Received 21 May 2023, accepted 30 May 2023, date of publication 8 June 2023, date of current version 14 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3283986

APPLIED RESEARCH

Smart Farming Robot for Detecting Environmental Conditions in a Greenhouse

PAUL D. ROSERO-MONTALVO[1], CARLOS A. GORDILLO-GORDILLO[2], AND WILMAR HERNANDEZ[3], (Senior Member, IEEE)
[1]Computer Science Department, IT University of Copenhagen, 1172 Copenhagen, Denmark
[2]Computer Science Department, University of Salamanca, 37008 Salamanca, Spain
[3]Facultad de Ingenieria y Ciencias Aplicadas, Universidad de Las Americas, Quito 170513, Ecuador

Corresponding author: Wilmar Hernandez (wilmar.hernandez@udla.edu.ec)

This work was supported in part by the Novo Nordisk Fonden under Award NNF20OC0064411; in part by Corporacion Ecuatoriana para el Desarrollo de la Investigacion y la Academia (CEDIA), Ecuador, under Project: CEPRA XII-2018-13; and in part by Universidad de Las Americas (UDLA), Quito, Ecuador, under Project: IEA.WHP.21.02.

ABSTRACT Agricultural production is on most countries' national agenda because climate change affects crops, fruits, vegetables, and insect infestation. Therefore, achieving maximum production results is a challenge faced by professional growers, who have seen greenhouses as a very good option to guarantee these results. By using new technologies inside greenhouses, farmers can reduce the damaging effect of insects on plants and improve indoor cultivation through climate control. However, to efficiently manage agricultural fields and greenhouses today, farmers have to apply technologies in line with Industry 4.0, such as: robots, Internet of Things devices, machine learning applications, and so on. In this context, deploying sensors plays a key role in collecting data and finding information supporting the farmer's decision-making. As a feasible solution for small farms, this paper presents an autonomous robot that moves through greenhouse crop paths with previously-planned routes and can collect environmental data provided by a wireless sensor network, where the farmer does not have previous information about the crop. Here, an unsupervised learning algorithm is implemented to cluster the optimal, standard, and deficient sectors of a greenhouse to determine inappropriate growth patterns in crops. Finally, a user interface is designed to help farmers plan both the route and distance to be traveled by the robot while collecting information from the sensors to observe crop conditions.

INDEX TERMS Smart farming robot, environmental monitoring, applications of intelligent sensors, integrated sensors, the Internet of Things.

I. INTRODUCTION

Agricultural production plays a fundamental role in the welfare of society and economic exchange between countries [1]. Therefore, food demand is continuously growing; by 2050, farms will need to produce 70% more than the current years [2]. Nevertheless, some production estimates tell countries that the agriculture sector can produce only 3% more because farmers have postponed seeding several crops on their lands because of climatic changes (e.g., droughts and/or floods) [3]. Furthermore, it also has diminished their profits. Therefore, to counteract undesirable environmental conditions and crops damages because of infestations of insects, in some cases, farmers use fertilizers and pesticides to speed up the sowing and harvesting of crops. Unfortunately, in some cases they use substances and materials without the knowledge to manage them on a large scale [4]. As a result of the incorrect use of fertilizers and pesticides, the quality of the land decreases as well as the crops cultivated on it (e.g., fruits and vegetables, among others) lose essential nutritional properties [5].

In this context, greenhouses are considered an appropriate choice to maintain environmental conditions within a desirable range. This prevents crops from being exposed to

The associate editor coordinating the review of this manuscript and approving it for publication was Santosh Kumar.

uncontrolled external factors and yields much better harvesting results [6]. However, inhomogeneous and uncontrolled conditions inside a greenhouse can cause it to fail. Furthermore, ensuring the uniformity of the climate is an important issue since greenhouses have areas drier or more humid than others, and irregular soils. What is worse, differences in the type of irrigation and geographical location, added to the problems mentioned above, can make the greenhouse unprofitable [7]. Moreover, if the farmer lacks sufficient experience, his work, while carrying out the crop management process, could be liable to err [8]. For this reason, accurate and efficient monitoring and automation are keys to turning a failed greenhouse into a profitable one.

Having said this, to avoid the above-mentioned concerns, advanced technologies have been introduced in the agriculture sector to make it more efficient and manageable [9]. These technologies have brought novel solutions that both reduce and optimize the use of fertilizers and pesticides [10]. Among these technologies, the Internet of Things (IoT) stands out. In short, IoT is well worth using because it allows the connection among sensors, machines, and humans [11]. Besides, IoT constitutes the foundations of smart farming, merging emerging technologies with machine learning (ML) applications [12]. Nevertheless, smart farming requires several on-site or remote sensors to collect data and find patterns that humans cannot recognize by themselves.

Wireless sensor networks (WSN) traditionally deploy sensors in situ to gather crop data. However, this rigid electronic design requires checking the batteries constantly, and WSN nodes can be damaged when farmers work near them [4]. The above justifies the need to apply flexible remote sensing techniques using autonomous robots, because unmanned aerial vehicles (UAVs) have to face many problems to fly inside indoor environments [9]. In fact, autonomous robots can move through crop lines collecting data while traveling along planned routes [13]. In addition, they can acquire data in specific hours, return to their initial point to recharge batteries, and check the current performance of sensors [14]. However, they have to deal with extreme scenarios when moving on certain wet or uneven grounds [15]. This harsh environment could inject noise and drift to sensors when they are collecting data. Besides, due to erroneous readings of sensors, sensors wear, and unpredictable events, this could be a non-easy task to detect/remove them. This task is traditionally carried out in powerful servers allocated far away from the greenhouse [11]. As a result, this fact brings about an open communication challenge, because greenhouses are usually placed outside cities and do not have high-bandwidth wireless-communication channels to send the information. Additionally, this procedure of storing information remotely consumes energy in WSN nodes [16].

Furthermore, working with robots requires a robust central node to coordinate, plan routes, and reprogram tasks. Moreover, in greenhouses, robots face maintenance requirements due to the fact that they use microcontrollers, sensors, and actuators which might be worn or damaged in outdoor activities. Therefore, farmers need to deal with managing these existing technologies rather than manual tools [17]. Lastly, the robot itself cannot collect data, because it needs sensors. This brings about another concern to synchronize/prioritize tasks into the microcontroller [18].

This research is focused on developing an ML application that works with an autonomous robot, which collects data from crops in a small-medium-scale greenhouse, on showing farmers relevant information on changes in environmental conditions to support decision-making. This is done to detect specific areas inside a greenhouse where crops do not grow in optimal conditions. The above-mentioned autonomous robot has two well-defined subsystems and planned routes to reach an unsupervised ML application in a decentralized network where the greenhouse lacks a stable internet connection. The first subsystem collects data with information on the greenhouse's environmental conditions and sends them to a central node by LoRaWan protocol. The second subsystem is in charge of the design and control of the robot to guarantee its correct performance while it moves through planned crop routes. To achieve this, a proportional-integral-derivative (PID) controller controls the DC motors that regulate the robot's movement. The PID controller is designed so that the robot can move over uneven terrain and in harsh environments, traveling at low speed along crop lines and without losing the grip provided by the chassis and tracks.

All the information is stored in a central node in charge of training unsupervised algorithms since farmers do not have previous knowledge of the crops. The results are presented in a graphical user interface (GUI) without exchanging information with the cloud or big servers. The farmer can plan the robot route and traveling distance on this GUI, before taking each data acquisition sample and observing crop conditions. In addition, to compare the farmer's decision-making, the robot has a trained model to detect inadequate environmental conditions in real-time monitoring. Consequently, this local decision allows double-checking with the farmer's experience to make future decisions.

As the main results of this research, the `k-means` algorithm is defined as the unsupervised machine learning algorithm to build three clusters. These clusters represent the optimal, standard, and deficient growth patterns. Finally, the main contributions of this paper are as follows:

- Here, it is presented an intelligent autonomous robot that is capable of planning routes by using lightweight PID control in uneven paths, without interfering with usual cultivation activities in greenhouses.
- An entire ML pipeline is defined for facing situations in which the farmer does not have previous data about the crop, and needs to improve the harvesting process with limited bandwidth to send information to the cloud. For this reason, the central node runs the ML application, representing a local computation.
- A GUI is developed in the central node to be an interface between the farmer and the robot, to work together to improve decision-making in the harvesting process.

The rest of the paper is organized as follows. Related research is presented in Section II. Section III shows the design of the electronic system. Section IV presents the planning routes design and its visualization. Section V presents the ML pipeline. Section VI shows the results of this paper. Section VII presents the discussion of this work and the conclusions are provided in Section VIII.

II. RELATED RESEARCH

This section shows the related work to, first, autonomous robots in greenhouses and, second, the new trend to implement ML models with robots to make decisions locally. The following papers presented electronic systems applied to smart farming. However, there are still open problems, such as implementing early stages of ML applications when farmers have not yet applied smart technologies to improve crop production in their greenhouses. Moreover, presenting ML applications running on the device is still an open space for innovations.

A. AUTONOMOUS ROBOTS IN GREENHOUSES

Smart farming by IoT devices has become an important research area. Therefore, several researchers have presented novel contributions. For example, Han et al. [13] presented a data collection system to measure soil moisture to train a prediction algorithm that activates the water repository and keeps the crop under adequate conditions. In addition, Singh et al. [8] showed the tradeoffs of energy use and efficiency mechanisms in monitoring systems using wireless sensor networks in greenhouses. Mohamed et al. [19] defined a real-time self-tuning motion controller for mobile robot systems. Jiang and Moallem [20] presented an intelligent control system for mixing color ratios using LED lights in a greenhouse. Additionally, in [20] proposed testbed provides an easy-to-use plant growth system with IoT-enabled control and monitoring features.

Furthermore, using autonomous vehicles, Durmus and Gunes [7] presented a mobile robot to gather data from agricultural fields or greenhouses and then send this data to a web application. Rosero-Montalvo et al. [14] showed a design of a quadruped robot for monitoring rose crops using supervised learning algorithms. In addition, [21], [22], [23], [24] presented advances in harvesting by data acquisition schemes in smart farming. Moreover, [25], [26], [27] were focused on collision avoidance in the movements of the robots within the greenhouses.

B. ML IN AUTONOMOUS ROBOTS

Fernando et al. [28] presented a greenhouse farming support system with robotic monitoring that measures temperature, soil moisture, humidity, and pH through a cloud-connected mobile robot which can detect unhealthy plants by using image processing and machine learning (ML) techniques. Therefore, this work needs a good communication channel to establish the connection between the greenhouse and the cloud. Then, [29] and [30] presented systems to collect data in greenhouses, and they used ML models to improve their robots' movements. Next, using new and small cameras, Ge et al. [31], presented a fruit localization and environment perception for strawberry harvesting robots by deep learning techniques.

Nowadays, there are works focused on detecting disease in specific crops. In this scenario, Fernández et al. [32] presented an automatic detection system of field-ground cucumbers for robotic harvesting with images. Finally, working with external datasets, works such as [33], [34], and [35] have presented ML models with high accuracy and well-defined deep learning architectures. Despite the fact that the above-mentioned systems have at their core architectures based on machine learning techniques through the use of artificial intelligence, we believe that more can still be done in terms of providing help to farmers with little or no experience in managing of IoT-based systems.

III. ELECTRONIC SYSTEM DESIGN

The electronic system presented in this research has been divided into two subsystems. The first subsystem is the autonomous robot that is designed to move within the greenhouse by distance control marks. The greenhouse is digitized to plan routes and decide the distance for each data gathering point, which we call marks. The second subsystem is a wireless sensors network (WSN), which is developed to gather and send data to a central node when the robot reaches a mark.

A. AUTONOMOUS ROBOT

The robot is designed to move at low speed over paths with irregularities, sectors with wet terrain, and obstacles such as some small stones, lumps of earth, potholes, and so on [36], and [37]. Therefore, a D-type (intelligent-sensory) rolling robot approach is employed with caterpillar-shaped gears, which gives grip to the robot and better weight distribution to avoid stalling [38], [39]. Therefore, a double bearing shock-absorbing `SZDoit Smart Robot` type track is used, with dimensions $27.99 \times 24.99 \times 11.51$ cm and 1.1 kg of weight that can support up to 5 kg. The gear motors are `Metal Gearmotor 25Dx65L mm HP 12V with 48 CPR Encoder` and they are used to move the chassis. In addition, they have a speed of $150 \pm 10\%$ rpm at 12 V, a torque of 9.5 kg and 1.200 mA of current consumption, with two quadrature encoders that use Hall effect magnetic sensors [40].

Furthermore, as a controller (uC), an `Arduino Nano 33 BLE sense` is used for the control of the robot movements by using DC motors. Additionally, `HC-SR04` sensors are used to avoid obstacles in front of the vehicle, and a `L298N` dual full-bridge driver is used as a control interface between the uC and gear motors. Moreover, the power supply is a 12 V/9000 mA LiPo battery with its respective charger. The diagram of the connection of all the electronic components is shown in Fig. 1.

As a result, by combining the caterpillar-shaped gears, chassis, and PID control, the autonomous vehicle can

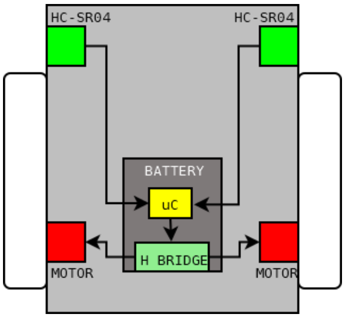


FIGURE 1. Diagram of the connection of all the electronic components of the autonomous robot.

satisfactorily handle roads in rough situations. In addition, ultrasonic sensors (`HC-SR04`) are placed on the front part of the robot to alert obstacles, measure the distance between the crop and the robot, and obtain the location of the robot to make turns, which will be explained in Section VII.

It is worth mentioning that the robot uses the sensors mentioned above, because alternative sensors, such as LIDAR sensors, usually use PWM signals to measure the distance between the sensor and the object, and we do not want to use the PWM pins for this, because we want to keep them available for other applications that will use cameras.

On the other hand, in order to carry out the differential motor control, the velocity of the gear motors is determined. To do this, we take advantage of the fact that the quadrature encoder generates two digital signals out of phase by 90°, due to the rotating magnetic disk installed at the bottom of each motor. Therefore, two-pin change interruptions are used in the `Arduino Nano 33 BLE sense` controller to receive the pulses of the encoder input signal, detect the changes in their logic states and establish the rotation of the shaft of the motor. Next, the encoder resolution, in pulses per revolution (ppr), is determined by using (1) [41].

$$R = m_H s r \tag{1}$$

where $m_H = 75$ is the number of pulses per revolution, $s = 4$ is the numbers of states of the encoder, and $r = 20.4$ is the gearbox ratio. These data were obtained from the specifications of the manufacturer [40]. As a result, $R = 6120$ ppr is the resolution of the encoder. Consequently, with the value of R, the angular velocity is set for the PID controller design (2):

$$\omega = \frac{2\pi n}{tR} \tag{2}$$

where n is the total amount of Hall sensor turns in the encoder, $t = 100$ ms is the sample rate, and ω the angular velocity (rad/s).

For the tuning of controllers in a feedback control loop, it is necessary to determine the dynamic behavior of the plant to be controlled, which in this case is a DC motor. And in order to do this, for convenience, very often a reduced-order system model is used, which could also have a transport lag. This model can be given by (3), which is the transfer function of the plant used in this paper, because here (a) the PID-controller design is aimed at guarantying that the system response to a step change is monotonically increasing, while reaching the new balance point, and (b) the microcontroller has computational limitations.

$$G_p(s) = \frac{k_p e^{-T_l s}}{T_p s + 1} \tag{3}$$

where $s \in \mathbb{C}$ is a frequency parameter, k_p is the gain of the plant and is given by (4), T_l is the transport lag, and T_p is determined from the closed-loop system response to a step change in the input, which is an S-shaped curve [42].

The identification process uses information obtained from an open-loop test, with a step input signal $u(t)$ of size K_u and the DC motor response $y(t)$. Therefore, k_p can be determined by (4).

$$k_p = \frac{Y_u}{K_u} \tag{4}$$

where Y_u is the DC motor response to a step change of size K_u in the input, and the output of the plant is given by (5) [43].

$$y(t) = k_p K_u \left(1 - e^{-(t-T_l)/T_p}\right) \chi_{[T_l,\infty)} \tag{5}$$

where $\chi_{[T_l,\infty)}$ is the indicator function of the interval $[T_l, \infty)$ [44].

On the other hand, regarding the kinematic model of the robot, this model explains the relationship between the point of interest speed and the speed of the DC motors in a two-dimensional reference plane. The Jacobian matrix (6) gives the model for a point of interest (inside the robot) [45]. This model can be graphically appreciated by digitizing the robot and observing its displacement, in order to determine the control point $h(x, y)$ at the center of the vehicle (see Fig. 2).

$$\dot{h}(t) = J(q(t))\dot{q}(t) \Rightarrow \begin{bmatrix} \dot{h}_x \\ \dot{h}_y \end{bmatrix} = \begin{bmatrix} \cos\varphi \\ \sin\varphi \end{bmatrix} \begin{bmatrix} u \\ w \end{bmatrix} \tag{6}$$

where $\dot{h}_x$ and $\dot{h}_y$ are the linear velocities of the robot in the directions of the x and y axes respectively, with orientation vector $\vec{u}$ and angular velocity w, and φ is the robot inclination angle. This can be seen in Fig. 2 using geometry functions for the displacement in x and in y, and applying the derivative to obtain velocities (i.e., $\dot{h}_x = ucos\varphi$ and $\dot{h}_y = ucos\varphi$). Additionally, the second derivative is applied to obtain the acceleration ($\dot{\varphi} = w$).

Pruna et al. [46] establishes that for DC motors and the use of embedded controllers, a standard PID-controller design can be used (7).

$$\begin{aligned} G_c(s) &= K_{pid}\left(\frac{\tau_i \tau_d s^2 + (\tau_i + \tau_d)s + 1}{\tau_i s}\right) \\ &= \frac{K_{pid}(\tau_i + \tau_d)}{\tau_i} + \frac{K_{pid}}{\tau_i s} + K_{pid}\tau_d s \\ &= K_{pid}(1+\delta) + \frac{K_{pid}(1+\delta)}{T_i s} + K_{pid}(1+\delta)T_d s \\ &= K_{pro} + \frac{K_{int}}{s} + K_{der} s \end{aligned} \tag{7}$$

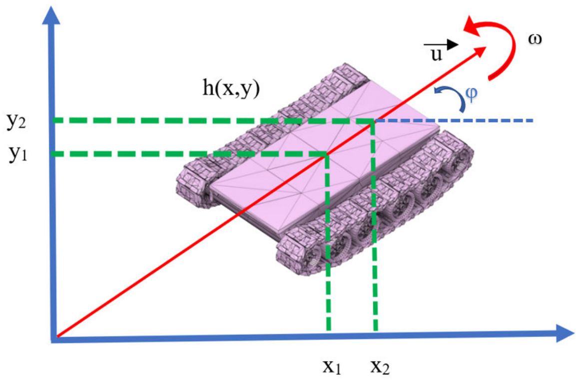


FIGURE 2. Robot-movement kinematic-modeling in the plane.

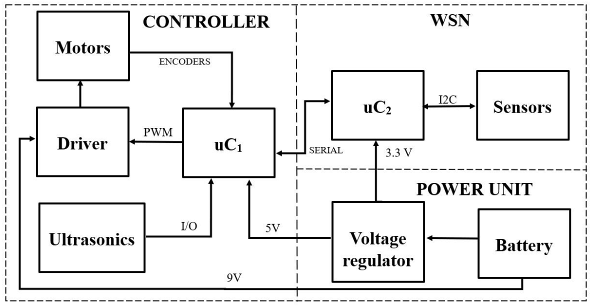


FIGURE 3. Electronic system design.

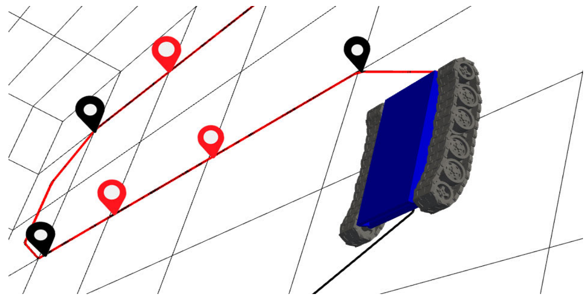


FIGURE 4. Planned routes of the robot in a digitalized environment. Red marks: data collecting points. Black marks: turns' marks.

where K_{pro}, K_{int}, $K_{der} \in \mathbb{R}_{\geq 0}$ are the proportional, integral and derivative coefficients, respectively, of the PID controller [42]. Furthermore, $T_i \in \mathbb{R}_{>0}$ and $T_d \in \mathbb{R}_{\geq 0}$ are the integral time and derivative time, respectively, and $K_{pid} \in \mathbb{R}_{\geq 0}$ is the gain of the transfer function of the PID controller built as the series connection of a PI controller with a PD controller.

The above-mentioned parameters can be updated after carrying out experimental tests aimed at improving the performance of the PID controller, and this process can be performed using tuning functions such as: Lambda, Ziegler-Nichols, and Integral Absolute Error (IAE) performance index [46], [47], [48].

The robot has a LiPo battery for the power supply. This battery has cells with a voltage between 3 V and 4.2 V to achieve 12 V and 9000 mA. Therefore, a voltage divider is built at 4 V as the maximum safe value, and 2.8 V as the minimum value. This way, the microcontroller can monitor the voltage and send alerts if the battery is discharging.

B. WIRELESS SENSOR NETWORK

The Food and Agriculture Organization of the United Nations (FAO) shows that the climatic variables that directly influence the crops inside the greenhouse are temperature, relative humidity, CO_2, and amount of UV rays [6], [13]. Thus, several brand-new sensors are available to deploy WSN nodes. Then, precision and accuracy are relevant in selecting a suitable sensor, especially for CO_2. However, there are other requirements to consider, such as: size and communication protocols. As a result, the chosen sensors were as follows: (1) The VEML6075 UVA / UVB / UV Index Sensor. This sensor has a photodiode that measures ultraviolet (UV) radiation levels, A (320 - 400 nm) and B (280 - 320 nm), allowing the calculation of the UV index with a variation of ± 10 nm and that sends information using I_2C communication, with a resolution of 16 bits. And (2), for the measurement of CO_2, humidity, and temperature the SCD30 sensor was used. For the measurement of CO_2, the accuracy of the SCD30 sensor is ± 30 ppm ± 3% (25°C, 400 - 10000 ppm). The humidity has a variability of 3% on a scale from 0 to 95%, and the temperature has a variability of 0.5°C, with measurements up to 70°C.

The above-mentioned sensors were chosen based on functionality, features, and usability requirements. For example, they have the I_2C interface to send data to the controller. Also, they have proper libraries to calibrate them and also put them in sleep mode.

In addition, the SparkFun LoRa Gateway device was also used as a controller, because it is a powerful 3-network capable device thanks to an onboard ESP32 WROOM module and an RFM95W LoRa modem. The RFM95W handles the 915MHz band, while the ESP32 takes care of Bluetooth and WiFi capabilities. The LoRa Gateway can act as either a gateway (hence the name) or a device, but not both simultaneously. Finally, the Arduino Nano 33 BLE sense is the controller that was used for the electronic design of the robot. These microcontrollers communicate between them by serial communication. Fig. 3 shows the block diagram of the proposed electronic system.

IV. ROUTES PLANNING WITH LoRa COMMUNICATION

Error-free PID control operation allows planning the robot route in a two-dimensional chart. In this scenario, the farmer can determine the distance of each data collection point regarding the crop type. For example, the farmer can configure the robot to collect data every 2 meters for small crops or every 5 meters for large crops. In addition, knowing the dimensions of the greenhouse and the beds of the crops, we can decide on "marks" to make turns and keep collecting data on the rest of the crop. Additionally, using a Python interface developed in this research, the robot and its characteristics were digitized to test the route and the accuracy of reaching the data collecting point or the turns' mark. Fig. 4 shows a simulation of the designed robot.

The energy needed to send data through the sensor node to the central node is closely related to how long the data is in the air. Therefore, this parameter allows for defining the transmission power of the network, the wireless protocol type to be used, and the processing time required to send information [49], [50]. The protocols with a connection range of one kilometer to cover the distance between the greenhouse and the central node are LoRaWan and ZigBee. However, ZigBee is a restrictive protocol under design conditions since it is necessary to implement additional hardware. Consequently, we use a `SparkFun LoRa Gateway` with a `U.FL` connector. Additionally, this protocol works in the 868 MHz and 915 MHz band [51]. To define the transmission power of each WSN node, the calculation of time on the air (ToA) is given by (8). Lastly, the central node is a Raspberry 4.0 with a `SparkFun LoRa Gateway` to receive data from the robot; it is placed in the delivery room, which is 400 meters from the greenhouse.

$$ToA = TT[100 - (duty\ cycle)] \tag{8}$$

where TT is the processing time, which in this microcontroller is 200 ms, and the *duty cycle* is considered to be just the 1% of the entire firmware. In this case, a $ToA =$ 19.8 s is defined as the send time between messages to avoid collisions. We define $Ptx = 20$ dBm as the transmission power, $Ga = 10$ dBm as the antenna gain, and $Pc =$ 5 dBm as the cable loss. Then, a link power of 25 dBm is obtained with a transmission speed between 0.3 Kbps and 22 Kbps.

V. ML PIPELINE

This section shows the ML data analysis to support the farmer's decision-making. Therefore, given that the greenhouse does not have previous information about the crop, unsupervised learning was the suitable solution.

A. DATA COLLECTION

Given that the robot is designed to collect data from the environmental conditions of the crops, with the farmer's support, the central started to store samples that the robot sends by LoRa. The samples were taken in three different schedules. First, in the morning, after the water pump sprays the entire crop. Second, in the afternoon, when the crop receives the maximum ray UV. And third, at night, when the temperature decreases until its minimum value. Therefore, there are 40 data collection points around the crop and 5 turn marks. The WSN collects 10 consecutive samples in each mark to save the average value to reduce sensor errors and DC peaks. As a result, when the robot finishes its day route, the central node receives 120 samples. The Ruscus crops, which are large green leaves (around 1 meter) to add to the flower buckets, need 6 months to harvest. Therefore, the data collection stage took 4 months to have time to resolve the crop's issues. As a result, 13.440 samples were taken of each variable, representing 215 MB.

B. MODEL DESIGN

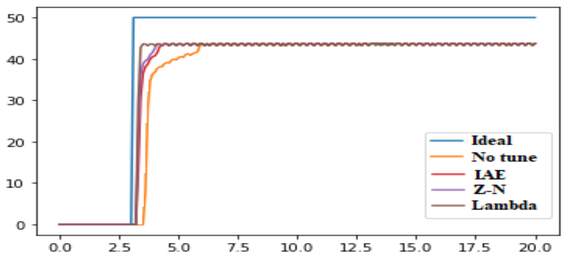
One of the principal challenges in embedded systems is the ability to recognize features to make groups with similar characteristics. Therefore, `k-means` method aims to partitioning a set of n observations into k groups with lightweight computational cost. Due to its straightforward method, the centroids of each cluster could be easily imported into robots with constrained hardware [38]. Each value of n belongs to a group of k whose mean distance value is the closest [52]. However, the randomness of the k values can lead to different forms of clustering. Therefore, it is necessary to properly define its value to group only the data containing the largest number of similar attributes. Consequently, the version of `k-means ++` allows eliminating this problem by analyzing a set of observations $(x_1, x_2, \ldots, x_n)$, where each observation is a real d-dimensional vector, and the `k-means` grouping aims at dividing the n observations into $k \leq n$ sets $S = S_1, S_2, \ldots, S_k$. Here, the variability of the observations is measured by using the within-cluster sum of squares (WCSS) [53].

C. MODEL EVALUATION

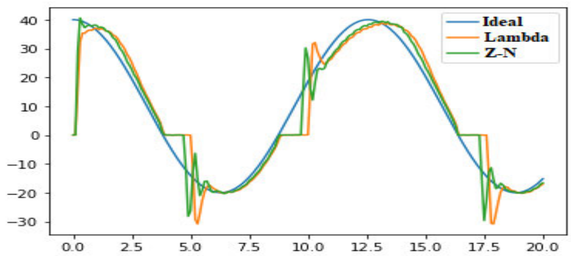
`k-means` algorithm allows making groups division approach quantitatively. Hence, the WCSS metric measures the average square distance of all data to the centroid, which determines the affinity of the groups. As a result, the distance between them can be observed, and the value of k can be better chosen. Therefore, we create weekly training tests to define the k value (i.e., cluster numbers) with daily data. We tested the k value from zero to ten, and when WCSS began to level off (elbow method). As a result, $k = 3$ was chosen as a suitable parameter. This process was carried out weekly to confirm that the parameter remained constant. Then, with the support from related researchers in the agricultural sector and the farmer, it was possible to establish which group belongs to each data sample for the clusters. Here, it is important to point out that the `k-means` algorithm was developed and tested in a Python environment due to the good libraries it has for this purpose.

VI. RESULTS

The autonomous robot's outcome aims to collect data from the greenhouse to represent three different levels of the environmental conditions of the crops to support the farmer's decision-making. This section shows the PID control designed to control DC motors while the robot moves around the greenhouse. Therefore, the data collection process starts by powering the robot with one switch attached to the electronic board. Then, the robot automatically sends messages to the central node to establish wireless communication. Once the communication is set, the farmer can put the robot on the initial mark inside the greenhouse. Then, the robot moves around the crop collecting data and sending data to the central node to visualize those data. When the robot reaches the final mark, since it stores the mark numbers that it needs to



(a) Process variable response in the tuning of the PID controller.



(b) Curvilinear paths results.

FIGURE 5. **Some results of the tuning functions tests (Z-N stands for Ziegler Nichols).**

reach, it sends the final payload with an additional message indicating that the job is done. Finally, the robot goes to sleep mode until awake again if the central node sends an activation message from the GUI.

A. PID CONTROL

The reaction time of the DC motors can be fine-tuned using the previously described tuning functions, where a PID control is applied. The transfer function of each DC motor is given by (9), and it is necessary to determine the best tuning function for the case under study. In other words, it is desired to determine which tuning function best adjusts the parameters of the PID controller (i.e., K_{pro}, K_{int}, and K_{der}) to obtain the best performance from the motors.

$$G_p(s) = \frac{1.8785\ e^{-0.0897\ s}}{0.0966\ s + 1} \tag{9}$$

Therefore, tuning function tests were carried out to know the response of the DC motors (see Fig. 5(a)) in several real-life trial experiments. In this research, the `Lambda` and `Ziegler-Nichols` tuning functions provided the best results in the linear motor drive [46]. However, in curvilinear paths, which depend on better motor control, when simulating a sinusoidal trajectory, the `Ziegler-Nichols` tuning showed a more significant number of errors in dead time tests. On the other hand, the `Lambda` tuning showed a better representation of the path and fewer errors in the system activation. The curvilinear path results can be seen in Fig. 5(b). Finally, Table 1 shows the PID controller parameters obtained from the above-mentioned fine-tuning methods.

B. ML PIPELINE

Agricultural experts take the collected data and annotate their corresponding cluster's names. They mention the cluster's characteristics, based on their knowledge and industrial tools, to compare the results with high-level farming machines. Furthermore, they recommend setting the cluster's names as follows: 1) standard (good humidity and temperature with moderate UV rays exposure levels and normal CO_2 emissions), 2) optimum (good humidity and temperature with low UV rays exposure levels and normal CO_2 emissions), and 3) deficient (high humidity and temperature with either low or high UV rays exposure levels and either low or high CO_2 emissions). Then, we export the centroids to the robot to decide the corresponding cluster to each mark. This model is evaluated weekly to refresh the centroids and check the decision capacity of the robot. In real scenarios, the robot showed to choose the correct cluster over 90% of the times, coinciding in good agreement with the true farmer experience in each decision. Fig. 6 shows the graphical result of the `k-means` algorithm in the greenhouse crops of this paper. However, this local decision from the robot allows the farmer to understand the differences between activating the water pump in the exact schedules and different ones.

TABLE 1. **Parameters of the PID controller.**

Method	Parameters		
	K_{pro}	K_{int}	K_{der}
IAE	0.85	0.10	0.16
Ziegler-Nichols	0.68	0.08	0.04
Lambda	0.23	0.14	0.03

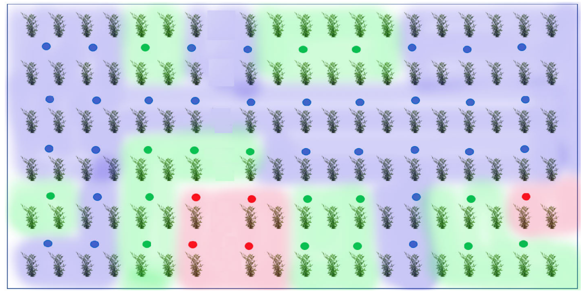


FIGURE 6. **Graphical mapping by means of the `k-means` algorithm. Blue: standard, green: optimum, and red: deficient.**

C. GUI AND WSN

To ensure the data obtained from the robot can be considered trustworthy, we compared their values with reference-grade instruments. For instance, for the CO_2 variable, the Testo 315-3 - CO and CO_2 meter took samples simultaneously with the WSN inside the greenhouse. This comparison demonstrated that the data obtained from the WSN had a 5% error in representing the environmental conditions of each variable. Then, the communication channel is checked to avoid losing data. Therefore, the Received Signal Strength Indicator (RSSI) is set into the central node to measure the communication power. We figured out the longest distance the robot can send information is around 1.4 km, where the central node received packages from the WSN between −80 dBm and −90 dBm of power, and the package took around 120 ms to reach the central node. As a result, LoRa communication is enough to covert the greenhouse. The package's payload has one array split by commas of each environmental variable, the battery power, the decision made by the robot, and the communication channel metrics.

The payload is processed by the central node (Raspberry Pi), which also runs the GUI to show the farmer relevant information about the crops' environmental conditions, the battery, and the transmission status. The data is shown in bar graphs. Once the robot reaches the final mark and the last payload is sent to the central node, the GUI will show the greenhouse sections where the environmental conditions are optimal, standard, or deficient. The information will store on the device for further data analysis. Lastly, the farmer can set the gap between marks for the following data collection round. The GUI is shown in Fig. 7.

D. PROPOSED SMART-FARMING ROBOT FOR THE HARVESTING DECISION SUPPORT

One switch turns on the robot that is directly connected to the battery and the voltage regulator. The WSN and robot controllers communicate via a serial protocol, where the robot sends a message to the WSN node saying that it has found a data collection point. Next, upon receiving the message, the WSN node activates the sensors and collects data showing measurements of environmental variables in the greenhouse. After doing this, the electronic system makes a decision and sends it together with the obtained data to the central node to be displayed in the GUI. Finally, the WSN node responds to the robot with a message telling it to continue to the next mark. In this way, the electronic system has real-trial measurement results on the environmental condition of the greenhouse. Then, we tested the battery consumption to check how long the robot could work without recharging its battery. It took the robot 2 hours to go down all the greenhouse crop paths (2500 square meters), following previously-planned routes, and it was found that the robot could do the above four times (8 hours) on a single charge.

Finally, when the robot sends a low-battery alert signal, the farmer takes the robot to his office to charge its battery by using a battery charger. In addition, he does not need to take the battery out of the robot to charge it, because the battery has two terminals that connect to the charger. In short, it has two banana plugs (colored red and black) to connect the charger to the battery. Figure 8 shows the proposed robot while working in a greenhouse.

VII. DISCUSSION

The main concern about the autonomous robot is controlling its navigation around the greenhouse without losing its position to reach the next mark. Even when the lightweight PID control correctly works in controlled environments, it might fail in unbalanced paths or when cultivation beds are not appropriately aligned between them in trial-real scenarios. Therefore, HC-SR04 ultrasonic sensors play a relevant role in checking the distance between the crop and the robot, especially when the robot needs to make turns. One sensor is placed on the left top of the robot at 45 inclination degrees, and the other is placed on the right top at 90 degrees. The ultrasonic sensor's data sheet specifies 15 error degrees. Therefore, Fig 9 shows how the sensor can determine the expected distance (Z_2) and its possible errors (Z_1 and Z_3). Those distances create rectangle triangles with the crops and robot (XYZ_1, XYZ_2, and XYZ_3). Following the same scenario, the next ultrasonic sensor makes the isosceles triangle with the greenhouse's border, and the robot (C_1C_2A), segment $\overline{AB}$ is the expected distance. The robot is configured to work 40 cm from the crops (segment $\overline{XY}$); therefore, using (10)-(15) to get $\hat{Z}$, $\hat{XY}$ and $\hat{AB}$ by taking three samples consecutively from each sensor, representing the average distance obtained by considering the sensor's degrees of error.

Using (12)-(15) and the following values: $\beta = 45°$, $\alpha = 7.5°$, $\overline{XY} = 40$ cm and $\overline{AB} = 100$ cm, Z_1, Z_2, and Z_3 were obtained to get $\hat{Z}$, which is 59.91 cm. Compared with the optimal value $Z_2 = 63.63$ cm, the average error is around 4 cm. Then with the $\hat{Z}$ value, the robot can get the distance to crops, which has an error of $\pm$ 2.36 cm of the initial settings ($\overline{XY} =$ 40 cm). The limited value is $\overline{AB}$ to detect the greenhouse's border, which the sensor needs to recognize. However, with the sensor's error readings, it can detect the $\hat{AB}$, which is 99.4 cm instead of 100 cm. This data is compared with the turns' marks to update the robot's location and check its error. Besides, the sensor on the robot's top left could help recognize the crops' end and update its actual location with the next turn's mark. In real-trial experiments, the robot could keep the 40 cm distance by sending commands to the PID control to update the motor velocity and return to the planned route. Furthermore, by updating the robot's location with ultrasonic sensors, the average error was 10 cm to the mark's turns, which is acceptable in the working conditions that the robot is facing. Lastly, the farmer must place the robot at a specific starting reference point within the greenhouse before beginning to collect data. From there, the robot knows on which mark it is located and the path it has to travel. The robot then follows the established path and the GUI helps the farmer to make decisions about the harvest process.

$$Z_{1,3} = \overline{XY}/cos(\beta \pm \alpha) \tag{10}$$
$$Z_2 = \overline{YX}/cos(\beta) \tag{11}$$
$$\hat{Z} = (Z_1 + Z_2 + Z_3)/3 \tag{12}$$
$$\hat{XY} = \hat{Z} \cdot cos(\beta) \tag{13}$$
$$C_{1,2} = \overline{AB} \cdot sin(\alpha) \tag{14}$$
$$\hat{AB} = (\overline{AB} + C_{1,2})/3 \tag{15}$$

To better understand the functionalities of the proposed robot and the novelty of this work, we will now compare our device with others that are also intended to provide industrial solutions. In short, smart-farming robots such as Carbon Robotics and Tortuga Agtech provide different AI solutions to help the farmer monitor both crops and the harvesting process. From our point of view, these robots have very good characteristics. However, we think that to use them in the specific application to which we have given a solution in

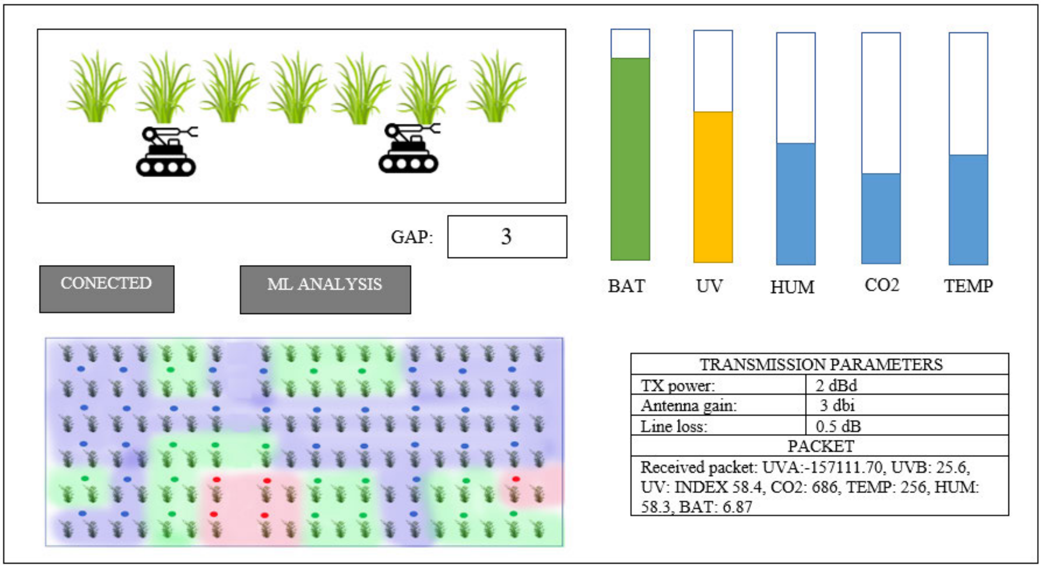


FIGURE 7. GUI to monitor crop conditions in greenhouses by the autonomous robot.

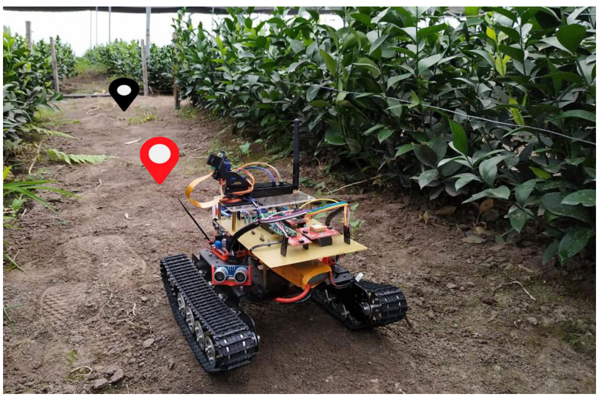


FIGURE 8. Prototype of the proposed-smart-farming robot while collecting data in a greenhouse.

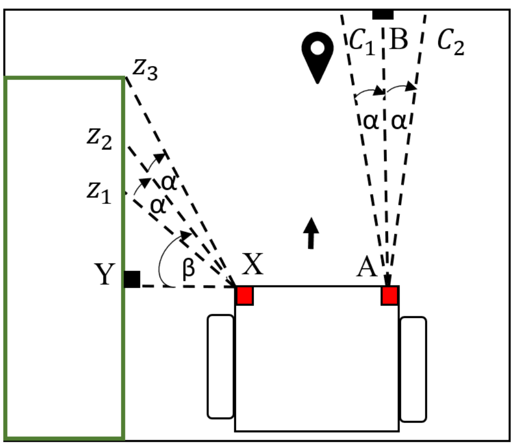


FIGURE 9. Ultrasonic sensors functionality explanation.

our research, the farmer would have to make changes to be able to adapt to the characteristics of the robots. These changes would be due to the size of these robots and their functionalities. This could be expensive and might reduce the profit of the farmer. In addition, the above-mentioned robots need more maintenance from experts, which also increases the production cost.

In [7], the prototype of a mobile robot to collect data in agricultural fields was shown, and the motivation of that research was to combine precision agriculture, IoT and robotics. However, according to the authors of [7], the work presented in that paper needs to be further developed in different areas. For example, its design should be improved as an autonomous system that can maneuver in the fields. Additionally, to that system must be added capabilities for precision agriculture and mapping of agricultural fields, using sensors that, according to the authors of [7], could be expensive and fragile, such as a spectrometer. Furthermore, as future work, the authors of [7] will intend to make their design robust and reliable using software engineering. Moreover, to provide the needed information properly and visualize the data, the design proposed in [7] requires a good website. Finally, in [7] good internet connection is needed to share data to the cloud, which is not an easy requirement to fulfill in greenhouses.

On the other hand, the smart-farming robot presented in our research does not have most of the aforementioned problems. In addition, it is able to plan routes without interfering with normal greenhouse cultivation activities and can help farmers who do not have previous crop data to improve the harvest process. Furthermore, the proposed system has a GUI that helps farmers to improve decision-making in the harvest process and that does not need to exchange information with the cloud or big servers.

Other works aimed at detecting the growth of agricultural products, their location, and the perception of the environment are [31], [32]. On the one hand, these works focus on the use of artificial intelligence techniques to process images in agricultural applications. But on the other hand, they neither take into account the design of the smart-farming robot nor the design of the sensor network needed to measure the climatic variables that, according to FAO, directly influence the crops inside the greenhouse (see Section III-B). In short, our novelty also lies in how two cutting-edge technologies can be combined (i.e., ML models and farming robots) to interact with the farmer without previous information about the crop and help him to achieve success over a long time. Moreover, designing a lightweight PID control to run into small robots and an ML model with a small memory footprint is a new challenge that this work fulfills.

Finally, we would not like to end this section without highlighting that all the papers that have been discussed here present devices that are very good, which, of course, have advantages and disadvantages. And that these papers, like many others, show that worldwide there is a great interest in improving agricultural production using robotics, smart sensors, IoT, artificial intelligence, and so on.

VIII. CONCLUSION

The main conclusions of this paper can be summarized as follows:

- The wireless sensor network reached the objective of acquiring valuable data from the greenhouse. Indeed, taking the average of ten samples reduces the chance of getting extreme observations significantly. Therefore, the ML model can describe the actual environmental conditions in the greenhouse.
- The designed PID controller gives the opportunity of planning straight and curvilinear trajectories that adapt to the shape of the crop. In addition, having an interface allows excellent flexibility in using the different products that can be grown in the greenhouse.
- For the specific greenhouse where the smart farming robot was tested, the proposed data analysis effectively determined the zones of the greenhouse where the environmental conditions affected the crops. Specifically, the deficient zone was in the location near the door and water pump. Farmers often do not close the door correctly, and those parts of the crop are directly exposed to the sunlight. Moreover, in some cases, it is well-known that the water pump is faulty, and consequently it continues supplying water to the crop, even after the pump key has already been closed.

As future works, we plan to build an autonomous robot with computer vision to give the farmer more information about his crops and the possibility of activating water pumps using smart systems decisions based on efficient data processing techniques.

REFERENCES

[1] A. Soheyb, T. Abdelmoutia, and T. S. Labib, "Toward agriculture 4.0: Smart farming environment based on robotic and IoT," in *Proc. 4th Int. Symp. Adv. Electr. Commun. Technol. (ISAECT)*, Dec. 2021, pp. 1–5.

[2] S. Garg, P. Pundir, H. Jindal, H. Saini, and S. Garg, "Towards a multimodal system for precision agriculture using IoT and machine learning," in *Proc. 12th Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT)*, Jul. 2021, pp. 1–7.

[3] J. Pak, J. Kim, Y. Park, and H. I. Son, "Field evaluation of path-planning algorithms for autonomous mobile robot in smart farms," *IEEE Access*, vol. 10, pp. 60253–60266, 2022.

[4] Y.-Y. Zheng, J.-L. Kong, X.-B. Jin, X.-Y. Wang, and M. Zuo, "CropDeep: The crop vision dataset for deep-learning-based classification and detection in precision agriculture," *Sensors*, vol. 19, no. 5, p. 1058, Mar. 2019.

[5] K. Fathallah, M. Abid, and N. B. Hadj-Alouane, "Enhancing energy saving in smart farming through aggregation and partition aware IoT routing protocol," *Sensors*, vol. 20, p. 2760, May 2020.

[6] *Smart Farming is Key for the Future of Agriculture | FAO*, FAO, Rome, Italy, 2017.

[7] H. Durmus and E. O. Günes, "Integration of the mobile robot and Internet of Things to collect data from the agricultural fields," in *Proc. 8th Int. Conf. Agro-Geoinformatics (Agro-Geoinformatics)*, Jul. 2019, pp. 1–5.

[8] B. Singh, M. Kaur, S. Soni, J. Singh, A. Kumar, and A. Das, "Spatio-temporal mapping of green house gas emission in urban settings using a vehicle mounted IoT enabled pollution sensing modules," in *Proc. 4th Int. Conf. Inf. Syst. Comput. Netw. (ISCON)*, Nov. 2019, pp. 366–369.

[9] A. Z. M. T. Kabir, A. M. Mizan, N. Debnath, A. J. Ta-Sin, N. Zinnurayen, and M. T. Haider, "IoT based low cost smart indoor farming management system using an assistant robot and mobile app," in *Proc. 10th Electr. Power, Electron., Commun., Controls Informat. Seminar (EECCIS)*, Aug. 2020, pp. 155–158.

[10] S. Yang, J. Ji, H. Cai, and H. Chen, "Modeling and force analysis of a harvesting robot for button mushrooms," *IEEE Access*, vol. 10, pp. 78519–78526, 2022.

[11] N. N. Misra, Y. Dixit, A. Al-Mallahi, M. S. Bhullar, R. Upadhyay, and A. Martynenko, "IoT, big data, and artificial intelligence in agriculture and food industry," *IEEE Internet Things J.*, vol. 9, no. 9, pp. 6305–6324, May 2022.

[12] Y. Chen and Y. Li, "Intelligent autonomous pollination for future farming—A micro air vehicle conceptual framework with artificial intelligence and human-in-the-loop," *IEEE Access*, vol. 7, pp. 119706–119717, 2019.

[13] X. Han, J. A. Thomasson, T. Wang, and V. Swaminathan, "Autonomous mobile ground control point improves accuracy of agricultural remote sensing through collaboration with UAV," *Inventions*, vol. 5, no. 1, p. 12, Mar. 2020.

[14] P. D. Rosero-Montalvo, V. C. Erazo-Chamorro, V. F. López-Batista, M. N. Moreno-García, and D. H. Peluffo-Ordóñez, "Environment monitoring of Rose crops greenhouse based on autonomous vehicles with a WSN and data analysis," *Sensors*, vol. 20, no. 20, p. 5905, Oct. 2020.

[15] S. P. Adhikari, G. Kim, and H. Kim, "Deep neural network-based system for autonomous navigation in paddy field," *IEEE Access*, vol. 8, pp. 71272–71278, 2020.

[16] R. Rayhana, G. Xiao, and Z. Liu, "Internet of Things empowered smart greenhouse farming," *IEEE J. Radio Freq. Identificat.*, vol. 4, no. 3, pp. 195–211, Sep. 2020.

[17] M. Erazo, D. Rivas, M. Perez, O. Galarza, V. Bautista, M. Huerta, and J. L. Rojo, "Design and implementation of a wireless sensor network for Rose greenhouses monitoring," in *Proc. 6th Int. Conf. Autom., Robot. Appl.*, Apr. 2015, pp. 256–261.

[18] R. K. Singh, R. Berkvens, and M. Weyn, "Energy efficient wireless communication for IoT enabled greenhouses," in *Proc. Int. Conf. Commun. Syst. Netw. (COMSNETS)*, Jan. 2020, pp. 885–887.

[19] M. Boukens, A. Boukabou, and M. Chadli, "A real time self-tuning motion controller for mobile robot systems," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 1, pp. 84–96, Jan. 2019.

[20] J. Jiang and M. Moallem, "Development of an intelligent LED lighting control testbed for IoT-based smart greenhouses," in *Proc. IECON 46th Annu. Conf. IEEE Ind. Electron. Soc.*, Oct. 2020, pp. 5226–5231.

[21] S. Nesteruk, D. Shadrin, M. Pukalchik, A. Somov, C. Zeidler, P. Zabel, and D. Schubert, "Image compression and plants classification using machine learning in controlled-environment agriculture: Antarctic station use case," *IEEE Sensors J.*, vol. 21, no. 16, pp. 17564–17572, Aug. 2021.

[22] E. Elbasi, N. Mostafa, Z. AlArnaout, A. I. Zreikat, E. Cina, G. Varghese, A. Shdefat, A. E. Topcu, W. Abdelbaki, S. Mathew, and C. Zaki, "Artificial intelligence technology in the agricultural sector: A systematic literature review," *IEEE Access*, vol. 11, pp. 171–202, 2023.

[23] A. Dutta, S. Roy, O. P. Kreidl, and L. Boloni, "Multi-robot information gathering for precision agriculture: Current state, scope, and challenges," *IEEE Access*, vol. 9, pp. 161416–161430, 2021, doi: 10.1109/ACCESS.2021.3130900.

[24] D. Xue and W. Huang, "Smart agriculture wireless sensor routing protocol and node location algorithm based on Internet of Things technology," *IEEE Sensors J.*, vol. 21, no. 22, pp. 24967–24973, Nov. 2021.

[25] M. H. Ko, B. Ryuh, K. C. Kim, A. Suprem, and N. P. Mahalik, "Autonomous greenhouse mobile robot driving strategies from system integration perspective: Review and application," *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 4, pp. 1705–1716, Aug. 2015.
[26] P. Glotfelter, I. Buckley, and M. Egerstedt, "Hybrid nonsmooth barrier functions with applications to provably safe and composable collision avoidance for robotic systems," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 1303–1310, Apr. 2019.
[27] R. Polvara, F. D. Duchetto, G. Neumann, and M. Hanheide, "Navigate-and-seek: A robotics framework for people localization in agricultural environments," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 6577–6584, Oct. 2021.
[28] S. Fernando, R. Nethmi, A. Silva, A. Perera, R. D. Silva, and P. W. K. Abeygunawardhana, "AI based greenhouse farming support system with robotic monitoring," in *Proc. IEEE REGION 10 Conf. (TENCON)*, Nov. 2020, pp. 1368–1373.
[29] M. Edmonds and J. Yi, "Efficient multi-robot inspection of row crops via kernel estimation and region-based task allocation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 8919–8926.
[30] S. J. LeVoir, P. A. Farley, T. Sun, and C. Xu, "High-accuracy adaptive low-cost location sensing subsystems for autonomous rover in precision agriculture," *IEEE Open J. Ind. Appl.*, vol. 1, pp. 74–94, 2020.
[31] Y. Ge, Y. Xiong, G. L. Tenorio, and P. J. From, "Fruit localization and environment perception for strawberry harvesting robots," *IEEE Access*, vol. 7, pp. 147642–147652, 2019.
[32] R. Fernández, H. Montes, J. Surdilovic, D. Surdilovic, P. Gonzalez-De-Santos, and M. Armada, "Automatic detection of field-grown cucumbers for robotic harvesting," *IEEE Access*, vol. 6, pp. 35512–35527, 2018.
[33] L. Zhang, J. Jia, G. Gui, X. Hao, W. Gao, and M. Wang, "Deep learning based improved classification system for designing tomato harvesting robot," *IEEE Access*, vol. 6, pp. 67940–67950, 2018.
[34] S. Fernando, R. Nethmi, A. Silva, A. Perera, R. D. Silva, and P. K. W. Abeygunawardhana, "Intelligent disease detection system for greenhouse with a robotic monitoring system," in *Proc. 2nd Int. Conf. Advancements Comput. (ICAC)*, vol. 1, Dec. 2020, pp. 204–209.
[35] H. Durmus, E. O. Günes, and M. Kirci, "Disease detection on the leaves of the tomato plants by using deep learning," in *Proc. 6th Int. Conf. Agro-Geoinformatics*, Aug. 2017, pp. 1–5.
[36] X. Lu and J. Fei, "Tracking control of wheeled mobile robots using iterative learning controller," in *Proc. 15th Int. Conf. Control, Autom. Syst. (ICCAS)*, Oct. 2015, pp. 1172–1176.
[37] F. Leonardi, F. Messina, and C. Santoro, "A risk-based approach to automate preventive maintenance tasks generation by exploiting autonomous robot inspections in wind farms," *IEEE Access*, vol. 7, pp. 49568–49579, 2019.
[38] Y. Ma, Z. Wang, H. Yang, and L. Yang, "Artificial intelligence applications in the development of autonomous vehicles: A survey," *IEEE/CAA J. Autom. Sinica*, vol. 7, no. 2, pp. 315–329, Mar. 2020.
[39] D. Mitchell, J. Blanche, O. Zaki, J. Roe, L. Kong, S. Harper, V. Robu, T. Lim, and D. Flynn, "Symbiotic system of systems design for safe and resilient autonomous robotics in offshore wind farms," *IEEE Access*, vol. 9, pp. 141421–141452, 2021.
[40] *CHIHAI MOTOR GM25–370 DC Gear Motor 6V 100/210/300RPM Encoder Motor*, CHIHAI MOTOR, BANGGOOD, Baiyun, Guangzhou, 2022.
[41] A. Konno and M. Uchiyama, "Modeling of a flexible manipulator dynamics based upon Holzer's model," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 1996, pp. 223–229.
[42] K. Åström and R. M. Murray, *Feedback Systems: An Introduction for Scientists Engineers*, 2nd ed. Princeton, NJ, USA: Princeton Univ. Press, 2021.
[43] T. Takagi, Y. Hirama, M. Harita, H. Hamane, and K. Miyazaki, "One parameter tuning method for PID controller," in *Proc. SICE Annu. Conf.*, Sep. 2011, pp. 2939–2943.
[44] W. F. Trench, *Introduction to Real Analysis*. London, U.K.: Pearson, 2003.
[45] F. Nadi, V. Derhami, and M. Rezaeian, "Visual servoing control of robot manipulator with Jacobian matrix estimation," in *Proc. 2nd RSI/ISM Int. Conf. Robot. Mechatronics (ICRoM)*, Oct. 2014, pp. 405–409.
[46] E. Pruna, E. R. Sasig, and S. Mullo, "PI and PID controller tuning tool based on the lambda method," in *Proc. CHILEAN Conf. Electr., Electron. Eng., Inf. Commun. Technol. (CHILECON)*, Oct. 2017, pp. 1–6.
[47] K. H. Ang, G. Chong, and Y. Li, "PID control system analysis, design, and technology," *IEEE Trans. Control Syst. Technol.*, vol. 13, no. 4, pp. 559–576, Jul. 2005.
[48] R. C. Dorf and R. H. Bishop, *Modern Control Systems*, 14th ed. London, U.K.: Pearson, 2022.
[49] S. Jeong, T. Sung, K. E. Lee, and J. Kang, "Joint TOA/AOA-based localization in wireless sensor networks," in *Proc. 8th Int. Conf. Signal Process. Commun. Syst. (ICSPCS)*, Dec. 2014, pp. 1–5.
[50] O. Khutsoane, B. Isong, N. Gasela, and A. M. Abu-Mahfouz, "WaterGrid-sense: A LoRa-based sensor node for industrial IoT applications," *IEEE Sensors J.*, vol. 20, no. 5, pp. 2722–2729, Mar. 2020.
[51] X. Wang, L. Kong, L. He, and G. Chen, "MLoRa: A multi-packet reception protocol in LoRa networks," in *Proc. IEEE 27th Int. Conf. Netw. Protocols (ICNP)*, Oct. 2019, pp. 1–11.
[52] B. Araujo, *Aprendizaje Automático: Conceptos Básicos y Avanzados: Aspectos Prácticos Utilizando el Software Weka*. London, U.K.: Pearson, 2006.
[53] P. Joshi, D. Das, V. Udutalapally, M. K. Pradhan, and S. Misra, "RiceBioS: Identification of biotic stress in Rice crops using edge-as-a-service," *IEEE Sensors J.*, vol. 22, no. 5, pp. 4616–4624, Mar. 2022.

PAUL D. ROSERO-MONTALVO received the Engineering degree in electronics and telecommunications from Universidad Técnica del Norte (UTN), Ecuador, in 2013, the master's degree in data management systems from Universidad de las Fuerzas Armadas ESPE, Ecuador, in 2018, and the Ph.D. degree from the University of Salamanca, Spain, in November 2020, under the supervision of Prof. Vivian Lopez-Batista. He was a Research Assistant Professor with the Applied Science Department, UTN, for seven years. At the same time, he was a part-time Lecturer with the IT Department, Instituto Tecnologico Superior 17 de Julio, Ecuador. He has been a Postdoctoral Researcher with the IT University of Copenhagen (ITU), Denmark, since June 2021. His current research interest includes emerging microcontrollers to run machine learning models in decentralized networks.

CARLOS A. GORDILLO-GORDILLO received the degree in electronic and network communication engineering from Universidad Técnica del Norte, Ecuador. He is currently pursuing the Ph.D. degree with the University of Salamanca, Spain. His current research interests include embedded systems, the IoT, and robots.

WILMAR HERNANDEZ (Senior Member, IEEE) received the degree in electronics engineering and the Specialist degree in microelectronics from Instituto Superior Politecnico Jose Antonio Echeverria (ISPJAE), Havana, Cuba, in 1992 and 1994, respectively, and the M.S. degree in signal treatment and the Ph.D. degree in electronic engineering from Enginyeria La Salle, Universitat Ramon Llull, Barcelona, Spain, in 1997 and 1999, respectively. From 1992 to 1995, he was a Lecturer with the Electrical Engineering Faculty, ISPJAE, and a Researcher with the Microelectronics Research Center, ISPJAE. From 1999 to 2003, he was with the Department of Electronics and Instrumentation, University Institute for Automobile Research, Universidad Politecnica de Madrid (UPM), Spain, where he was the Technical Director of the Department of Electronics and Instrumentation, from January 2003 to January 2004. From January 2004 to March 2013, he was an Associate Professor of circuits and systems with the Department of Circuits and Systems, EUIT de Telecomunicacion, UPM. From September 2014 to September 2015, he was a Researcher with SENESCYT, Ecuador, under the Prometeo Fellowship Program. From December 2015 to November 2017, he was a Professor with Universidad Tecnica Particular de Loja, Ecuador. Since January 2018, he has been a Professor with Universidad de Las Americas, Ecuador.