World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:14, No:11, 2020

Low-Cost Mechatronic Design of an Omnidirectional Mobile Robot

S. Cobos-Guzman

***Abstract*—**This paper presents the results of a mechatronic design based on a 4-wheel omnidirectional mobile robot that can be used in indoor logistic applications. The low-level control has been selected using two open-source hardware (Raspberry Pi 3 Model B+ and Arduino Mega 2560) that control four industrial motors, four ultrasound sensors, four optical encoders, a vision system of two cameras, and a Hokuyo URG-04LX-UG01 laser scanner. Moreover, the system is powered with a lithium battery that can supply 24 V DC and a maximum current-hour of 20Ah.The Robot Operating System (ROS) has been implemented in the Raspberry Pi and the performance is evaluated with the selection of the sensors and hardware selected. The mechatronic system is evaluated and proposed safe modes of power distribution for controlling all the electronic devices based on different tests. Therefore, based on different performance results, some recommendations are indicated for using the Raspberry Pi and Arduino in terms of power, communication, and distribution of control for different devices. According to these recommendations, the selection of sensors is distributed in both real-time controllers (Arduino and Raspberry Pi). On the other hand, the drivers of the cameras have been implemented in Linux and a python program has been implemented to access the cameras. These cameras will be used for implementing a deep learning algorithm to recognize people and objects. In this way, the level of intelligence can be increased in combination with the maps that can be obtained from the laser scanner.

***Keywords*—**Autonomous, indoor robot, mechatronic, omnidirectional robot.

I. INTRODUCTION

THE new industry 4.0 requires high levels of digitalization for increasing the level of intelligence and make smart decisions in the automation ecosystem. Hence, the industry of logistics requires smart mobile robots that can interact with humans and robots to deliver products from points of loading and unloading. In this scenario, several robots and humans could collaborate sharing the same environment. Therefore, mobile robots should recognize other robots and other people for making smart decisions during the trajectory or in the points for delivering the materials or loads.

To reach the previous point, this paper presents an omnidirectional robot with a combination of sensors that can be used to navigate autonomously and recognize other robots, objects, areas, and people using the vision system. The proposed mechatronic design uses 4 powered omnidirectional wheels in a similar layout like the configuration of The Carnegie Mellon Uranus robot [1] or the model of youBot from KUKA [2], [3]. This configuration is not the only one when we talk about these types of robots since there are currently different wheel configurations and layouts such as robots with 3 omnidirectional wheels [4] or robots with 4 wheels but with a different layout [5]. These different layouts and configurations are possible thanks to the kinematic properties of the omnidirectional wheel [6]. Moreover, using the layout of 4 powered omnidirectional wheels, the robot can use the modes of control that it will describe in section III. These modes of movement are useful in logistic applications where the high density of robots and people are high. On another hand, for achieving more intelligence in the robot, the designed robot has a vision system that can be used for detecting other robots or humans, in this way the robot cannot only use sensors for collision avoidance. Thus, if the robot requires recognizing a human for a collaboration task, the visual recognition system is used for achieving this objective.

The prototype uses low costs open-source hardware for controlling the system. The use of open-source hardware is important because it can help develop more devices and share with the community new modules that roboticists can use for the future design of robots such as ROS [7].

The edge devices selected have limitations in terms of memory and computational hardware [8]. However, a good selection of real-time hardware and light software is possible to control all the devices that the robot can use. Thus, this effort of integration will help in the future generate more robots based on edge devices. Finally, image recognition helps increase the level of intelligence of the robot in combination with the rest of the sensors.

The structure of the paper is as follows: the mechatronic design of the robot is described in Section II; the modes of control of the robot are defined in Section III; the software system based on ROS modules are presented in Section IV; the stereo vision system and the recognition system based on deep learning is described in Section V. Finally, the conclusions are presented in Sections VI.

II. MECHATRONIC DESIGN

The mechatronic design of the robot is based on 4 omnidirectional Mecanum wheels, two of them have an angle of 45 degrees (right wheel) and two of them have an angle of -45 degrees (left wheel). The wheels are placed crosswise so that the angles of the rollers can coincide, as shown in Fig. 1. Each wheel is powered by a DC gear motor with a speed up to 130 rpm, operating at 24 V of voltage. Each motor can provide 3 Nm and uses 3 A of current in the air with an increase of temperature of 70 °C.

S. Cobos-Guzman is with the Engineering Department, Universidad Internacional de La Rioja, Rectorado – Avda. de la Paz, 137, Logroño, La Rioja, Spain (phone: +34 941 210 211; e-mail: salvador.cobos@unir.net).

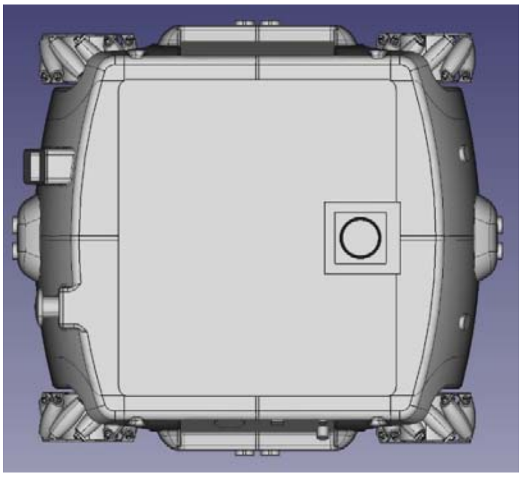


Fig. 1 CAD design of the omnidirectional robot

Each motor (see Fig. 2) has incorporated an optical encoder bidirectional with 25 pulses for every turn. Thus, these encoders are used for knowing the number of turns and the direction of each turn for controlling the position.

Fig. 2 Gear motor with optical encoder for powered each wheel

Fig. 3 shows the initial distribution of the motors for transmitting movement through a conical gear base transmission. The blue part from Fig. 3 represents the lithium battery of the system. This battery can supply a voltage of almost 30V with a maximum current-hour of 20Ah for the entire power system.

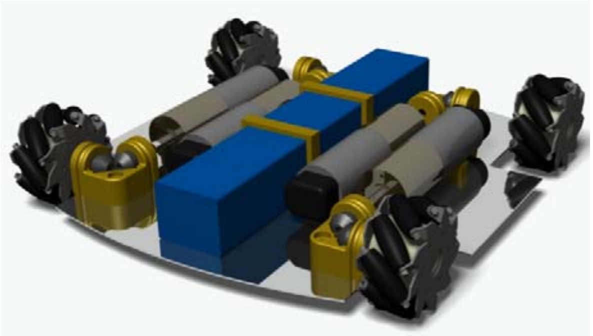
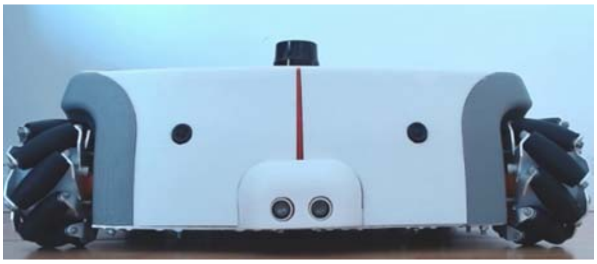


Fig. 3 Initial design to transmit movement to the 4 wheels using 4 DC gear motors

The sensory perception of the robot is based on 4 ultrasonic sensors HC-SR04 [9] used for collision avoidance, 1 Hokuyo URG-04LX-UG01 laser scanner [10] used for the autonomous navigation, and two TTL serial cameras Adafruit [11] for recognizing objects in the environment.



(a)

(b)

(c)

(d)

Fig. 4 Omnidirectional robot-built design

Fig. 4 (a) shows the laser scanner on the top (Hokuyo sensor), the first ultrasonic sensor on the bottom, and the two cameras in the middle of the front panel. Fig. 4 (b) shows the main power button on the left, the emergency button on the right, and the second ultrasonic sensor on the bottom. Fig. 4 (c) shows the Raspberry Pi Touch Display [12] and the third ultrasonic sensor on the bottom. Finally, Fig. 4 (d) shows USB ports in the left, the battery charger port in the right, the fourth ultrasonic sensor on the bottom, and an auxiliary power button for activating the Raspberry Pi.

This robot has a combination of mechanized mechanical parts for the base and a combination of rapid prototyping parts for holding all the devices described in Fig. 4.

Measurements of the robot are: 430 (width) x 450 (long) x 150 (height) mm. The payload area to put objects on the top is 350 mm x 200 mm, and the maximum weight is 60 kg (15 kg for each wheel). Fig. 5 shows the payload area. Moreover, the robot has integrated the Arduino Mega 2560 [13] for controlling the motors, encoders, and ultrasonic sensors while the Raspberry Pi 3 Model B [14] is used to control the Hokuyo laser, and uses ROS as the main controller.

Fig. 5 Payload area for putting objects on the top of the robot

A. Power Distribution

Since the system is designed to operate at 24V, it is necessary to use a first DC-DC converter to supply 24V and provide 20A. The DC-DC converter used for this robot can provide power up to 480W. The Input could be from 18-36V, therefore if the voltage of the battery is going down (up to 18V), the DC-DC converter can keep a constant voltage of 24V with a maximum current of 20 A. This first output is used to supply the 4 motors giving place to an initial current of 12A only for moving the industrial motors.

The motors use an industrial encoder that requires a second DC-DC converter of an output of 6V. This voltage is useful because the Arduino Mega 2560 also requires a minimum voltage of 6V to energize the board.

The ultrasound sensors required another DC-DC converter of 5V. This led to the development of an electronic board in collaboration with the company Robotic-Scientific LTD [15]. This board has been designed for supplying the power of the 4 ultrasound sensors and the encoders. This board has four connectors for providing 24 V to the rest of the motors. Fig. 6 shows the electronic board to provide all the voltages required.

The last power to consider is the system that uses the Hokuyo URG-04LX-UG01 laser scanner, Raspberry Pi, 2 cameras, the display, and a fan for cooling the system. The system requires at least another DC-DC converter of 5V, 5A, and 25W.

Fig. 6 Electronic board designed to provide different voltages

According to the experiments, it is not recommended to use converters that can provide less than 5A and less than 25W. This power is required for all the peripherals used in the USB ports of the Raspberry Pi. Moreover, the power cable for the raspberry pi should support 5A as well; otherwise, the cable could burn.

III. COMMUNICATION AND LOW-LEVEL CONTROL

The robot has two edge devices as described in the previous section. The Arduino Mega is responsible in the first instance, to control the velocity of the motors generating a PWM signal to the motor driver, change the direction of rotation and stop the motors; the second task is to read the encoders and generate a counter for knowing the number of turns that the wheel is produced. The third task is to read the distance of the four ultrasonic sensors.

The Raspberry Pi is acting as the main controller under ROS and it is responsible to activate the 2 cameras, read the Hokuyo sensor and send commands to the Arduino Mega for sending information from the sensors or activate the modes of control of the robot. These communications among the two devices are produced through serial communication as shown in Fig. 7.

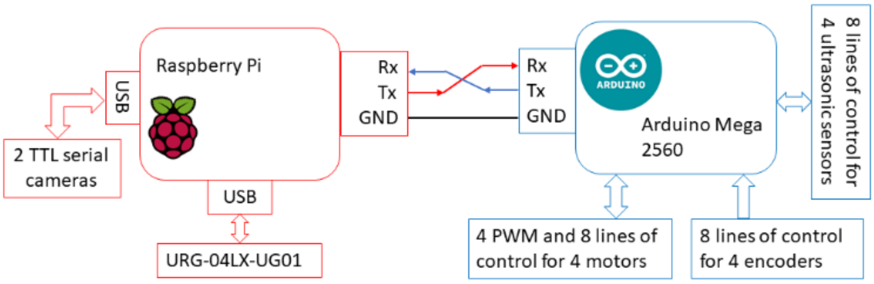
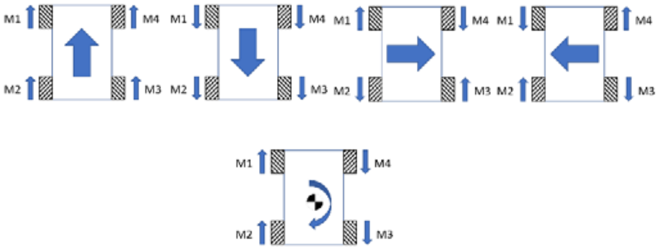


Fig. 7 System communication among all the devices

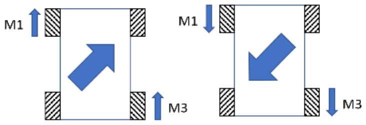
The robot has 11 modes of control that produce different types of movements depending on the direction of rotation of each wheel and the activation of the motors.

When the four motors are activated, the robot can produce five types of movements in the following order: forward, backward, lateral movement to the right, lateral movement to the left, and rotation regarding the center of the robot as shown in Fig. 8 (a).

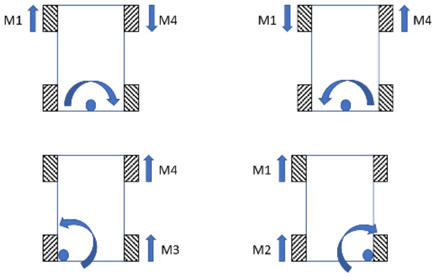
Diagonal movements can be produced when only two motors are activated (see Fig. 8(b)) or rotational movements with a different center of rotation as shown in Fig. 8 (c).



(a) Movements of forward, backward, lateral movement to the right, lateral movement to the left and rotation with respect to the center of the robot



(b) Diagonal movements



(c) Different rotations regarding different points of rotation

Fig. 8 Modes of control of the robot

he algorithm for controlling these 11 modes of control has been implemented in the Arduino and the node of ROS can be accessed to each mode via serial communication. Moreover, the algorithms for reading the ultrasonic sensors and encoders were implemented in the Arduino board and rosserial [20] protocol was used for activating the nodes in ROS.

IV. SOFTWARE SYSTEM

The operating system implemented in the robot is Ubuntu version 18.04 for 64 bits arm systems (aarch64) [16]. The version of ROS is Melodic Morenia [17]. The version of Ubuntu allows using the capabilities of a 64-bit system.

The Hokuyo URG-04LX-UG01 laser scanner has been tested efficiently in ROS Melodic and visualized the first tests in the 3D visualization tool for ROS is Rviz [18]. Fig. 9 shows the first test of visualization of the robot in the Unified Robotic Description Format (URDF).

After the first experiments are carried out with the laser scanner, the next important node to test is the slam node in ROS.

Hector_Slam [19] was used for generating the first maps and checking the efficiency of the Raspberry Pi. Fig. 10 shows the results of the SLAM for generating the map of a room.

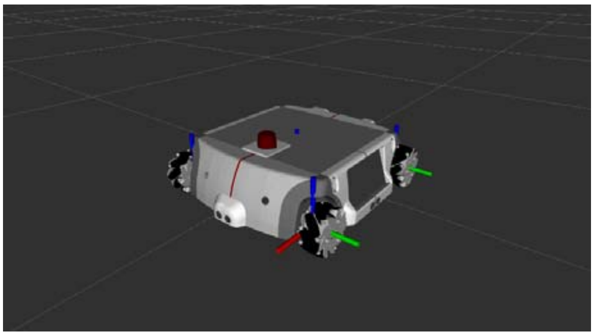
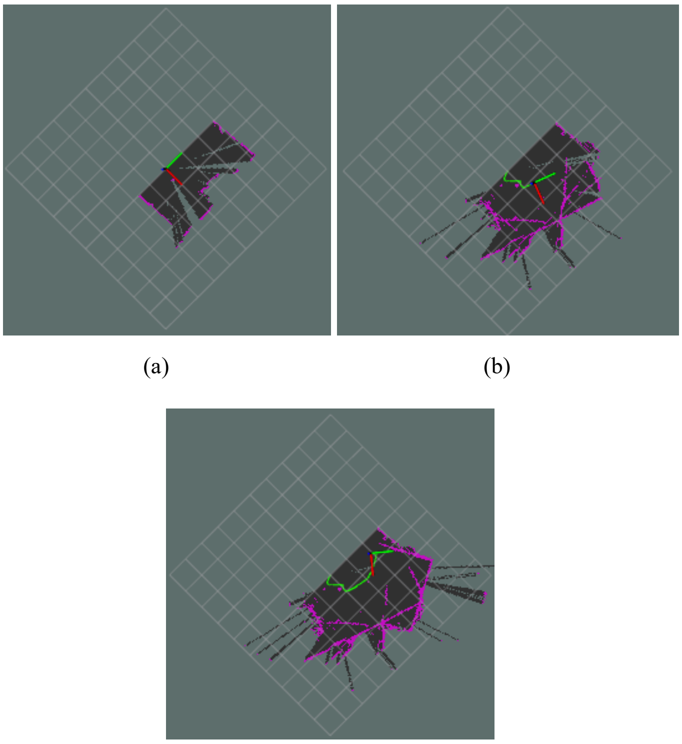


Fig. 9 First tests of the URDF model on Rviz



(a)

(b)

(c)

Fig. 10 First maps generated of the environment using the slam algorithm

V. STEREO VISION SYSTEM AND IMAGE RECOGNITION

A vision system is important to implement in autonomous robots because it can help increase the level of intelligence in autonomous navigation. Currently, there exist real industrial problems when an autonomous is disoriented due to false-negative pulses from the laser scanner and the robot does not know where to go. When this situation is produced, the operator must relocate the robot manually to a reference position or move it to the home position for restarting the navigation. Therefore, in these cases, the characteristic of autonomy is lost.

To avoid the previous situation, it is important to integrate more mobile robots with vision systems that can help to recognize the environment and make smart decisions.

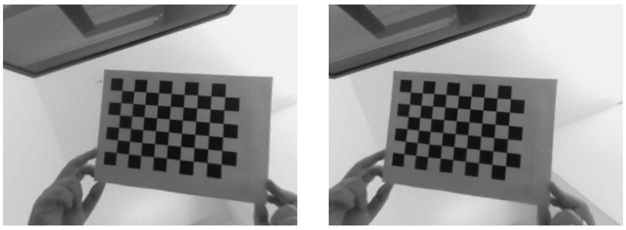
Below the stereo vision system and the recognition system of the robot are described.

A. Stereo Vision System

The stereo vision system must be calibrated to eliminate all the distortion on the cameras. Fig. 11 shows the big radial distortions that must be calibrated for obtained a correct deep map.

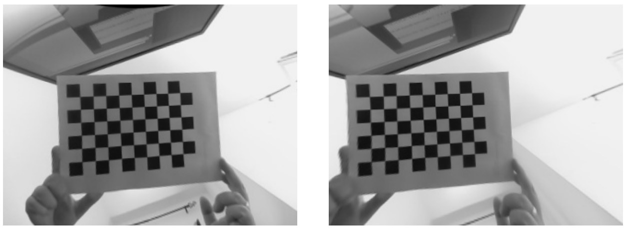
Fig. 11 Original images without calibration (left and right cameras)

For the calibration process, a 9 x 6 OpenCV chessboard [21] was used to taking 50 images for generating the matrix cameras and distortion coefficients. Fig. 12 shows the undistorted images for each camera and the calibration result can be shown in Fig. 13.



(a) left image (b) right image

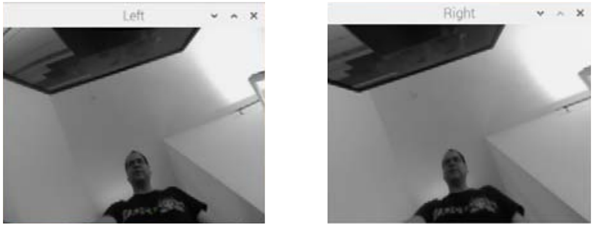
Fig. 12 Undistorted images



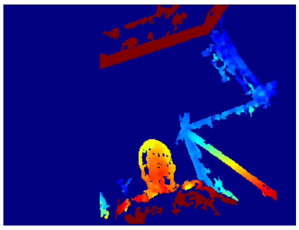
(a) left image (b) right image

Fig. 13 Calibrated images

Once the system is calibrated, a deep map can be tuned and generated. Figs. 14 and 15 show two examples of the generation of the deep map.

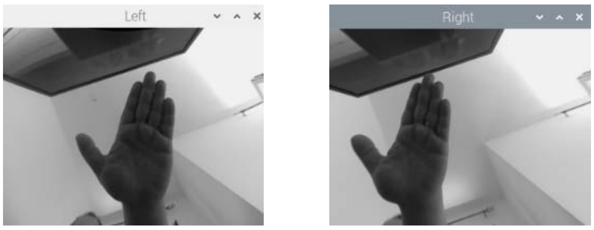


(a) left image (b) right image

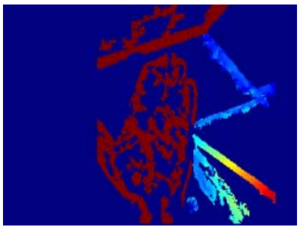


(c) deep map

Fig. 14 First deep map example with a person



(a) left image (b) right image



(c) deep map

Fig. 15 Second deep map example with a hand

The results of the deep map calculation efficiency were 20 FPS in real-time using the compute module 3+ of the raspberry pi [22] and a micro SD card that can read up to 95 MB/s and write speeds up to 90 MB/s.

B. Recognition System

A deep learning algorithm is implemented in OpenCV for testing the performance in the architecture aarch64 of Ubuntu 18.04 using python 3.6.

YOLO Real-Time Object Detection [23] and COCO dataset [24] were used for recognizing people and objects. The configuration files and weights of a pre-trained model were used in the deep neural network module from OpenCV (cv2.dnn.readNet). As an example, an efficient recognition of experts is implemented. Fig. 14 (a) was used for recognizing if there are experts in the image as is shown in Fig. 16. This recognition can be used for receiving commands or follow the person for a specific task. Thus, using these techniques, the neural network can be trained for recognizing people or areas of the industrial environment according to the application.

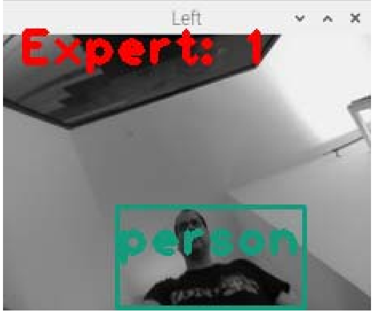


Fig. 16 Imagine recognition using a deep learning neural network

VI. CONCLUSION

The current technology of autonomous mobile robots that are used in the current industry requires more levels of intelligence for recognizing better the environment.

A vision system can help recognize the environment when the robot is disoriented due to false-negative pulses from the laser scanner. Thus, the system can recognize areas, characteristics from objects, and people in work areas that can be associated with the map. In this way, the autonomous robot cannot be lost in the environment. Therefore, the system proposed can perform an autonomous task with more artificial intelligence regarding classic robots that use only laser scanners and proximity sensors for collision avoidance. Moreover, the prototype can handle industrial motors with high currents in an efficient manner thanks to the low-level control implemented and it has been tested efficiently. This characteristic is especially important because in the industry motors like these can be used. Another aspect to consider is the hardware based on Raspberry Pi. These devices produce high temperatures and a proper cooling system is required, otherwise, it cannot work properly.

Finally, this robot was created using several open-source software and low-cost hardware that can help to minimize the cost of the robot.

ACKNOWLEDGMENT

This work was performed thanks to the private financial support of the Universidad Internacional de La Rioja.

REFERENCES

[1] R. Siegwart and I. Nourbakhsh, Introduction to autonomous mobile robots. Cambridge, Mass: MIT, 2004.

[2] "youBot wiki", Youbot-store.com, 2020. (Online). Available: http://www.youbot-store.com/wiki/index.php/Main_Page. (Accessed: 04- Apr- 2020).

[3] S. Ahmed, A. Topalov, N. Shakev and V. Popov, "Model-Free Detection and Following of Moving Objects by an Omnidirectional Mobile Robot using 2D Range Data", IFAC-PapersOnLine, vol. 51, no. 22, pp. 226-231, 2018. Available: 10.1016/j.ifacol.2018.11.546.

[4] H. Huang and C. Tsai, "Adaptive Trajectory Tracking and Stabilization for Omnidirectional Mobile Robot with Dynamic Effect and Uncertainties", IFAC Proceedings Volumes, vol. 41, no. 2, pp. 5383-5388, 2008. Available: 10.3182/20080706-5-kr-1001.00907.

[5] J. Song and J. Kim, "Energy efficient drive of an omnidirectional mobile robot with steerable omnidirectional wheels", IFAC Proceedings Volumes, vol. 38, no. 1, pp. 73-78, 2005. Available: 10.3182/20050703-6-cz-1902.01282.

[6] O. Diegel, A. Badve, G. Bright, J. Potgieter and S. Tlale, "Improved Mecanum Wheel Design for Omni-directional Robots", in Australasian Conference on Robotics and Automation, 2002, pp. 117-121.

[7] "ROS.org | Powering the world's robots", Ros.org, 2020. (Online). Available: https://www.ros.org. (Accessed: 04- Apr- 2020).

[8] M. Murshed, C. Murphy, D. Hou, N. Khan, G. Ananthanarayanan and F. Hussain, "Machine Learning at the Network Edge: A Survey", arXiv.org, 2020. (Online). Available: https://arxiv.org/abs/1908.00080. (Accessed: 04- Apr- 2020).

[9] "Ultrasonic Distance Sensor - HC-SR04 - SEN-15569 - SparkFun Electronics", Sparkfun.com, 2020. (Online). Available: https://www.sparkfun.com/products/15569. (Accessed: 04- Apr- 2020).

[10] "Scanning Rangefinder Distance Data Output/URG-04LX-UG01 Product Details | Hokuyo Automatic CO., LTD.", Hokuyo-aut.jp, 2020. (Online). Available: https://www.hokuyo-aut.jp/search/single.php?serial=166. (Accessed: 04- Apr- 2020).

[11] "TTL Serial Camera", Adafruit Learning System, 2020. (Online). Available: https://learn.adafruit.com/ttl-serial-camera. (Accessed: 04- Apr- 2020).

[12] “Raspberry-pi-touch-display”, Raspberrypi.org, 2020. (Online). Available: https://www.raspberrypi.org/products/raspberry-pi-touch-display/. (Accessed: 04- Apr- 2020).

[13] "Arduino - ArduinoMega2560", Arduino.cc, 2020. (Online). Available: https://www.arduino.cc/en/Guide/ArduinoMega2560. (Accessed: 04- Apr- 2020).

[14] “Raspberry-pi-3-model-b”, Raspberrypi.org, 2020. (Online). Available: https://www.raspberrypi.org/products/raspberry-pi-3-model-b/. (Accessed: 04- Apr- 2020).

[15] "Robotic-scientific ltd", Robotic-scientific ltd, 2020. (Online). Available: https://robotic-scientific.co.uk. (Accessed: 06- Apr- 2020).

[16] "Ubuntu ARM-RaspberryPi", 2020. (Online). Available: https://wiki.ubuntu.com/ARM/RaspberryPi. (Accessed: 26- Jun- 2020).

[17] "Melodic - ROS Wiki", Wiki.ros.org, 2020. (Online). Available: https://wiki.ros.org/melodic. (Accessed: 16- Apr- 2020).

[18] "rviz - ROS Wiki", Wiki.ros.org, 2020. (Online). Available: http://wiki.ros.org/rviz. (Accessed: 06- Apr- 2020).

[19] S. Kohlbrecher, J. Meyer, O. von Stryk and U. Klingauf, "A Flexible and Scalable SLAM System with Full 3D Motion Estimation", in Proc. IEEE International Symposium on Safety, Security and Rescue Robotics, 2011.

[20] "ros-drivers/rosserial", GitHub, 2020. (Online). Available: https://github.com/ros-drivers/rosserial. (Accessed: 17- Apr- 2020).

[21] "OpenCV", SourceForge, 2020. (Online). Available: https://sourceforge.net/projects/opencvlibrary/. (Accessed: 26- Jun-2020).

[22] Compute Module 3, 2020. (Online). Available: https://www.raspberrypi.org/products/compute-module-3/. (Accessed: 26- Jun- 2020).

[23] J. Redmon, "YOLO: Real-Time Object Detection", Pjreddie.com, 2020. (Online). Available: https://pjreddie.com/darknet/yolo/. (Accessed: 26-Jun- 2020).

[24] "COCO - Common Objects in Context", Cocodataset.org, 2020. (Online). Available: https://cocodataset.org/#home. (Accessed: 26- Jun-2020).

S. Cobos Guzman (B.Sc’03–A.I.E’09–A.S.D’07–Ph.D.’10) is Professor of Robotics, Cyberphysical Systems, and Artificial Intelligence in the Faculty of Engineering at the Universidad Internacional de La Rioja, Spain. He teaches in the master’s degree in Industry 4.0.

He received his first B.Sc. in Industrial Robotics from National Polytechnic Institute (I.P.N.), Mexico, in July 2003. The second Engineering degree received is in Automation and Industrial Electronics (A.I.E.) from Polytechnic University of Madrid (U.P.M), Spain, in 2009. Also, he received an Advanced Studies Diploma (A.S.D.) in Robotics and Automation from Polytechnic University of Madrid (U.P.M), Spain, in 2007, and his Ph.D. with honors (“Sobresaliente Cum Laude”) in Robotics and Automation from Polytechnic University of Madrid (U.P.M), Spain, in 2010.