Sistemas de Control y Robótica Laboratory, attached to the Instituto Tecnológico Metropolitano and the project "FORTALECIMIENTO Y CONSOLIDACIÓN DEL GRUPO AUTOMÁTICA, ELECTRÓNICA Y CIENCIAS COMPUTACIONALES PARA RESPONDER A LAS NECESIDADES DE LAS INDUSTRIAS 4.0" code P21101.

**References**

[1] Mostafa Sharifi, Matthew Samuel Young, Xiao Qi Chen, Don Clucas, Christopher Pretty, Mechatronic design and development of a non-holonomic omnidirectional mobile robot for automation of primary production, Cogent Eng. 3 (1) (2016).

[2] Seok Ju Lee, Jongil Lim, Cirma Tewoldu, Jeroek Kwon, Autonomous tour guide robot by using ultrasonic range sensors and QR code recognition in indoor environment, in: IEEE International Conference on Electro Information Technology, 2014, pp. 410–415.

[3] Jacqueline Lee Fang Ang, Wai Kong Lee, Boon Yaik Ooi, Thomas Wei Min Ooi, Location Sensing using QR codes via 2D camera for Automated Guided Vehicles., in: 2020 IEEE Sensors Applications Symposium, SAS 2020 – Proceedings, 2020.

[4] Payam Nazemzadeh, Daniele Fontanelli, David Macii, Luigi Palopoli, Indoor localization of mobile robots through QR code detection and dead reckoning data fusion, IEEE/ASME Trans. Mechatron. 22 (6) (2017) 2588–2599.

[5] Julio Vega, José M. Cañas, PiBot: an open low-cost robotic platform with camera for STEM education, Electronics (Switzerland) 7 (12) (2018).

[6] Brent Utter, Alexander Brown, Open-source five degree of freedom motion platform for investigating fish-robot interaction, HardwareX 7 (2020) e00107.

[7] Calvin Gregory, Andrew Vardy, microUSV: a low-cost platform for indoor marine swarm robotics research, 2020.

[8] Daniel F. Carlson, Alexander Führerling, Lasz Vesterley, Mathias Skovby, Simon Sjöer Pedersen, Claus Melvad, Saren Rysgaard, An affordable and portable autonomous surface vehicle with obstacle avoidance for coastal ocean monitoring, HardwareX 5 (2019) e00059.

[9] S.D. Grigorescu, F.C. Argatu, S.V. Paturca, C. Cepisca, G.C. Seritan, F.C. Aduochie, B. Enache, Robotic platform with medical applications in the smart city environment, in: 2019 11th International Symposium on Advanced Topics in Electrical Engineering, ATEE 2019, 2019, pp. 1-6.

[10] Jean Luc Liardon, D.A. Barry, Adaptable imaging package for remote vehicles, 2017.

[11] Jingjing Jiang, Christian Claudel, A high performance, low power computational platform for complex sensing operations in smart cities, 2017.

[12] Guillel Camprodon, Òscar González, Victor Barberán, MÁximo Pérez, Viktor Smai, Miguel Angel de Heras, Alejando Bizzotto, Smart Citizen Kit and Station: An open environmental monitoring system for citizen participation and scientific experimentation, 2019.

[13] J. S. Botero-Valencia, J. Valencia-Aguirre, D. Durmus, A low-cost IoT multi-spectral acquisition device, 2021.

[14] Mainardo Gaudenzi Asinelli, Moisés Serra Serra, Jordi Serra Espaullella, The smARTS_Museum_V1: An open hardware device for remote monitoring of Cultural Heritage indoor environments, 2018.

[15] Sara Bragança, Eric Costa, Ignacio Castellucci, Pedro M. Arezes, A brief overview of the use of collaborative robots in industry 4.0: Human role and safety, Stud. Syst. Decis. Control 202 (2019) 641–650.

**Fig. 12.** Acceleration comparison in the Z axis.
[16] Rinat Galin, Roman Meshcheryakov, Automation and robotics in the context of Industry 4.0: the shift to collaborative robots, IOP Conf. Ser.: Mater. Sci. Eng. 537 (3) (2019) 2–7.
[17] Ksenia Shabalina, Artur Sagitov, Evgeni Magid, Comparative analysis of mobile robot wheels design, in: Proceedings – International Conference on Developments in eSystems Engineering, DeSE, 2018-September, 2019, pp.175–179.
[18] Masaki Takahashi, Takaumi Suzuki, Francesco Cinquegrani, Rosario Sorbello, Enrico Pagello, A mobile robot for transport applications in hospital domain with safe human detection algorithm, in: 2009 IEEE International Conference on Robotics and Biomimetics, ROBIO 2009, 003, 2009, pp.1543–1548.
[19] Tianran Peng, Jun Qian, Bin Zi, Jiakui Liu, Xingwei Wang, Mechanical Design and Control System of an Omni-directional Mobile Robot for Material Conveying, Proc. CIRP 56 (2016) 412–415.
[20] Matthew Spenko, Yu. Haoyong, Steven Dubowsky, Analysis and design of an omnidirectional platform for operation on non-ideal floors, Proc. IEEE Int. Conf. Robot. Autom. 1 (February) (2002) 726–731.
[21] Yiqun Liu, Hao Li, Liang Ding, Leilei Liu, Tao Liu, Jianfeng Wang, Haibo Gao, An omnidirectional mobile operating robot based on mecanum wheel, in: 2017 2nd International Conference on Advanced Robotics and Mechatronics (ICARM), 2017, pp. 468–473.
[22] Hon Ming Yip, Xin Kwan Ho, Man Hin Anson Chu, King W.C. Lai, Development of an omnidirectional mobile robot using a RGB-D sensor for indoor navigation, in: 4th Annual IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems, IEEE-CYBER 2014, 2014, pp. 162–167.

D. Betancur-Vásquez Mechatronic Engineer and MSc (e) in Automation and Industrial Control of Instituto Tecnológico Metropolitano in Medellin Colombia, with experience in robotics and control systems, specifically in mobile robotics, mechanical prototyping and electronic development. He is currently a Professor in the Department of Electronics and Telecommunications of the Faculty of Engineering of the Metropolitan Technological Institute.

M. Mejía-Herrera Mechatronic Engineer and MSc (e) in Automation and Industrial Control of Instituto Tecnológico Metropolitano in Medellin Colombia, Member of the research group classification A1 “Automática, Electrónica y Ciencias Computacionales” since 2018 with experience in artificial vision, embedded systems, IoT and 3D printing for the development of low-cost prototypes.

J.S. Botero-Valencia Magister in Automation and Industrial Control, and PhD in Engineering, has experience in control systems and robotics, specifically in the Internet of Things (IoT) and mobile robotics. He currently works as a Professor in the Department of Mechatronics and Electromechanics of the Faculty of Engineering of the Metropolitan Technological Institute, and belongs to the Laboratory of Control Systems and Robotics.