Online-Journals.org

iJOE International Journal of Online and Biomedical Engineering

iJOE | eISSN: 2626-8493 | Vol. 22 No. 2 (2026) | OPEN ACCESS

https://doi.org/10.3991/ijoe.v22i02.58423

PAPER

Autonomous Mobile Robot Prototype for Vegetative Furrows Following Based on Optimal Control over ROS

Henry B. Guerrero[1] (✉), **Juan J. López A.**[2], **Andrés L. Jutinico**[2]

[1]BITPOINTER SAS, Villavicencio, Colombia

[2]Universidad Distrital Francisco José de Caldas, Bogotá, Colombia

hbguerrero@ieee.org

ABSTRACT

This study focuses on the development of autonomous reactive navigation strategies for mobile robots that must navigate between agricultural crop rows using optimal control techniques. Therefore, we configured a differential-drive mobile robot that navigates the crop row. Designed, implemented, and tested a linear quadratic regulator (LQR) and linear quadratic Gaussian (LQG) controller for a mission related to the following of crop rows. Sensors such as a laser light detection and ranging (LiDAR) sensor for distance measurement, an inertial measurement unit (IMU) that measures orientation, and encoders that measure motor angular speeds are included in the robot. To integrate sensors, actuators, and control algorithms, the robot operating system (ROS) framework was used as middleware. There is a detailed discussion of the methodology for utilizing sensors and actuators and implementing optimal control algorithms. Results obtained in this study demonstrate the effectiveness of the LQR and LQG controllers in maintaining the robot's trajectory and correcting deviations caused by disturbances.

KEYWORDS

precision agriculture, optimal controllers, autonomous mobile robot, laser light detection and ranging (LIDAR), inertial measurement unit (IMU), noise, RMSE, robot operating system (ROS)

1 INTRODUCTION

Agriculture is one of the most critical sectors of the world economy, as it provides the food and raw materials needed to sustain the world's population. However, the agricultural industry faces significant challenges, such as the need to increase productivity, reduce environmental impact, and improve working conditions. These challenges are especially pressing in the context of global population growth, climate change, and the increasing demand for sustainable practices. In response, modern agriculture has turned to technological innovations, particularly robotics and automation, to address these issues and improve efficiency [1, 2, 3, 4]. Recent research has illustrated the diverse applications of these technologies. For instance,

Guerrero, H. B., López A. J. J., Jutinico, A. L. (2026). Autonomous Mobile Robot Prototype for Vegetative Furrows Following Based on Optimal Control over ROS. *International Journal of Online and Biomedical Engineering (iJOE)*, 22(2), pp. 4–20. https://doi.org/10.3991/ijoe.v22i02.58423

Article submitted 2025-08-26. Revision uploaded 2025-11-17. Final acceptance 2025-11-17.

[4] implemented IoT-based robotic greenhouse systems with zoned automation to optimize tomato cultivation. Furthermore, [5] underscored the significance of user-centered design in the development of mobile agricultural monitoring platforms. Additionally, [6] emphasized that the successful integration of such innovations by agri-food microenterprises is contingent upon training that is tailored to local needs and the digital literacy of the workforce.

The integration of robotics in agriculture, often referred to as agricultural robotics, has the potential to revolutionize the industry. Robots can perform tasks such as planting, harvesting, and monitoring crops with precision and consistency, reducing reliance on manual labor and minimizing waste of resources [2, 3]. However, the deployment of agricultural robots requires advanced control systems to ensure precise navigation, obstacle avoidance, and task execution in dynamic, unstructured environments [1, 3].

According to [7], in agriculture, plant density refers to the number of plants per unit area of ground. Since plant density is affected by plant spacing, which, according to [8], is the arrangement of plants on the planted area. As part of vegetable crop production, plant spacing is crucial for ensuring all plants receive sufficient sunlight, water, and nutrients, facilitating air circulation between plants, and enabling the detection of pests and diseases. In addition, it facilitates access to pest control measures such as handpicking and spraying, as well as intercultural operations such as weeding and pruning. From the point of view of a planting pattern layout scheme, planting in straight rows can be categorized into two categories: (a) vertical rows and (b) alternate rows. A row of vegetables is planted exactly perpendicular to its neighbors in the former pattern (square and rectangular systems) [7, 9]. In accordance, in this approach, potted plants are arranged in a rectangular arrangement as shown in Figure 1.

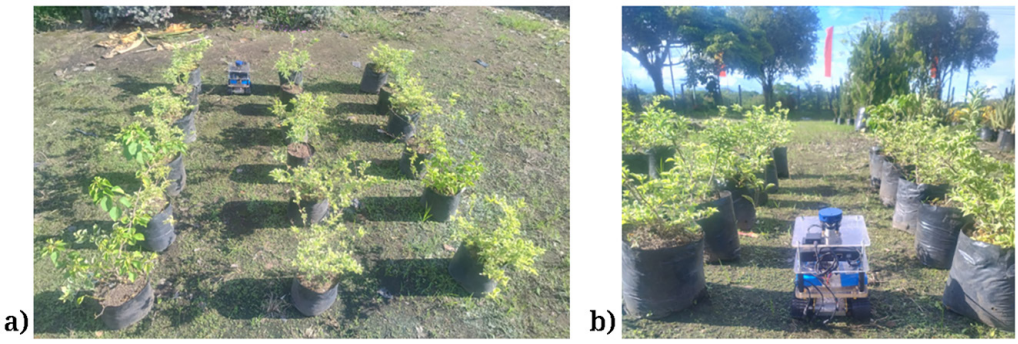

Fig. 1. The field of experimentation

The main objective of this study is to design and implement optimal control methods, specifically linear quadratic regulator (LQR) and linear quadratic Gaussian (LQG), to control the navigation of a differential drive mobile robot capable of autonomously navigating crop rows. These control techniques have been selected due to their ability to minimize errors in trajectory tracking and disturbance rejection [10]. Additionally, the study examines the impact of laser light detection and ranging (LiDAR) sensor noise caused by leaves and stems of plants, on the performance of the robot.

In the robot mission, autonomous robotic reactive navigation plays an important role. According to [11], reactive navigation is a well-known paradigm for controlling

an autonomous mobile robot, which suggests making all control decisions through some light processing of the current or recent sensor data. Among the many advantages of this paradigm are 1) the possibility to apply it to robots with limited and low-priced hardware resources and 2) the fact of being able to safely navigate a robot in completely unknown environments containing unpredictable moving obstacles. In the line of reactive navigation for mobile robots and according to [12], wall-following behavior refers to moving along contours of walls or the edge of obstacles and keeping a constant distance between mobile robots and the wall. Wall following behavior is the basis of reactive navigation between agricultural crop rows. In this context, [13, 14] used this principle to achieve a reactive navigation system based on H∞ control and LiDAR readings on corn crops. In [15], a fuzzy controller was used to maintain a differential drive mobile robot in a path between crop rows using the robotic operational system (ROS). In [16] crop rows detection is documented using the Terrasentia robot.

This study distinguishes itself from prior research that applied optimal control to laboratory-scale robots by offering a practical comparison between LQR and LQG controllers [10], within a real agricultural context, specifically in vegetative furrows, utilizing the ROS framework [17, 18]. This integration of optimal control and agricultural robotics offers novel insights into reactive navigation amidst natural disturbances, such as LiDAR noise induced by leaves and stems.

In Section 2, how the mobile robot has been configured for this study is described. Section 3 introduces the lateral distance measurement procedure for following crop rows. Section 4 describes the implementation in ROS. The closed loop optimal control system is discussed in Section 5, followed by results and conclusions.

2 MOBILE ROBOT DESCRIPTION

A differential-driven mobile robot prototype was constructed. Figure 2 illustrates the robot's dimensions in centimeters. Figure 2a illustrates a side view, Figure 2b depicts a front view, Figure 2c illustrates a top view, and Figure 2d illustrates an underside view. The mobile robot weighs approximately 1.5 kilograms, and it is propelled by two DC motors controlled by its caterpillar system. To carrying the vehicle devices, the mobile robot has been structured into three floors. The devices were distributed, as shown in Figure 3, as follows: 1) under the first floor, there are two DC motors; 2) on the first floor, there are two 12VDC batteries. One battery supplies power to a mini-PC, micro controller Board, inertial measurement unit (IMU), and LIDAR sensor, while the second battery supplies power to two H-bridges that control the two DC motors. The first floor also contains two H-bridges and one IMU BNO055; the second floor contains a minicomputer (mini-PC); the third floor has a LiDAR sensor; and finally, a caterpillar locomotion system is suitably assembled under the robot chassis.

An x86 fan-less computer was deployed over the robot structure, since a key software element of this approach is the ROS [17, 18]. It corresponds to the Hyundai MiniPC, which is equipped with a Celeron N4020 processor running at up to 2.8 GHz, 4 GB of RAM, and 128 GB of solid-state storage. Linux Ubuntu 20.04 was installed on the computer, as well as the noetic ROS distribution.

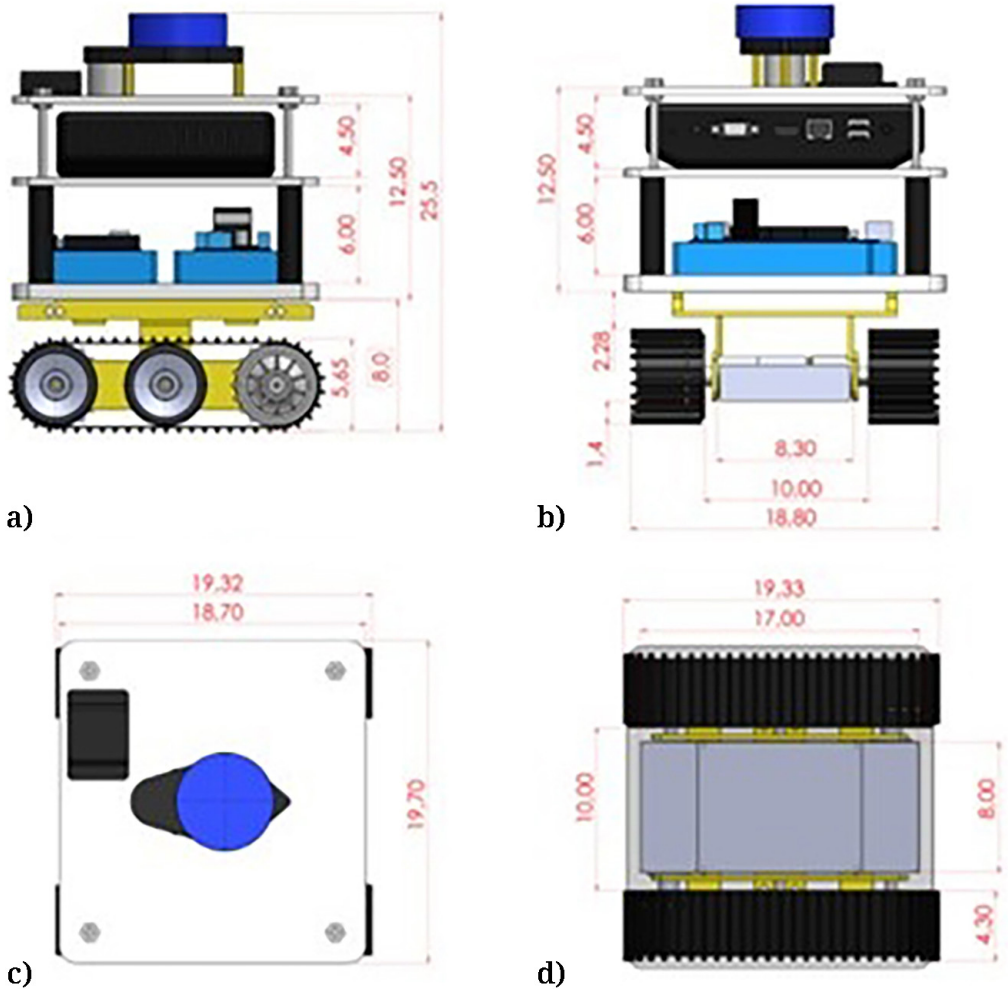

Fig. 2. Mobile robot dimensions

A LiDAR sensor is mounted over the robot structure, whose model name is YDLiDAR X2. The sensor has been developed by Shenzhen EAI Technology Co., Ltd. LiDAR is a remote sensing technology that utilizes laser light beams with defined intensity and focus, with the reflected beam arriving time being measured by photodiodes (PD) within the sensor [15, 19, 20, 21]. In addition to measuring distances, LIDAR sensors are capable of detecting obstacles in their surroundings [19, 20, 21]. A description of how to set up the YDLidar sensor for use with the Linux Ubuntu distribution is included in the YDLidar user manual, as are instructions on how to install ROS drivers. This manual specifies that YDLidar is compatible only with Ubuntu 18.04; however, we tested YDLidar on Linux Ubuntu 20.04 and obtained acceptable results [19].

The BNO055 inertial measurement unit (IMU) is described in [22, 23, 24, 25] as a nine-axis absolute orientation sensor that integrates a three-axis accelerometer, a three-axis gyroscope, and a three-axis magnetometer. This sensor is capable of measuring rigid body pitch, yaw, and raw angles. The mobile robot is equipped with an IMU whose readings can be read via I2C protocol. A microcontroller was used to transfer the data from the IMU to the main computer through USB communication.

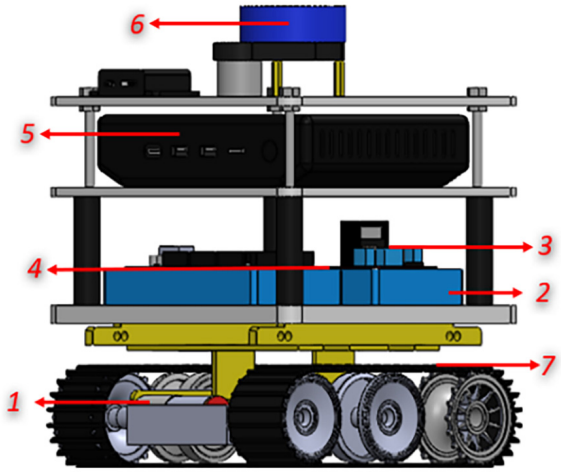

Fig. 3. Arrangement of components on the robot chassis: (1) DC motors; (2) 12VDC batteries; (3) H-bridges; (4) the IMU BNO055; (5) Mini-PC; (6) LiDAR sensor; (7) Caterpillar locomotion system

Robot locomotion is dependent on a caterpillar tracking system controlled by two DC motors whose commercial reference is GM25-370CA. This model is equipped with a gearbox and is manufactured by TT Motor (HK) Industrial Co., Ltd. Each motor is equipped with a quadrature encoder, which allows for the calculation of the rotational speed and the angular position of its axis.

In Figure 4, a block diagram depicts the data communication between the vehicle devices, in accordance the microcontroller reads the IMU sensor via I2C communication protocol, and sends the IMU measurements to the miniPC via USB connection. Also through the USB connection, the microcontroller receives commands from the miniPC to drive the DC motors. The LiDar sensor is equipped with a UART, which enables it to communicate with the miniPC via USB. Therefore, LiDar readings are transmitted to the miniPC, which interprets them by using Python libraries. Using a double L298 H-Bridge connected to the microcontroller, two DC motors can be driven. Angular speed is measured by Hall Effect quadrature encoders connected to the microcontroller on both DC motors.

As the main processor is an x64 computer running Ubuntu 20.04, it has been configured to run Python3 [26, 27] as well as the Noetic distribution of ROS [17, 18].

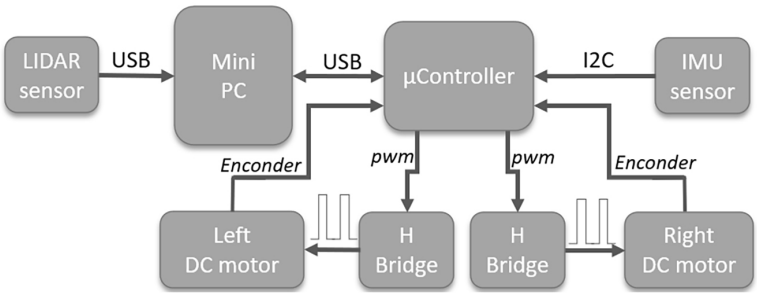

Fig. 4. Devices relations

3 THE LATERAL DISTANCE MEASUREMENT

Since LiDAR measures distances to obstacles all around it, it reports corresponding angles also [19, 20, 21]. Based on experiments and approaches documented in [13, 14, 15], we found that using a fixed LIDAR beam pointing at a specific angle to calculate the

lateral distance to a furrow is not appropriate for determining the distance between plants since the beam will eventually pass through the separation between the plants. The phenomenon also occurs when several beams are used, as shown in Figure 5a, where some beams escape from the furrow. Also, it was noted that the orientation of the vehicle must be taken into account when calculating the distance from the mobile robot to the crop file ($\boldsymbol{l_m}$). For resolving these circumstances, five restrictions were considered. The five restrictions considered in the design of the lateral distance measurement algorithm are as follows: 1) Sensor coverage, 2) Vehicle orientation, 3) LiDAR measurement region (LMR), 4) Filtering of invalid readings, 5) Validation of projected distances.

As a first restriction, it is important to keep in mind that the LIDAR measures distances to obstacles all around it, as well as the angle for each distance measured. Accordingly, the mode in which the angles are reported is depicted in Figure 5b, where for a LiDar side the angles are reported between 0 *rads* and π *rads*, while for the opposite LiDar side, the angles are reported between –0 *rads* and π *rads*.

Regarding the second restriction, the vehicle heading orientation (ψ) which is measured by an IMU (reference) must also be considered. We therefore use ψ to calculate the current orientation error ($e_\psi = \psi_d - \psi$) where ψ_d corresponds to the desired vehicle heading during the robot mission.

A third restriction involves configuring a ***LMR*** that consists of high and low thresholds, as shown in Figure 5b. According to equations (1) and (2), the ***LMR*** will generate an observation window which is defined by an angle ($\boldsymbol{\alpha}$) in relation to $\pi/2$. Consequently, it is appropriate to indicate that the ***LMR*** remains unemployed in situations *a*1, *a*2, and *a*3 of Figure 5c. However, equations (1) and (2) allow the displacement of the ***LMR*** in order to ensure that the measurement window during the robot mission remains orthogonally in relation to the ideal heading angle, as illustrated in Figure 5d, situations *d*1, *d*2 and *d*3, which represent the ***LMR*** correction for situations depicted in Figure 5a, situations *a*1, *a*2 and *a*3 respectively.

$$high_{threshold} = \frac{\pi}{2} + \alpha + e_\psi, \tag{1}$$

$$low_{threshold} = \frac{\pi}{2} - \alpha + e_\psi. \tag{2}$$

As a fourth restriction, due to the fact that measurements reported as zero in the LiDar sensor correspond to distances so close or so far from the LiDar, those measurements are discarded in the correct ***LMR*** and the remainder are projected. Let us illustrate this point by using Figure 5c, which shows a situation in which ***LMR*** has been corrected and some measurements have been extracted just to demonstrate how $\boldsymbol{l_m}$ can be calculated. Consequently, we outlined the process in Figure 5c by taking only one set of LiDar measurements, such that each measurement corresponds to a LIDAR angle. In equations (3) and (4), $\boldsymbol{l_x}$ represents each projection, and each measurement must be projected as described in the equations.

$$l_x = l_n \cos\left(\frac{\pi}{2} + \alpha_n\right) \forall \alpha \geq \frac{\pi}{2}, \tag{3}$$

$$l_x = l_n \cos\left(\frac{\pi}{2} - \alpha_n\right) \forall \alpha < \frac{\pi}{2}. \tag{4}$$

The fifth restriction aims to distinguish valid measurements from invalid measurements. In Figure 5d, the furrows in our experiment field are separated by a distance of $\boldsymbol{w}$. If $\boldsymbol{l_x}$ is greater than $\boldsymbol{w}$, then this projection must be rejected.

The remaining projections will be considered valid projections, which will allow us to compute $\boldsymbol{l_m}$ using equation (5), where $\boldsymbol{n}$ indicates the number of valid projections.

$$l_m = \frac{\sum_1^n l_x}{n}. \tag{5}$$

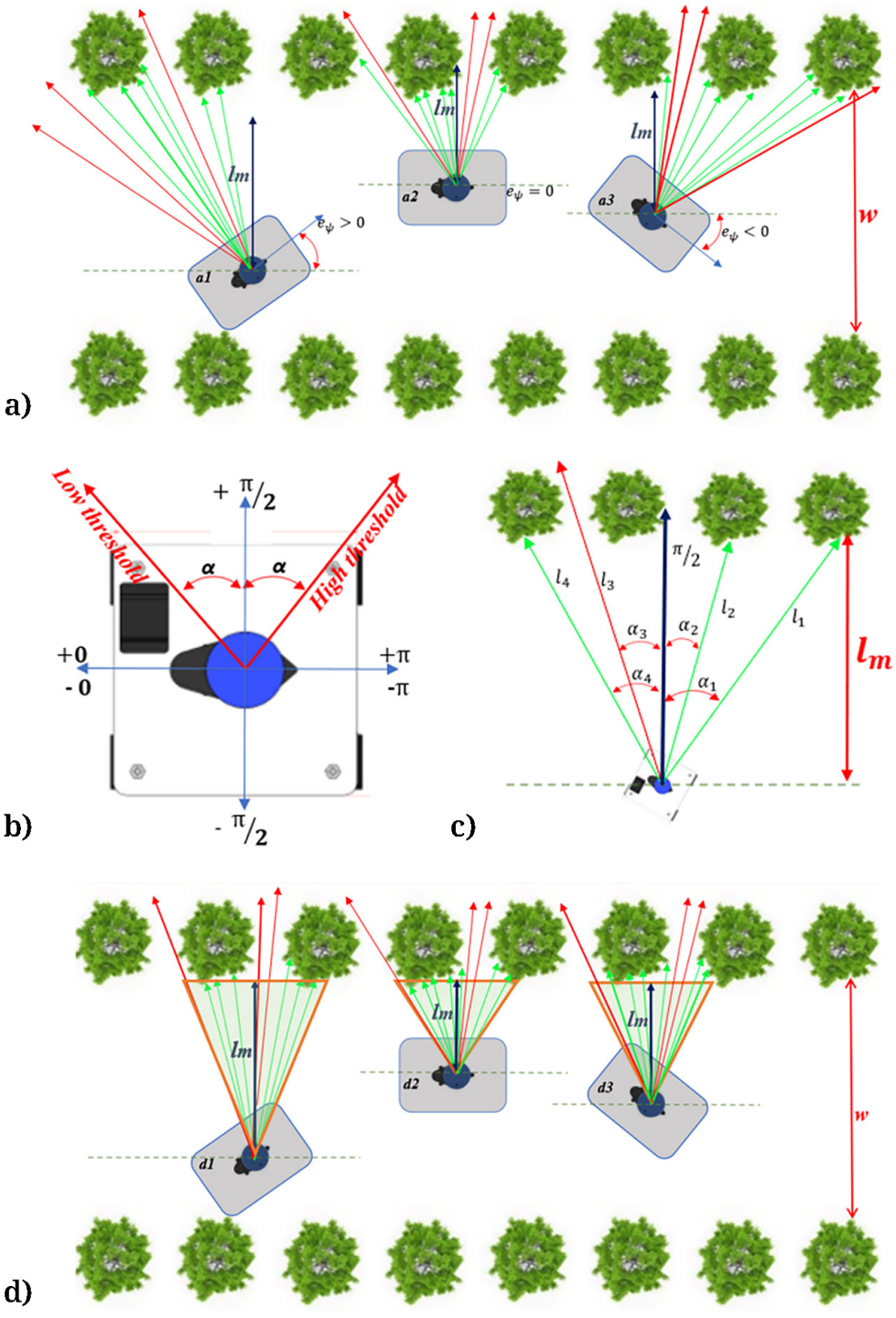

Fig. 5. (a) Error measurement due to the beam passing through the separation between the plants; (b) Angular configuration of the LiDar sensor; (c) Definition of threshold for considered measurements; (d) Displacement of the ***LMR*** to ensure that the measurement window during the robot mission

4 ROS UNDERTAKING

The ROS Noetic distribution was used in this project, with ROS nodes configured in Python3 programming language. A library for utilizing the LiDar sensor from ROS was also required [19], a library for using the IMU sensor [25], as well as the serial library [17]. Referring again to Figure 4, we are able to correlate the information flux with the configured ROS nodes as shown in Figure 6. In accordance with the LiDAR sensor, angle and distance measurements are sent to the miniPC. In contrast, orientation measurements are sent from the IMU to the microcontroller, which in turn passes this data to the miniPC. In order to clarify, it is appropriate to specify that the *controller* and the *LIDAR* nodes reside on the miniPC, while the *μcontroller* node resides on the microcontroller. *LIDAR* node publishes all LIDAR measurements (distances and angles) to the topic *laser_fan*. In order to receive LIDAR measurements, the *controller* node subscribes to the *laser_fan* topic. It has a publisher that publishes the orientation measured by the IMU sensor heading to the topic ψ. The *controller* node has another subscriber who is subscribed to the ψ topic.

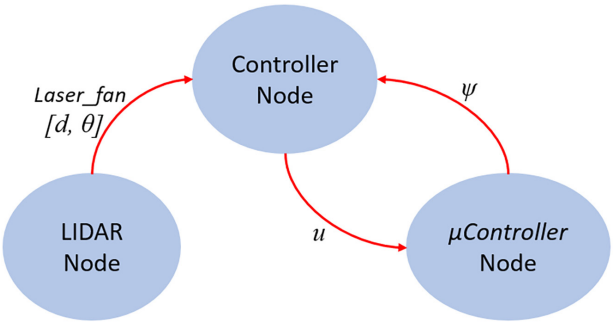

Fig. 6. ROS nodes and topics in the project

Considering that the controller node subscribes to the *laser_fan* and ψ topics, LIDAR measurements and vehicle orientation are processed by this node according to Figure 6 and equations (1) to (5), which allow the robot mission to be accomplished by calculating a control action. In the *controller node*, control actions are delivered through a publisher that publishes information to the topic *u*. The *μController node* subscribes to the topic *u*. Control actions are calculated using a closed loop control system strategy.

5 CLOSED LOOP OPTIMAL CONTROL SYSTEM

A closed-loop and optimal control system deals with the robot's mission, as shown in Figure 7. This figure depicts two levels of control, where the high-level control receives information from the LiDAR and the BNO055 sensors. The high-level control calculates the control action (u) aiming at the accomplishment of the robot's mission. The low-level control stage consists of a PID controller that guarantees wheel velocities. Notice that the velocity reference is the control action given by the kinematic control plus a nominal angular velocity, which is set like $\omega_n = 1.8\ rad/s$. Figure 7 shows the LQG approach, which consists of the feedback given by the Kalman filter estimation and the LQR control vector. In the graph, the robot is a non-linear kinematic model, and motors are the functions G_R and G_L for the right and left wheels, respectively.

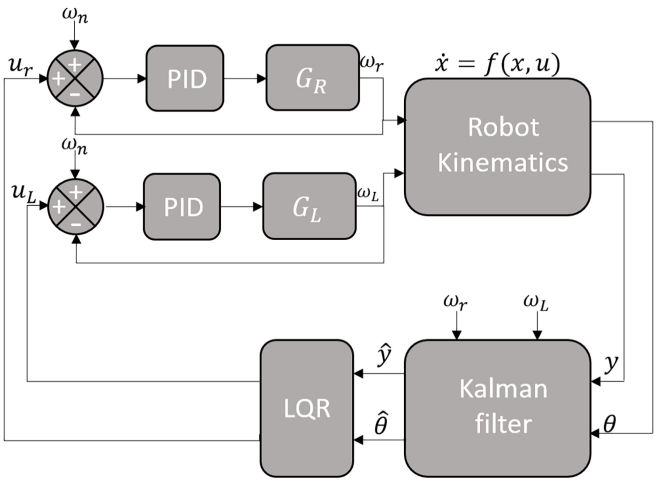

Fig. 7. Overall control system

5.1 Optimal controller for high-level control system

Optimal control systems are based on a mathematical model of the plant to be controlled [28, Ch. 9 page 359, 29]. The adopted mathematical model corresponds to a differential drive, then under kinematic considerations is defined by equations (6) and (7), where $\dot{y}$ is the lateral speed of the robot, ω_L and ω_R are the angular speeds of the left and right robot sides respectively. In accordance to shown in Figure 8, r is the radius of the wheel which that tracks the caterpillar system and finally, b is a distance between the longitudinal axis of the robot chassis and a longitudinal axis of belt of the track system and finally, $\dot{\alpha}$ is the angular speed of the mobile robot.

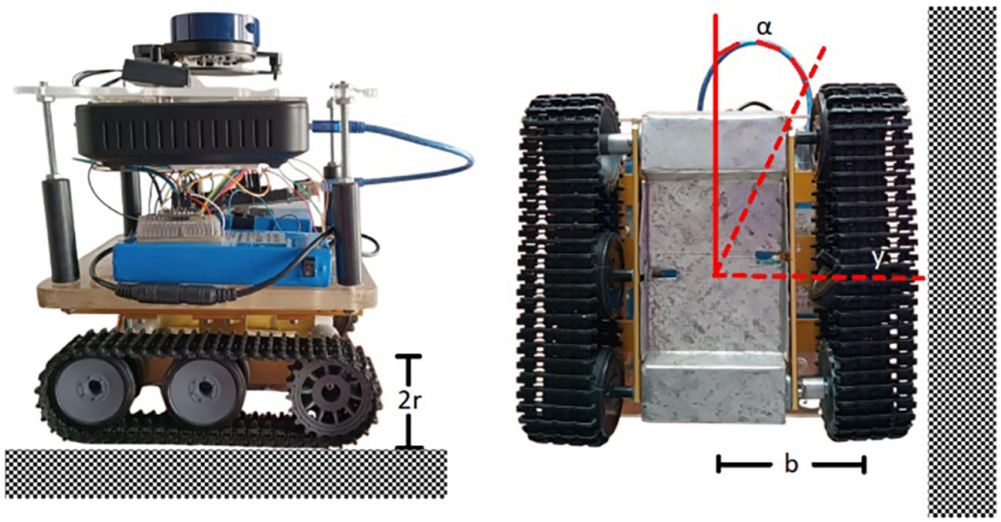

Fig. 8. Robot kinematic model

$$\dot{y} = \frac{(\omega_L + \omega_R)r}{2} \sin\alpha, \tag{6}$$

$$\dot{\alpha} = \frac{(\omega_L - \omega_R)r}{2b}. \tag{7}$$

In accordance with the kinematic model the vehicle angular speed depends on the angular speeds difference, the radius r and the separation b. Kinematic model is linearized around the set point $\alpha = 0$ using the Taylors approach, getting the model expressed in space state as follows,

$$\begin{bmatrix} \dot{y} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} 0 & \frac{(\omega_L + \omega_R)r}{2} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y \\ \alpha \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{r}{2b} & -\frac{r}{2b} \end{bmatrix} \begin{bmatrix} \omega_L \\ \omega_R \end{bmatrix}. \tag{8}$$

5.2 Linear quadratic regulator

Consider the discrete time linear mathematical model,

$$x_{i+1} = Fx_i + Gu_i, \quad i = 0, \ldots, N-1, \tag{9}$$

Where, $x_i \in \mathcal{R}^n$ is the state vector, $u_i \in \mathcal{R}^m$ is the control action and $F \in \mathcal{R}^{nxn}$ and $G \in \mathcal{R}^{nxm}$ are parametrical matrices. The goal of the LQR is to find the sequence u_i which minimizes the cost function indicated in the equation (10) subjects to the model in equation (9).

$$J = x_{N+1}^T P_{N+1} x_{N+1} + \sum_{j=0}^{N} (x_i^T Q_j x_j + u_j^T R_j u_j). \tag{10}$$

The ponderation matrices with suitable dimensions $P \succ 0$, $Q \succ 0$, $R \succ 0$ allows to get the optimal control law $u_i^* = -Kx_i$ and the control vector K through the usage of the following recursive equations,

$$P_i = F^T(P_{i+1} - P_{i+1}G(R_i + G^T P_{i+1} G)^{-1} G^T P_{i+1})F + Q_i, \tag{11}$$

$$K = (R_i + G^T P_{i+1} G)^{-1} G^T P_{i+1} F. \tag{12}$$

In this research, we calculated for validation purposes an LQR control by the following matrices,

$$Q = \begin{bmatrix} 1000000 & 0 \\ 0 & 300000 \end{bmatrix}, \; R = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}, \tag{13}$$

where the control vector is given by:

$$K = \begin{bmatrix} 20.07 & 16.25 \\ -29.07 & -16.25 \end{bmatrix}. \tag{14}$$

5.3 Linear quadratic Gaussian regulator

Consider the following discrete time linear mathematical model,

$$x_{i+1} = Fx_i + Gu_i + \Gamma\gamma_{i+1}, \quad i = 0, \ldots, N, \tag{15}$$

$$z_i = Cx_i + v_i, \tag{16}$$

Where, $x_i \epsilon \mathcal{R}^n$ is the state vector, $u_i \epsilon \mathcal{R}^m$ is the control action, $\gamma_i \epsilon \mathcal{R}^{m_2}$ is the disturbance, $z_i \epsilon R_p$ is the measured vector, $v_i \epsilon R_t$ is the noise vector, F, G, Γ and C are parametric matrices with suitable dimensions. In accordance with separation principle [28, Ch. 8 page 302], the LQG control aims to find the sequence u_i that minimizes an expected cost function using the LQR in the same way as in equations (9) to (14), however, the state variables are estimated using a Kalman filter. This filter aims to find the optimal estimates of the states $\hat{x}_i^*, \hat{x}_{i+1}^*$ minimizing the following quadratic criterion.

$$\min_{\hat{y}_i, \hat{v}_i, \hat{x}_i, \hat{x}_{i+1}} \left\{ \left\| \hat{x}_i - \hat{x}_{i|1i-1} \right\|^2_{P^{-1}_{n,i|i-1}} + \begin{bmatrix} \hat{y}_i \\ \hat{v}_i \end{bmatrix}^T \begin{bmatrix} Q_n & 0 \\ 0 & R_n \end{bmatrix}^{-1} \begin{bmatrix} \hat{y}_i \\ \hat{v}_i \end{bmatrix} \right\}, \tag{17}$$

Subject to the model in equations (15) and (16). In this context, matrices $P_n \succ 0$, $Q_n \succ 0$, $R_n \succ 0$ are the a-posteriori covariance estimation, the noise covariance of the process, and the noise covariance of the observation, respectively. The updating of the matrix $P_{n,i|i} \succ 0$ is done by the Ricatti equation as follows,

$$P_{n,i|i} = P_{n,i|i-1} - P_{n,i|i-1} C^T (R_n + C P_{n,i|i-1} C^T)^{-1} C P_{n,i|i-1}, \tag{18}$$

$$P_{n,i+1|i} = Q_n + F P_{n,i|i} F^T, \tag{19}$$

and the Kalman gain L_i is given by,

$$L_i = F P_{n,i|i-1} C^T (R_n + C P_{n,i|i-1} C^T)^{-1}. \tag{20}$$

The LQG control law is given by the following equations,

$$u_i = -K \hat{x}_{i|i-1}, \tag{21}$$

$$\hat{x}_{i+1} = F \hat{x}_{i|i-1} + G u_i + L_i \left(z_i - C \hat{x}_{i|i-1} \right). \tag{22}$$

In this study, we calculated for validation purposes an LQG control by the following matrices,

$$Q_n = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}, \; R_n = \begin{bmatrix} 1 * 10^{-6} & 0 \\ 0 & 0.305 \end{bmatrix}, \tag{23}$$

where, the Kalman and control vectors are the next,

$$L = \begin{bmatrix} 1.22 & 0.008 \\ 44.42 & 0.631 \end{bmatrix}, \; K = \begin{bmatrix} 29.07 & 16.25 \\ -29.07 & -16.25 \end{bmatrix}. \tag{24}$$

In this study, the Kalman filter is instrumental in estimating the robot's orientation by integrating inertial data from the BNO055 IMU with feedback from the LQG control loop. Its application significantly mitigates the effects of sensor noise and short-term drift, thereby providing a stable, reliable estimate of the robot's yaw angle during navigation between vegetative furrows. The strong compatibility between the Kalman filter and IMU sensors is well documented for improving

accuracy in dynamic, noisy environments, such as agricultural fields. Recent studies have demonstrated analogous benefits in real-time angle estimation using adaptive Kalman filters with forgetting factors [30], in robust localization based on GNSS–IMU data fusion via smoothed error-state Kalman filters [31], and in wearable motion tracking systems employing low-cost IMU sensors [32]. These findings support the suitability of the proposed LQG approach for improving orientation estimation and overall navigation performance in field robotics.

6 RESULTS

The experimental tests were conducted in an open field environment using potted Duranta plants arranged to simulate vegetative crop rows. Each plant was placed in an individual grow bag. The inter-row spacing was approximately 60 cm, while the intra-row spacing between plants was 30 cm. The robot was programmed to navigate through the rows while maintaining a lateral reference distance of 30 cm from the right-side row of plants, which served as the control target. This setup enabled realistic testing of the system's lateral tracking, LiDAR-based sensing, and robustness under irregular vegetation.

We perform two experiments to validate tracking control for vegetative furrows. The first experiment considers the LQR and the second the LQG controller. In the second test, the Kalman filter is incorporated to reject the disturbance and noise mainly due to errors in detecting leaves and seeds. Each test lasted 150 seconds, during which the robot navigated along the path and avoided colliding with the plant barrier. Figure 9 shows the experimental results of mobile robot tracking control for vegetative furrows. The experiments consider control tracking by an LQR in Figure 9a and an LQG controller in Figure 9b.

At the top of the graph is shown the regulation error for the vegetative furrows, in the middle top the angular position of the robot, and at the bottom, the control actions for the right and left wheels, respectively. The aim is to follow the line path between vegetative furrows. Notice that leaves and stems of plants produce noisy signals on the distance sensor, as shown at the top of the Figure 9. However, both control strategies achieve the regulation while moving forward on the path. Therefore, we employed the root mean square error ($RMSE_{\Delta}$) and the mean value $\bar{\Delta}$ to evaluate the performance obtained by the controllers, as follows,

$$RMSE_{\Delta} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\Delta_i^2}, \tag{25}$$

$$\bar{\Delta} = \frac{1}{n}\sum_{i=1}^{n}\Delta_i. \tag{26}$$

Where, n is the total number of samples and Δ represents the y and α variables for distance and angular position, respectively. The RMSE indicates the standard deviation of the measurements and the mean value of the robot's ability to regulate both the lateral distance and the angular position. Results are summarized in Table 1 and indicate that $RMSE_y$ is 0.13 m for the LQR test and 0.08 m for the LQG test. Also, $RMSE_{\alpha}$ is 0.19 *rad* for the LQR test and 0.13 *rad* for the LQG test. Therefore, there are fewer measurement variations with the LQG approach and, consequently, less noise due to foliage vegetation. The mean value of distance error $\bar{y}$ is –0.06 m for the LQR

test and –0.01 m for the LQG test. In the same way, the mean value of the angular position $\bar{\alpha}$ is 0.09 *rad* for the LQR test and 0.02 *rad* for the LQG test. Notice that in all tests, the LQG controller outperforms the *linear quadratic regulator*.

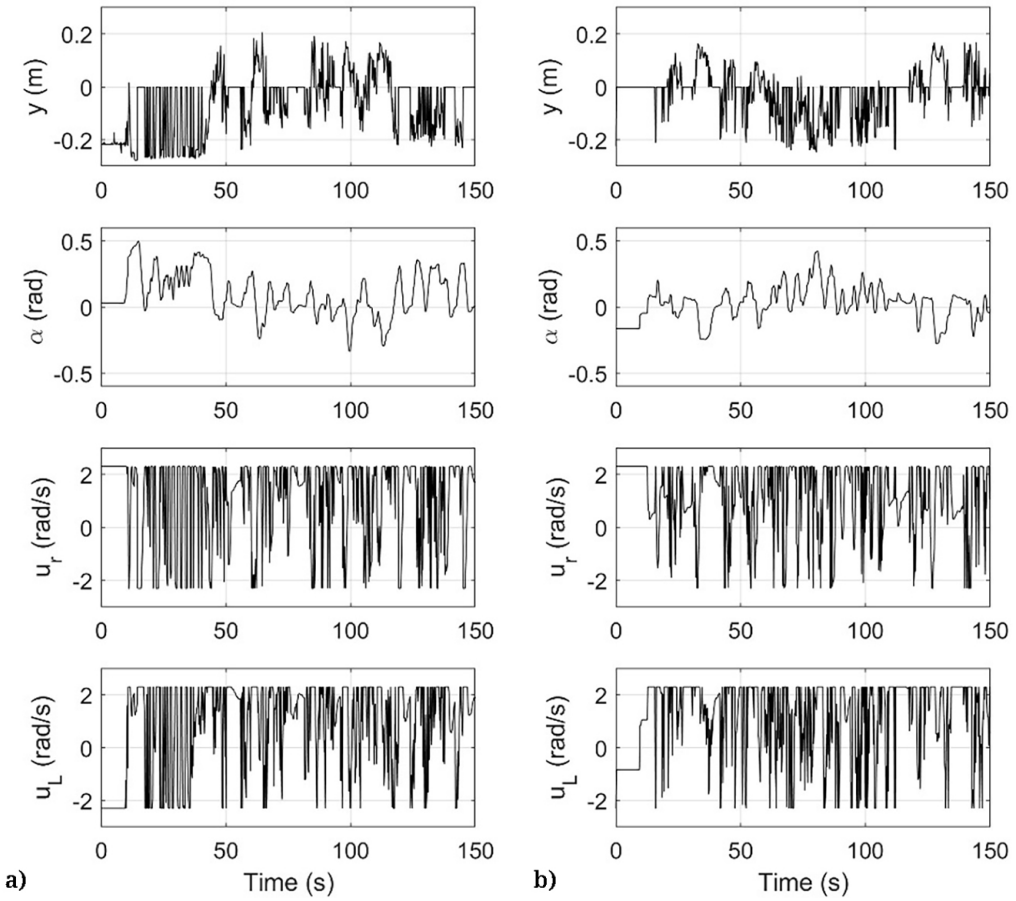

Fig. 9. Tracking control for vegetative furrows: (a) LQR test; (b) LQG test

Table 1. Comparison of experimental results using RMSE and mean values

	$RMSE_y$ (m)	$\bar{y}$ (m)	$RMSE_\alpha$ (*rad*)	$\bar{\alpha}$ (*rad*)
LQR	0.13	–0.06	0.19	0.09
LQG	0.08	–0.01	0.13	0.02

Figure 10 shows the signal dispersion of lateral position error and angular position error for the LQR and LQG tests using a box plot. Results show that the interquartile range is smaller when the robot uses the LQG controller, both for distance regulation and angular behavior. Also, the notches in each box indicate sample skewness, since the median is not centered in the box, which occurs in all tests but is most evident in distance regulation. We attribute the previous result to saturation in the control signals, a constraint imposed by the type of motors and gears used in the robot's construction. At the bottom of Figure 10, the control actions from the selected controllers are shown, highlighting a limit on the nominal velocities of 2.3 rad/s.

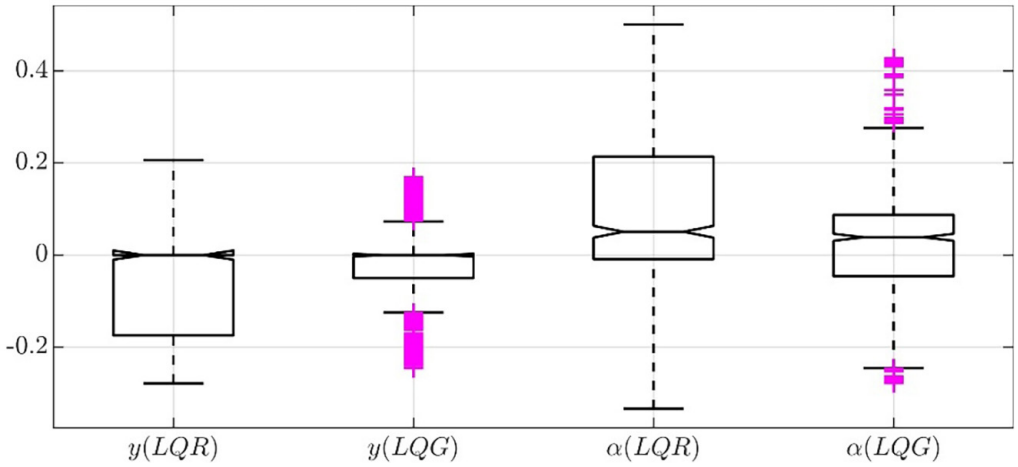

Fig. 10. Signal dispersion of lateral position error and orientation error for LQR and LQG test

In summary, the evidence in Figures 9 and 10, as well as the results in Table 1, shows that tracking the path within vegetative furrows is achieved more effectively with the LQG controller, since the robot's angular position exhibits fewer variations than with the LQR test.

7 CONCLUSIONS

In this study, we implemented an autonomous mobile robot prototype for vegetative furrows based on optimal control over ROS. We validate two kinematic control strategies based on a LiDAR that measures distance and an IMU that measures orientation. The robot control actions are driven by a low-level control based on PID that guarantees the suitable nominal velocities of wheels. We consider an algorithm with five restrictions for the lateral distance measurement based on IMU and LiDAR measurements to regulate distance and the robot's angular position. Experimental results show that the robot with the LQG controller achieves better performance than the LQR test. Although both control strategies showed distance and angle regulation, we plan to use other control techniques in future works to surpass the current results. In the future, we will consider approaches based on Markovian control and non-linear control [33, Ch. 5 page 164], as well as video camera and convolutional neural networks for perception. In consideration of our ongoing research, the authors affirm their willingness to provide information regarding the study upon reasonable request.

8 ETHICAL COMPLIANCE STATEMENT

This study did not involve human or animal subjects and adheres to both institutional and international ethical standards for engineering research.

9 ACKNOWLEDGMENT

The authors gratefully acknowledge the support provided by the Ministry of Science, Technology, and Innovation of Colombia (MINCIENCIAS) and the

Francisco José de Caldas Fund through the allocation of resources under Call 934 of 2023, contract 332-2023, project 99209. The authors also thank the LASER research group at Universidad Distrital Francisco José de Caldas for their guidance and advice during the development of this research project. Special thanks are extended to BitPointer S.A.S., the beneficiary of project 99209, particularly to its manager, Eng. Juan Fajardo Barrero, and its administrative director, lawyer Alexandra Reyes Agudelo, for their valuable contributions and commitment to this initiative.

10 REFERENCES

[1] A. J. Moshayedi, A. Khan, Y. Yang, J. Hu, and A. Kolahdooz, "Robots in agriculture: Revolutionizing farming practices," *EAI Endorsed Transactions on AI and Robotics*, vol. 3, 2024. https://doi.org/10.4108/airo.5855

[2] M. Eissa, "Precision agriculture using artificial intelligence and robotics," *Journal of Research in Agriculture and Food Sciences*, vol. 1, no. 2, pp. 35–52, 2024. https://doi.org/10.5455/JRAFS.20240404014009

[3] M. Naim, D. Rizzo, L. Sauvée, and M. Medici, "Advancing agroecology and sustainability with agricultural robots at field level: A scoping review," *Comput. Electron. Agric.*, vol. 237, p. 110650, 2025. https://doi.org/10.1016/j.compag.2025.110650

[4] A. Ariss, A. Lamjid, S. Ziti, J. Mabrouki, I. Ennejjai, and N. Kharmoum, "IoT-enabled smart greenhouse for robotic enhancement of tomato production: Leveraging 5G and edge computing for advanced data-driven automation, precision irrigation, and scalable zoning principles," *Int. J. Interact. Mob. Technol. (iJIM)*, vol. 18, no. 21, pp. 88–116, 2024. https://doi.org/10.3991/ijim.v18i21.50829

[5] N. Setiyawati, H. Dwi Purnomo, and E. Mailoa, "User experience design on visualization of mobile-based land monitoring system using a user-centered design approach," *Int. J. Interact. Mob. Technol. (iJIM)*, vol. 16, no. 3, pp. 47–65, 2022. https://doi.org/10.3991/ijim.v16i03.28499

[6] Z. Othman, S. Jahroh, N. Abu, E. Alias, N. Zaman, and S. Abdullah, "Mobile marketing training towards increasing business: A case study of agri-food micro-enterprises," *Int. J. Interact. Mob. Technol. (iJIM)*, vol. 18, no. 14, pp. 59–71, 2024. https://doi.org/10.3991/ijim.v18i14.49055

[7] M. Haque and S. Zaharah, "Planting arrangement and effects of planting density on tropical fruit crops—A review," *Horticulturae*, vol. 8, no. 6, p. 485, 2022. https://doi.org/10.3390/horticulturae8060485

[8] R. Pandey, S. Behera, P. Khairiya, and V. Tewari, "Planting techniques: Sowing, spacing and soil considerations," in *Vegetable Chronicles: Vegetable Science Compendium*, 2023, pp. 48–71.

[9] M. H. Ahmed, "Does row planting enhance farm productivity and reduce risk exposure? Insights from Ethiopia," *J. Agric. Appl. Econ.*, vol. 55, no. 1, pp. 133–150, 2023. https://doi.org/10.1017/aae.2023.12

[10] F. J. Rodriguez-Rojas, E. A. Villanueva-Serna, C. A. Perdomo-Charry, and A. L. Jutinico, "Lateral position optimal control for micro mouse robot based on PSoC 5LP," *IFAC-PapersOnLine*, vol. 54, no. 4, pp. 106–111, 2021. https://doi.org/10.1016/j.ifacol.2021.10.018

[11] J. A. Tobaruela and A. O. Rodríguez, "Reactive navigation in extremely dense and highly intricate environments," *PLoS ONE*, vol. 12, no. 12, p. e0189008, 2017. https://doi.org/10.1371/journal.pone.0189008

[12] X. Wei, E. Dong, C. Liu, G. Han, and J. Yang, "A wall-following algorithm based on dynamic virtual walls for mobile robots navigation," in *2017 IEEE Int. Conf. Real-time Comput. Robot. (RCAR)*, Okinawa, Japan, 2017, pp. 46–51. https://doi.org/10.1109/RCAR.2017.8311834

[13] H. B. Guerrero, "Desenvolvimento de um sistema de controle em um robô móvel agrícola em escala reduzida para deslocamento entre fileiras de plantio," Ph.D. dissertation, Escola de Engenharia de São Carlos, Univ. São Paulo, São Carlos, Brazil, 2016. https://doi.org/10.11606/T.18.2017.tde-24122016-154205

[14] A. E. B. Velasquez *et al.*, "Reactive navigation system based on H∞ control system and LiDAR readings on corn crops," *Precis. Agric.*, vol. 21, no. 2, pp. 349–368, 2020. https://doi.org/10.1007/s11119-019-09672-8

[15] H. Borrero-Guerrero, J. J. López-Aguilar, A. Orduz-García, S. V. Barrero, and A. L. Jutinico-Alarcón, "A differential drive mobile robot controlled by using the robotics operational system (ROS)," *Visión Electrónica*, vol. 17, no. 2, pp. 208–219, 2023.

[16] A. E. B. Velasquez, V. A. H. Higuti, M. V. Gasparino, A. N. V. Sivakumar, M. Becker, and G. Chowdhary, "Multi-sensor fusion based robust row following for compact agricultural robots," *Field Robotics*, vol. 2, pp. 1291–1319, 2022. https://doi.org/10.55417/fr.2022043

[17] L. Joseph and A. Johny, *Robot Operating System (ROS) for Absolute Beginners: Robotics Programming Made Easy*. Berkeley, CA: Apress, 2022, pp. 125–171. https://doi.org/10.1007/978-1-4842-7750-8

[18] D. A. Blubaugh, S. D. Harbour, B. Sears, and M. J. Findler, "Noetic ROS further examined and explained," in *Intelligent Autonomous Drones with Cognitive Deep Learning*. Berkeley, CA: Apress, 2022, pp. 417–436. https://doi.org/10.1007/978-1-4842-6803-2_12

[19] YDLidar, "X2 user manual," Shenzhen EAI Technology Co., Ltd., 2017. [Online]. Available: https://static.generation-robots.com/media/manuel-utilisation-ylidar-x2.pdf [Accessed: Jul. 10, 2025].

[20] I. A. Bhuiyan, "LiDAR sensor for autonomous vehicle," Technical Report, 2017. https://doi.org/10.13140/RG.2.2.16982.34887/1

[21] N. Umezu, S. Koizumi, K. Nakagawa, and S. Nishida, "Potential of low-cost light detection and ranging (LiDAR) sensors: Case studies for enhancing visitor experience at a science museum," *Electronics*, vol. 12, no. 15, p. 3351, 2023. https://doi.org/10.3390/electronics12153351

[22] Z. Lin, Y. Xiong, H. Dai, and X. Xia, "An experimental performance evaluation of the orientation accuracy of four nine-axis MEMS motion sensors," in *2017 5th Int. Conf. Enterprise Systems (ES)*, 2017, pp. 185–189. https://doi.org/10.1109/ES.2017.37

[23] C. McIver and T. Gahl, "Positioning and testing of inertial measurement units," in *Academic High Altitude Conf.*, 2017, vol. 1. https://doi.org/10.31274/ahac.5559

[24] P. Thavitchasri, D. Maneetham, and P. N. Crisnapati, "Intelligent surface recognition for autonomous tractors using ensemble learning with BNO055 IMU sensor data," *Agriculture*, vol. 14, no. 9, p. 1557, 2024. https://doi.org/10.3390/agriculture14091557

[25] K. Townsend, "Adafruit BNO055 absolute orientation sensor," Adafruit.com, n.d. [Online]. Available: https://cdn-learn.adafruit.com/downloads/pdf/adafruit-bno055-absolute-orientation-sensor.pdf [Accessed: Jul. 10, 2025].

[26] F. Bittmann, *Python 3 for Science and Engineering Applications*. London: Elektor International Media BV, 2020.

[27] F. Romano and H. Kruger, *Learn Python Programming: An In-depth Introduction to the Fundamentals of Python*. Birmingham: Packt Publishing Ltd., 2021.

[28] G. F. Franklin, J. D. Powell, and M. L. Workman, *Digital Control of Dynamic Systems*, 3rd ed. Menlo Park, CA: Addison-Wesley, 1998.

[29] F. M. Escalante, A. L. Jutinico, M. H. Terra, and A. A. Siqueira, "Robust linear quadratic regulator applied to an inverted pendulum," *Asian J. Control*, vol. 25, no. 4, pp. 2564–2576, 2023. https://doi.org/10.1002/asjc.2978

[30] Z. Anvari, Y. Salehi, and A. Mirhaghgoo, "Real-time angle estimation in IMU sensors: An adaptive Kalman filter approach with forgetting factor," *Mechatronics*, vol. 106, p. 103280, 2025. https://doi.org/10.1016/j.mechatronics.2024.103280

[31] Y. Yin, J. Zhang, M. Guo, X. Ning, Y. Wang, and J. Lu, "Sensor fusion of GNSS and IMU data for robust localization via smoothed error state Kalman filter," *Sensors*, vol. 23, no. 7, p. 3676, 2023. https://doi.org/10.3390/s23073676

[32] C. Keskinoğlu and A. Aydın, "Wearable wireless low-cost electrogoniometer design with Kalman filter for joint range of motion measurement and 3D modeling of joint movements.," *Proc. Inst. Mech. Eng., Part H: Journal of Engineering in Medicine*, vol. 235, no. 2, pp. 222–231, 2020. https://doi.org/10.1177/0954411920971398

[33] S. G. Tzafestas, *Introduction to Mobile Robot Control*. Amsterdam, The Netherlands: Elsevier, 2013.

11 AUTHORS

Henry B. Guerrero obtained a degree in electronic engineering from Universidad de los Llanos, Villavicencio, Colombia, in 2004. He subsequently earned a specialist degree in automatic and industrial informatics from Universidad Autónoma de Colombia in 2007, and a Ph.D. in mechanical engineering from the University of São Paulo, Brazil, in 2016. He is currently a postdoctoral researcher with BITPOINTER SAS Company, in collaboration with the Ministry of Science, Technology, and Innovation of Colombia (MINCIENCIAS) and the Francisco José de Caldas Fund, through the allocation of resources under Call 934 of 2023, contract 332-2023, project 99209 (E-mail: hbguerrero@ieee.org).

Juan J. López A. is an electronic engineering student at Universidad Distrital Francisco José de Caldas, Bogotá, Colombia, and a student member of the LASER research group at the same university (E-mail: jujlopeza@udistrital.edu.co).

Andrés L. Jutinico received a degree in electronic engineering from Universidad Distrital Francisco José de Caldas, Bogotá, Colombia, in 2005; the M.Sc. degree in industrial automation from Universidad Nacional de Colombia in 2012; and the Ph.D. degree in mechanical engineering from the University of São Paulo, Brazil, in 2019. Director of the LASER research group of the Universidad Distrital Francisco José de Caldas (E-mail: aljutinicoa@udistrital.edu.co).