



Robot for Navigation in Maize Crops for the Field Robot Event 2023

David Iván SÁNCHEZ-CHÁVEZ, Noé VELÁZQUEZ-LÓPEZ*, Guillermo GARCÍA-SÁNCHEZ,
Alan HERNÁNDEZ-MERCADO, Omar Alexis AVENDAÑO-LOPEZ, Mónica Elizabeth BERROCAL-AGUILAR
Universidad Autónoma Chapingo, Carretera Federal México-Texcoco Km 38.5, Texcoco, Z.C. 56230, México

ABSTRACT

Navigation in a maize crop is a crucial task for the development of autonomous robots in agriculture, with numerous applications such as spraying, monitoring plant growth and health, and detecting weeds and pests. The Field Robot Event 2023 (FRE) continued to challenge universities and other research teams to push the development of algorithms for agricultural robots further. The Universidad Autónoma Chapingo has been developing a robot for various agricultural tasks, aiming to provide a low-cost alternative to work with Mexican farmers in the future. For this edition of the FRE, a navigation algorithm was created using an encoder, an IMU (Inertial Measurement Unit), an RPLIDAR (Rotating Platform Light Detection and Ranging), and cameras to collect data for decision-making. The algorithm was developed in ROS Melodic, dividing the task into steps that were tested to determine the robot's actual movements. The system navigates by using ROIs (regions of interest) and the mass center to guide the robot between maize rows. It calculates the mean of the final orientation values before reaching the end of a row, which is detected using an RPLIDAR. For turns and straight-line movements to reach the next row, the orientation is used as a guide. To detect plants for spraying, lasers located on each side of the vehicle are employed. Obstacle detection relies on a YOLOv5 (You Only Look Once) trained model and a laser, while reverse navigation uses a rear camera. During the competition, the robot faced challenges such as dealing with grass, the small size of the plants, and the need to use a different power source, which affected its performance.

Keywords: machine vision, convolutional neural network (CNN), regions of interest (ROI), autonomous navigation

INTRODUCTION

Corn is an essential crop for farmers and is grown on more than 150 million hectares worldwide (Kannan et al., 2018). Corn is a cereal of great economic and social importance worldwide because corn is critical for human food, industrial use, domestic animal feeding, security of food supply, biodiesel and biofuels, agricultural exports and income, crop rotation, adaptation to climate change, and income sources for farmers (Monteiro et al., 2021).

Currently, agricultural demand is being surpassed by the growing number of inhabitants worldwide, either due to the migration of youth to large cities or the decrease in available land for cultivation. This need has sparked significant interest globally in the development of new technologies and advancements in the field of agricultural robots to aid in achieving better food production.

According to FAO (2009) and Calicioglu et al. (2019) a significant challenge in agriculture is to produce more food because of the increase in the global population, aiming for long-term equilibrium. On the other hand, Subeesh and

Mehta (2021) emphasize that traditional agriculture requires a lot of labor, with limitations in crop monitoring tasks. Additionally, there is a significant decrease in skilled labor, which is why traditional agricultural methods are not sufficient to achieve maximum productivity (Bai et al., 2023). Agricultural robots present an opportunity to strengthen agrifood systems by addressing labor shortages and reduce CO₂ emissions (Orum et al., 2023).

Many agricultural operations require machinery operators to have great skill to achieve good trajectories and configure technical parameters in real-time (Fujita et al., 2020). As an alternative solution, modern agriculture introduces agricultural robots and intelligent equipment, gradually replacing human operations as the direction for future agricultural development (Xie et al., 2022). Interest in agricultural robotic systems has surged in recent years, promoting the development of more autonomous and intelligent vehicles in agriculture.

Autonomous agricultural robots have the potential to increase agricultural production efficiency and reduce the consumption of natural resources (Khadatkar et al., 2021). In

*Correspondence to:

E-mail: nvelazquezl@chapingo.mx

agricultural environments, agricultural robots have been used for various tasks such as plowing, transplanting, pruning, weeding, harvesting, planting, spraying, fertilizing, and more (Mao et al., 2020).

An essential aspect to enable a robot to work autonomously in the field is autonomous navigation. With this goal in mind, research is being conducted in the field of navigation. In these environments, there are sensors capable of detecting crop lines, with the most common ones being known as LIDAR (light detection and ranging), which are laser-based sensors that measures distances, and cameras. Although LIDAR has certain advantages over cameras due to the characteristics of the environments where agricultural robots must navigate (typically open environments with frequent changes in lighting conditions) it is at this point where cameras present disadvantages related to the variability in color shades that can result in different contrast levels (Nehme et al., 2021).

Another group of sensors used for navigation in combination with the above mentioned are integrated in the inertial measurement unit (IMU), which is used to correct position errors (Feng et al., 2023).

For all the reasons mentioned above, Chapingo Autonomous University pioneered the development of a farm robot in Mexico, with the aim of designing, constructing, and evaluating an unmanned mobile vehicle for agricultural tasks. Additionally, in this work, an algorithm using affordable sensors and machine vision is proposed to navigate automatically between maize rows.

MATERIALS AND METHODS

For the development of the vehicle, the following methodology based on the mechatronics design was followed:

- 1) Mechanical design and construction of the vehicle.
- 2) Instrumentation of the vehicle and electrical system.
- 3) Development of the navigation system.
- 4) Design and construction of the sprayer.
- 5) Development of the artificial vision system.
- 6) Functional testing for FRE 2023 tasks.

The main components of the robot are mentioned next.

Mechanical components

For the construction of the multitask agricultural robot "Voltan", 0.125-inch (3.175 mm) aluminum was used for the body and chassis shown in Figure 1 to achieve a lightweight design resulting in lower battery consumption (Reyes and Velázquez, 2019). Steel bushings were placed in the chassis, where 24 ball bearings were installed to position the drive

axles that transmit the motion generated by the motors.

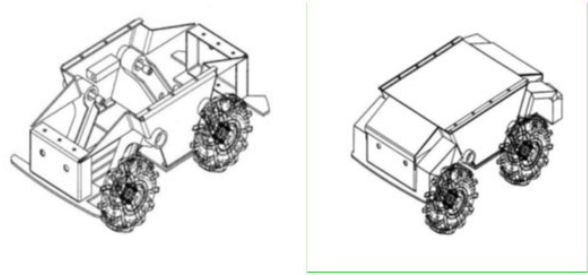


Figure 1: Body and chassis diagrams

The movement of the vehicle is of the skid-steer type, which means it executes turns by adjusting the velocities of the two sides of the robot. To achieve this, each of the two electrical motors is used to control two wheels on each side.

The transmission system consists of a pinion at the top and a sprocket at the bottom, connected by a chain. This design is replicated to drive all 4 tires, and the components have a pitch of 35. The pinions have 9 teeth, while the sprockets have 27 teeth. This configuration generates motion in agricultural tires sized 3.50-4, mounted on rims with a diameter of 4 inches (101.6mm). The components are shown in the Figure 2.

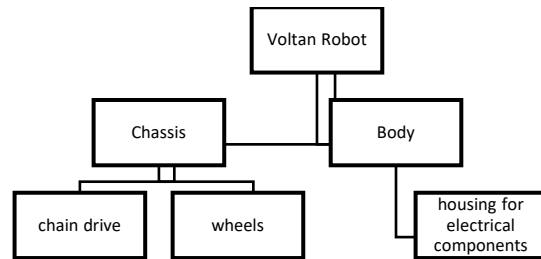


Figure 2: Mechanical diagram

For the connection between the chassis and the wheels, a steel arm was used, constructed with a rectangular tubular profile of 1.75 inches × 0.75 inches (44.45 mm × 19.05 mm) for each wheel.

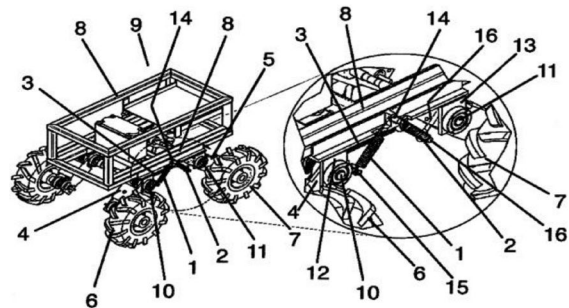


Figure 3: Suspension system

With the aim of ensuring proper performance on uneven terrains, a suspension system was designed and built for ground vehicles shown in Figure 3, whether autonomous or not (Reyes & Velázquez, 2020). This system is characterized by the use of a triangular arrangement consisting of three tension-operated helical springs. Another important component is the tires, which are designed for agricultural use and have a special tread pattern for better grip on the soil.

Electronic components

Electrical system

The electrical power components of the robot were selected with the consideration that the robot would be able to work in tilled soil without excessive skidding while dragging a furrow opener. This electrical system is designed to provide power and control to the DC motors, allowing them to drive the vehicle or machinery to which they are connected. The motor controllers play a crucial role in managing the speed and direction of the motors, while the battery serves as the primary power source. The electrical system in this setup includes the following components:

1. DC Motors: There are two 250 W DC gear reduction motors operating at 12 V each.
2. Sabertooth 2×60 Motor drive. Each motor is equipped with a Sabertooth 2×60 dual motor driver module with a capacity of 60 A. These controllers allow for independent management of motor speed and direction.
3. Battery: The system is powered by a 12 V sealed lead-acid battery from MHB. It has a capacity of 26 Ah, which indicates its energy storage capacity.
4. Wiring: Wiring is used to connect the various components of the electrical system, allowing for the flow of power from the battery to the motors and controllers.
5. Sensors. Cameras, IMU, RPLIDAR are used to obtain information of the environment and the orientation of the robot.

Encoder

The robot was equipped with an encoder model E50S8-5000-3-T-5, which is an optical sensor that emits 5000 pulses per revolution of the wheel axis through an infrared emitter and receiver. The algorithm programming was done in the Arduino IDE, and it calculated the encoder's degrees of advancement (Gr), allowing for the determination of the distance traveled by the robot. To achieve this, the encoder's resolution (5000 pulses per revolution) was used, and, along with the wheel's circumference, the real-time distance

traveled by the robot was determined using the following expressions 1 and 2, respectively.

$$Gr = \left(\frac{360}{ppr} \right) (pr) \quad (1)$$

Where:

- Gr = number of degrees of advancement of the encoder shaft, [°]
- ppr = total number of pulses per revolution of the encoder (5000), [Pulses/revolution]
- pr = pulses recorded by the Arduino, [Pulses/meter]

$$lr = \left(\frac{p}{360} \right) (Gr) \quad (2)$$

This is an incremental encoder, so for better data capture and to prevent pulse loss, the encoder and IMU were connected to an Arduino MEGA, which publishes the readings to ROS on a computer via Serial communication.

RPLIDAR

An RPLIDAR is a Rotating Platform LIDAR which uses this platform to provide a 360-degree scan of the surroundings with laser sensors.

The lidar used for the detection of the plants works knowing the next turn direction, this is written as text in a file using the same coding as the FRE 2023 examples (1R, 3L, etc.), a node of ROS looks for spaces of more than a meter long around the vehicle, so it is subscribed to a topic that publishes the measurements of the encoder, its located in the front part of the robot centered, at a low height of the soil level.

In the robot the RPLIDAR A1 was used, which is a low-cost 2D (360-degree) laser scanner (LIDAR) solution developed by SLAMTEC. The device can perform a 360-degree scan, has dimensions of 98.5 mm × 70 mm × 60 mm and a weight of 170 g. It features a distance range of 0.15 to 6 meters for white objects, and an angular range of 0 to 360 degrees. The distance resolution is less than 0.5 mm, with an angular resolution of 1 degree. The sampling duration is 0.5 ms, and the sampling frequency ranges from 2000 to 2010 Hz. The scanning speed ranges from 1 to 10 Hz, with a typical speed of 5.5 Hz.

This device showed problems with days with high solar illumination outdoors, which is because of the limited lasers in the sensor, better models include multiple rings of lasers which can improve the data obtained.

IMU (Inertial Measurement Unit)

IMUs are essential components for applications requiring precise motion tracking, orientation sensing, and environmental awareness. By combining data from

accelerometers, gyroscopes, and magnetometers, IMU sensors can provide a comprehensive view of an object's 3D motion and orientation in real-time. These sensors are widely used in robotics for navigation and control, in augmented reality for accurate head tracking, in drones for stable flight, and in many other applications where motion and orientation data are critical (Kurniawan, 2021).

Voltan have an IMU MPU6050 which is composed of a 3-axis accelerometer and a 3-axis gyroscope. Together, these sensors can provide the information to determine the heading, pitch, and orientation of an object. An inertial measurement unit (IMU) can be used for measuring acceleration and angular velocity (Cizmic et al., 2023).

The MPU-6050 features three 16-bit analog-to-digital converters (ADCs) for digitizing the gyroscope outputs and three 16-bit ADCs for digitizing the accelerometer outputs. For precision tracking of both fast and slow motions, the parts feature a user-programmable gyroscope full-scale range of ± 250 , ± 500 , ± 1000 , and ± 2000 dps (degrees per second) and a user-programmable accelerometer full-scale range of ± 2 g, ± 4 g, ± 8 g, and ± 16 g.

The main data used from this sensor was the orientation of the robot with respect to the plane of the soil, it was used to calculate turns. To do this the IMU is connected to an Arduino mega that is connected to the computer, the data is published in a ROS topic.

Navigation system

It is important to consider that agricultural autonomous navigation is a complex system engineering, which consists of four key technologies: environmental perception, precise positioning, decision-making and planning, and execution control (Binbin et al., 2023). The Voltan robot from the Universidad Autónoma Chapingo uses for its navigation data from an imu, an encoder, two LIDAR, a RPLIDAR, and cameras, then a program running on a laptop computer make the decision for the correct movements.

Main software in ROS

Voltan operates with code developed in ROS (Robot Operating System) due to its numerous advantages, as noted by Saavedra et al. (2023). These advantages include hardware abstraction, inter-process communication, package management, development tools, distributed computing, software reuse, and rapid testing. ROS enables encapsulating nodes in different programming languages, facilitating system growth and prototyping. The navigation of Voltan is divided into different phases to be performed sequentially.

First the movement between the rows, for this part the machine vision system sends values to adjust the trajectory of the vehicle, at the same time in a different ROS node the

RPLidar combined with encoder data detects the plants and look for spaces without plants to determinate the end of the row and stop the vision system control when its necessary. Second the movement of the final part its guided for the mean of the final 50 orientations of the robot. The third part is the turn, for this part the robot uses the gyroscope to perform 90 degrees turn over its own axis calculated using a mean of the latest orientations when the robot was navigating using the machine vision system. Fourth, it moves to the direction of the next row measuring the displacement with an encoder, once the desired distance is reached the robot stops and performs another 90 degrees turn. Fifth, finally navigates using the gyroscope and the machine vision control start again. All the process can be seen in Figure 4.

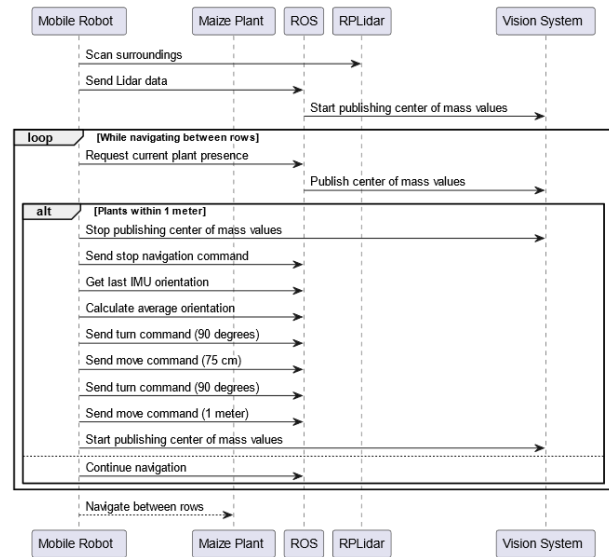


Figure 4: Steps of the navigation

Machine vision system

For the detection of the plants, machine vision was used. The advantages of this approach are the cheap cost of cameras, and the open resources like OpenCV, short for Open Source Computer Vision Library. This is a widely used open-source software library that focuses on computer vision and machine learning.

Recognition and detection of crop rows is one of the key technologies for automatic navigation in the field. Jiang et al. (2015) combined geometric features of crop rows and robot active zones by using several regions of interest (ROI) and extracted crop rows center of mass by clustering methods to obtain crop row centerlines. In a similar way, the Voltan robot use a segmentation algorithm in real time to detect the rows.

To develop this part, the program ignores the color space values that are not important and that includes sky, soil, and other elements. The system developed uses the mass

center and the coordinate in the x axis of the image captured for the camera, for this the camera is positioned in the front and to the center of the robot so this can use the values of the x coordinates in pixels of center of mass of the ROIs to guide its movement.

If the center of mass is allocated to the right or left the robot moves to maintain the point between the rows, it was necessary to limit the vision using masks when the camera is not seeing the rows correctly and this way don't allow other rows enter to the frame of interest. To reduce the compute resources, it was necessary to use a resolution of image smaller than the original raw information of the camera (640 × 480 pixels). The system has been developed to be useful even in different light conditions present in the outside on different time hours of day. For the code OpenCV was used along with ROS Melodic.

Movement algorithm

All the sensors and actuators are communicated to ROS Melodic in a laptop, which do the control and make decisions. All the sensors publish its data in ROS topics. Other node is used for the detection of the end of the row, this use information from the RPLIDAR and encoder, and when there is a space of more than a meter publish an "end of the row" message to stop the navigation using machine vision.

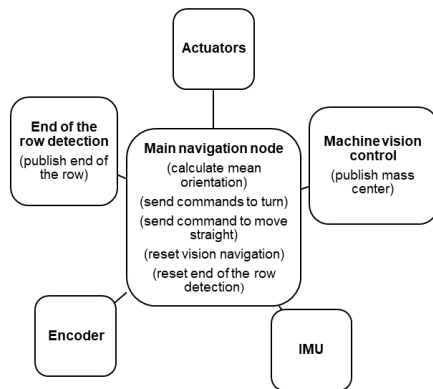


Figure 5: Nodes used for the navigation of Voltan

The main nodes shown in Figure 5 are:

1. Navigation node. This node follows the sequence used to navigate in the field, calculates orientation of the robot for the turns and straight movements, and reset other nodes when they're not necessary. It also reads the route of a text in a file with instructions for the next turn and next number of rows to be traverse.
2. End of the row detection. Detects the areas with no plants and determines if the distance of the row has been covered and the side of the next turn have the space for the turns.

3. Machine vision control. Segments the plants, calculates the mass center and publish the coordinates of the point to guide the robot.
4. Encoder. It's connected to an Arduino that publish the data to ROS.
5. IMU. It is connected to an Arduino that publish the data to ROS.
6. RPLIDAR. It is connected to the computer and publish readings of the laser to ROS.

Sprinkler System

For the localized sprinkler system, the robot was equipped with two Tf-Lidar Plus sensors, with one sensor placed on each rear side of the robot. The algorithm was programmed using Arduino IDE. Only one sensor can be read per microcontroller, for this reason, a slave microcontroller was used for each sensor, and the reading from each sensor was sent via serial communication to a master microcontroller where decision-making was performed.

The algorithm of the slave microcontrollers was programmed so that when an object is at a distance equal to or less than 40 cm, a signal is sent to the microcontroller equal to 1; otherwise, a value equal to 0 is sent. In the master microcontroller, the forward distance of the vehicle was measured, and it also received signals from the two slaves.

The algorithm is divided into two routines, with the first routine used when the robot enters between plants rows and is executed once.

The slave Arduinos have the function of monitoring the sides of the robot for the presence or absence of plants. The Lidar sensors detect objects once the vehicle begins to navigate. When an object appears at a distance equal to or less than 40 cm, a status value of 1 is sent via serial communication. Conversely, if there is no object within that range, a value of 0 is sent.

Using the data received by the master Arduino, actions are determined to activate or deactivate relays that control the sprinklers. Before executing any action, the flag value is checked to determine whether it is in the rows entry routine or already within it.

If both Lidar sensors detect the presence of an object on both sides of the sprinkler system and the flag has a value of 1, a 5-second delay is executed. This delay compensates for the time difference since the sensors and sprinklers are not located in the same position. With this compensation, there is time for the sprinkler to reach the plant that the sensor has monitored.

The activation or deactivation of the sprinklers depends on the detection or non-detection by the Lidar sensors. The sprinkler turns on 2 seconds before reaching the plant and continues spraying for an additional 2 seconds. When this

routine is completed, it indicates that the robot is now between rows, so the flag value is updated to 2. When the flag takes on this value, the algorithm begins executing the routine in which the delay is 3 seconds, and this routine continues throughout the navigation.

Detection of images system

To detect the obstacles represented for images of 3 categories: human, deer, and another category selected for every team in the competition a convolutional neural network architecture was used, YOLO (You Only Look Once) V5, which is a state-of-the-art, real-time object detection system that frame object detection as a single regression problem, straight from image pixels to bounding box coordinates and class probabilities (Redmon and Farhadi, 2016).

Object detection involves creating features from input images. These features are then fed through a prediction system to draw boxes around objects and predict their classes. The YOLO model was the first object detector to connect the procedure of predicting bounding boxes with class labels in an end-to-end differentiable network.

The YOLO network consists of three main pieces. Backbone: A convolutional neural network that aggregates and forms image features at different granularities. Neck: A series of layers to mix and combine image features to pass them forward to prediction. Head: Consumes features from the neck and takes box and class prediction steps (Solawetz, 2020).

For start the training is mandatory to have a dataset of images and label the dataset. The dataset was labeled using (Makesense, 2023), a free-to-use online tool for labeling photos that do not require installation, it makes suggestions and automate repetitive parts in the labelling process. During this task it was marked into a box the object of interest and put a label with the category of this object.

For the task, 'four categories were used: human, deer, goat and rooster, a dataset of images with these objects was created for the training of the neural network, the size of this is 1085 images between the three categories, each photo contains the corresponding label for the object that contains.

For the validation set, 84 images were set aside with this three categories and objects that no correspond to any of the four categories. The image processing was done in Google Collab, using and preexisting model of YOLO v5 programed in Python 3. A 100-epoch training process was established.

When the training was completed, a file containing the corresponding values for each category was obtained and, then this file was imported into a python program where is

possible to obtain data from a camera and make the boxes to point the objects that are part of one of the categories of interest.

When an object appears in the image, the software recognizes it and try to classify in one of the categories of interest, if the object is part of them, it appears into a box with a text with the name of the category to which it belongs. During the task, a Logitech C920 web camera, connected to a portable computer mounted on the robot, was used, and the signals of object detection were the labels and the boxes around each object as can be seen in Figure 6.



Figure 6: Detection of a goat using the code developed

RESULTS

This section includes a description of the performance of the robot Voltan in the Field Robot Event 2023 competition in the 4 different tasks but also includes data from testing of the steps used for the navigation in the simulated maize field.

Tests

Turns using the IMU

One of the steps in the navigation system involves making 90-degree turns. To test this aspect, actual turning angles were measured on a solid floor using a metal square and a protractor. The wheels on one side of the robot were aligned with reference marks, and the robot was then rotated using the data of the IMU to stop after 90 degrees. The actual degrees of rotation were recorded. This process was repeated 20 times for both left and right turns. The results are shown in the Table 1.

Table 1: Angles measured for turns to the left of the robot

Repetition	Actual angle	Sensor final angle	Angle objective	Sensor angle	Error
1	93	90.14	90	90.14	2.86
2	96	180.15	180.14	90.01	5.99
3	93	270.22	270.15	90.07	2.93
4	94	360.29	360.22	90.07	3.93
5	96	450.37	450.29	90.08	5.92
6	95	540.39	540.37	90.02	4.98
7	95	630.47	630.39	90.08	4.92
8	94	720.55	720.47	90.08	3.92
9	94	810.57	810.55	90.02	3.98
10	95	900.61	900.57	90.04	4.96
11	96	990.66	990.61	90.05	5.95
12	95	1080.78	1080.66	90.12	4.88
13	95	1170.87	1170.78	90.09	4.91
14	94	1260.89	1260.87	90.02	3.98
15	96	1351	1530.89	90.11	5.89
16	95	1441.01	1441	90.01	4.99
17	95	1531.1	1531.01	90.09	4.91
18	94	1621.22	1621.1	90.12	3.88
19	96	1711.32	1711.22	90.1	5.9
20	96	1801.41	1801.32	90.09	5.91
Mean	94.85			90.0705	
SD	0.98808693			0.039666372	
MAE					4.7795

SD – standard deviation; MAE – mean absolute error

Table 2: Angles measured for turns to the right of the robot

Repetition	Actual angle	Sensor final angle	Angle objective	Sensor angle	Error
1	92	90.03	90	90.03	1.97
2	93	180.11	180.03	90.08	2.92
3	92	270.12	270.11	90.01	1.99
4	93	360.26	360.12	90.14	2.86
5	92	450.31	450.26	90.05	1.95
6	92	540.33	540.31	90.02	1.98
7	93	630.52	630.33	90.19	2.81
8	92	720.62	720.52	90.1	1.9
9	93	810.71	810.62	90.09	2.91
10	93	900.72	900.71	90.01	2.99
11	92	990.78	990.72	90.06	1.94
12	92	1080.82	1080.78	90.04	1.96
13	93	1170.91	1170.82	90.09	2.91
14	94	1260.99	1260.91	90.08	3.92
15	92	1351.05	1350.99	90.06	1.94
16	91	1441.12	1441.05	90.07	0.93
17	93	1531.21	1531.12	90.09	2.91
18	92	1621.29	1621.21	90.08	1.92
19	92	1711.33	1711.29	90.04	1.96
20	94	1801.37	1801.33	90.04	3.96
Mean	92.5			90.0685	
SD	0.76088591			0.04368247	
MAE					2.4315

SD – standard deviation; MAE – mean absolute error

It can be inferred that the error is between 2 and 6 degrees more than the turn desired, so it was taken in consideration to calculate an objective angle more adequately. Using the mean average error, an objective angle of 85 degrees was included in the code for turns to the left. For the measurements recorded for the turns to the right the data is shown in the table 2.

It can be inferred that the error is between 0.9 and 4 degrees more than the turn desired, so it was taken in consideration to calculate an objective angle more adequately. Using the mean average error, an objective angle of 87 degrees was included in the code for turns to the left.

Test of displacement from row to row

To test the code for straight movement at the end of the row the movements were measured at the end of the row and until the robot is in the new row to restart the navigation using machine vision. For the robot to finish in a good final position its necessary that the calculation of the orientation angle mean that is in the final part of the row is done well, because this angle is the reference to finish in a parallel position in a new row. This part was divided into 5 movements as can be seen in the Figure 7.

To know the actual displacement after every turn at the end of the row a model was constructed using green paper rectangles that simulated the plants, attached to a 5-meter string with a 40 cm spacing between them. The robot had the objective to navigate using the machine vision system for 5 meters and then start the turns routine, first a displacement of 0.6 meters between rows was used.

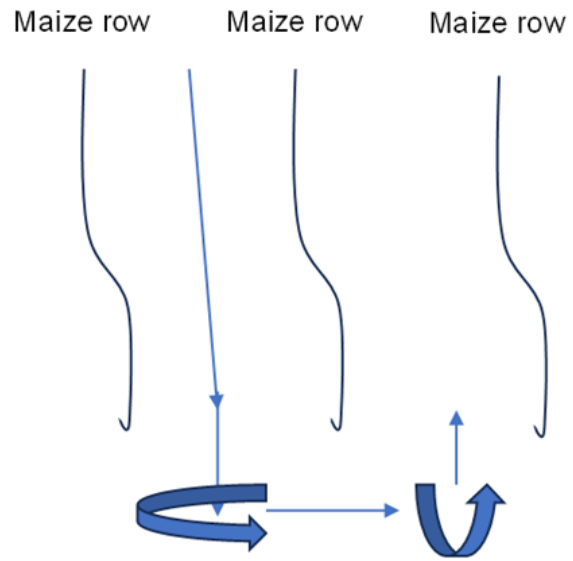


Figure 7: Movements of the robot to reach the next row

From these results the mean distance the robot moves after the turns of 90 degrees is 0.305 m for the first one and 0.262 for the second one as can be seen in table 3. These values were used to set the necessary straight displacement to move to the next rows. This experiment was repeated with a different distance for the sd2 the results are showed in the table 4.

The same was repeated a third time but just considering the last 5 movements the results are in the Table 5.

Table 3: Robot displacements measured in motion

Repetition	dbr	sd1	dat1	sd2	dat2	sd3
1	5.5	1.06	0.33	0.61	nd	0.52
2	5.2	1.01	0.28	0.57	0.34	0.5
3	5.14	1.01	0.4	0.68	0.26	0.63
4	4.98	1.05	0.27	0.6	0.25	0.55
5	4.3	1.08	0.28	0.59	0.23	0.57
6	4.64	1.1	0.27	0.56	0.23	0.55
Mean	4.96	1.051667	0.305	0.601667	0.262	0.553333
SD	0.391578	0.033375	0.04717	0.038909	0.040694	0.041096

dbr – displacement between rows, sd – straight displacement, dat – displacement after turn, SD – Standard deviation, nd – no data

Table 4: Robot displacements measured in motion

Repetition	dbr	dat1	sd2	dat2	sd3
1	5.03	0.47	1.6	0.38	0.5
2	5.3	0.38	1.49	0.34	0.49
3	5.27	0.32	1.69	0.27	0.5
4	4.85	nd	1.56	0.35	0.5
5	4.56	0.45	1.63	0.4	0.6
6	5.25	0.29	1.46	0.41	0.51
7	4.62	0.42	1.54	nd	0.53
Mean	4.982857	0.388333	1.567143	0.358333333	0.518571
SD	0.312181	0.358608	0.071949	0.07994	0.051153

dbr – displacement between rows, sd – straight displacement, dat – displacement after turn, SD – standard deviation, nd – no data

Table 5: Robot displacements measured in motion

Repetition	dat1	sd2	dat2	sd3
1	0.22	0.55	0.37	0.51
2	0.28	0.48	0.3	0.53
3	0.34	0.6	0.33	0.57
4	0.37	0.53	0.23	0.52
5	0.28	0.54	0.4	0.5
Mean	0.298	0.54	0.326	0.526
SD	0.058481	0.043012	0.065803	0.027019

sd – straight displacement, dat – displacement after turn, SD – standard deviation

Using the total number of dat1 and dat2 the mean displacements were obtained, shown in Table 6.

Table 6: Mean of the distances after turn 1 and 2

	dat1	dat2
Mean	0.33235294	0.318125
SD	0.07163531	0.06554579

dat – displacement after turn, SD – standard deviation

This is a better data of the real movements of the robot.

Test of detection of plants

Table 7: Results of the detection with lasers

Number of papers	Papers sprinkled	Success rate
36	36	100 %

The sprinkler made for the competition can be seen in Figure 8. In this test the same model using paper to represent the plants was used, it was annotated the plants where water was applied. The distance between the lines with plants was

Table 8: Metrics of the trained YOLOv5 model

Class	Images	Instances	P	R	mAP50	mAP50-95
All	84	93	0.46	0.49	0.523	0.352
Person	84	27	0.967	1	0.995	0.682
Goat	84	21	0.0231	0.022	0.187	0.143
Deere	84	16	0.152	0.25	0.207	0.142
roaster	84	29	0.696	0.69	0.702	0.442

P – precision, R – recall, mAP50 – mean average precision calculated with an IoU (intersection over union) threshold of 50%, mAP50-95 – mean average precision calculated with an IoU (intersection over union) threshold range from 50% to 95%

Results on the Field Robot Event 2023 competition

Task 1: navigation

For this task, the robot had problems with the size of the plants, which were too small for the camera and RPLIDAR to detect using the positions of the sensors and the programmed codes, so the team changed the sensors' location. Another challenge was the presence of grass in the

of 0.8 m. In the table 7 it can be seen that the system detected all the papers while the robot was navigating.

As it can be seen the lasers worked fine with a detection distance of 0.4 m.

**Figure 8:** Voltan with the sprinkler for the competition

Metrics of CNN

The results after training the YOLOv5 network are shown in Table 8. These results highlight the performance metrics and effectiveness of the model on the test dataset.

The results show that the CNN have problems with goats and deer, with a low precision, the person can be detected better than the rest of the classes and the roaster was a better option to propose for class. Well, these issues come from the low number of images used to train the network.

field, which created noise during the segmentation of the plant rows. The batteries had to be bought in Europe due to the restriction of the airline for this kind of devices.

At the moment of the task, the robot had problems following the curved sections of the rows because of the noise in the segmentation system and the different power source, causing it to deviate from the track. The final position for the task was the 11th place.

Task 2: treating (spraying) the plants

To address the issue with the presence of grass and trees, the vision of the camera was restricted by using masks in the OpenCV code, it only utilized data from the camera's lower left and right sections. But the use of just one battery changed the power provided for the driver to the motors causing problems to the movement and correction of the position.

During the test of task 2, the robot could not enter the field to initiate navigation due to low values of PWM and battery power. As a result, the final outcome was recorded as 'DNS' (Did Not Start).

Task 3: sensing and recognizing possible obstacles.

For this task, the YOLOv5 model, trained as mentioned in the materials and methods section, was used. The system encountered issues with the size of the images, making it unable to detect image categories at a distance. Consequently, the robot achieved the 7th position.

Task 4: static and dynamic obstacles.

For this task, a backward navigation camera was added. It operates by using an inverted version of the main machine vision system, allowing the robot to reverse when encountering obstacles and then enter the next row. Additionally, a laser was integrated to detect obstacles in the field within a distance of 0.4 m, halting the robot's movement for camera-based detection.

Despite these enhancements, the robot continued to experience issues related to low PWM values and battery power, which affected the accuracy of its movements. As a result, the robot finished in the 8th position.

CONCLUSION

The machine vision system for navigation is a good cheap alternative to the use instead of RPLIDAR, but it needs a more powerful computer. The inclusion of the IMU to guide the straight movements works fine. The navigation in the real test was poor and posed challenges to consider for future editions of the competition and work in a real agricultural field. The use of masks to limit the frame was a good addition to the system. The measurements of the real displacements show a possible loss of data of the encoder. The system, as it currently stands, is not reliable and requires several improvements. Firstly, it needs at least one more encoder and an updated version of ROS (Robot Operating System). The algorithms should be enhanced to combine data from multiple sensors (RPLIDAR, camera, and encoder) for detecting the end of the row and navigating between plants. Additionally, the system should calculate

the speed of movement, make necessary corrections, and determine current position values. Other improvements to be considered include adding an embedded computer, calculating position using odometry, implementing fuzzy logic control for movement.

Acknowledgements

We would like to express our sincere gratitude to the Universidad Autónoma Chapingo for the financial support that enabled our participation in the Field Robot Event 2023. This support allowed us to showcase our work and compete at an international level. We deeply appreciate the university's commitment to fostering innovation and supporting student projects.

REFERENCES

1. Bai, Y., Zhang, B., Xu, N., Zhou, J., Shi, J., & Diao, Z. (2023). Vision-based navigation and guidance for agricultural autonomous vehicles and robots: A review. *Computers and Electronics in Agriculture*, 205, 107584.
2. Calicioglu, O., Flammini, A., Bracco, S., Bellù, L., & Sims, R. (2019). The future challenges of food and Agriculture: An Integrated analysis of trends and solutions. *Sustainability*, 11(1), 222. Retrieved from: <https://doi.org/10.3390/su11010222>
3. Cizmic, D., Hoelbling, D., Baranyi, R., Breiteneder, R., & Grechenig, T. (2023). Smart boxing glove "RD a": IMU combined with force sensor for highly accurate technique and target recognition using machine learning. *Applied Sciences*, 13(16), 1-16. Retrieved from: <https://doi.org/10.3390/app13169073>
4. Food and Agriculture Organization (FAO). (2009). *How to feed the world in 2050*. Retrieved from: https://www.fao.org/fileadmin/templates/wsfs/docs/Issues_papers/Issues_papers_SP/La_agricultura_mundial.pdf
5. Feng, X., Liang, W. J., Chen, H. Z., Liu, X. Y., & Yan, F. (2023). Autonomous localization and navigation for agricultural robots in greenhouse. *Wireless Personal Communications*, 131, 2039-2053. Retrieved from: <https://doi.org/10.1007/s11277-023-10531-z>
6. Fujita, S., Emaru, T., Ravankar, A. A., & Kobayashi, Y. (2020). Development of robust ridge detection method and control system for autonomous navigation of mobile robot in agricultural farm. In Symposium on Robot Design, Dynamics and Control (pp. 16-23). Cham: Springer International Publishing.
7. Jiang, G., Wang, Z., & Liu, H. (2015). Automatic detection of crop rows based on multi-ROIs. *Expert Systems with Applications*, 42(5), 2429-2441. Retrieved from: <https://doi.org/10.1016/j.eswa.2014.10.033>
8. Khadatkar, A., Mathur, S. M., Dubey, K., & BhusanaBabu, V. (2021). Development of embedded automatic transplanting

- system in seedling transplanters for precision agriculture. *Artificial Intelligence in Agriculture*, 5, 175-184. Retrieved from: <https://doi.org/10.1016/j.aiia.2021.08.001>
9. Kannan, M., Ismail, I., & Bunawan, H. (2018). Maize dwarf mosaic virus: From genome to disease management. *Viruses*, 10(9), 492. Retrieved from: <https://doi.org/10.3390/v10090492>
10. Kurniawan, A. (2021). IMU sensor: Accelerometer and gyroscope. In: Beginning Arduino Nano 33 IoT. Apress, Berkeley, CA. Retrieved from: https://doi.org/10.1007/978-1-4842-6446-1_3
11. Makesense (2023). Makesense for Labeling [Software platform]. Retrieved from: <https://www.makesense.ai/>
12. Mao, S., Li, Y., Ma, Y., Zhang, B., Zhou, J., & Wang, K. (2020). Automatic cucumber recognition algorithm for harvesting robots in the natural environment using deep learning and multi-feature fusion. *Computers and Electronics in Agriculture*, 170, 105254. Retrieved from: <https://doi.org/10.1016/j.compag.2020.105254>
13. Monteiro, N., Alencar, E., Souza, N., & Leao, T. (2021). Ozonized water in the preconditioning of corn seeds: physiological quality and field performance. *Ozone Science and Engineering*, 43(5), 436-450. Retrieved from: <https://doi.org/10.1080/01919512.2020.1836472>
14. Nehme, H., Aubry, C., Solatges, T., Savatier, X., Rossi, R., & Boutteau, R. (2021). Lidar-based structure tracking for agricultural robots: Application to autonomous navigation in vineyards. *Journal of Intelligent & Robotic Systems*, 103, 1-16. Retrieved from: <https://doi.org/10.1007/s10846-021-01519-7>
15. Orum, J., Wubale, T., Marcus, S., Harold, A., Veldhuisen, B., & Hildrands, H. (2023). Optimal use of agricultural robot in arable crop rotation: A case study from the Netherlands. *Smart Agricultural Technology*, 5, 1-8. Retrieved from: <https://doi.org/10.1016/j.atech.2023.100261>
16. Redmon, J., & Farhadi, A. (2016). YOLO9000: Better, faster, stronger. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Retrieved from: <https://arxiv.org/abs/1612.08242>
17. Reyes-Amador, A., & Velázquez-López, N. (2019). Modelo industrial de carrocería para vehículo de cuatro ruedas (No. de solicitud MX/f/2018/003352). IMPI. Retrieved from: <https://siga.impi.gob.mx/>
18. Reyes-Amador, A., & Velázquez-López, N. (2020). Sistema de suspensión para vehículos terrestres autónomos o no autónomos (Patente No. MX 4369 B). IMPI. Retrieved from: <https://siga.impi.gob.mx/>
19. Saavedra Sueldo, C., Perez Colo, I., De Paula, M., Villar, S. A., & Acosta, G. G. (2023). ROS-based architecture for fast digital twin development of smart manufacturing robotized systems. *Annals of Operations Research*, 322(1), 75-99. Retrieved from: <https://doi.org/10.1007/s10479-022-04759-4>
20. Solawetz, J. (2020). What is YOLOv5? A Guide for Beginners. Retrieved from: <https://blog.roboflow.com/yolov5-improvements-and-evaluation/>
21. Subeesh, A., & Mehta, C. R. (2021). Automation and digitization of agriculture using artificial intelligence and internet of things. *Artificial Intelligence in Agriculture*, 5, 278-291. Retrieved from: <https://doi.org/10.1016/j.aiia.2021.11.004>
22. Xie, D., Chen, L., Liu, L., Chen, L., & Wang, H. (2022). Actuators and sensors for application in agricultural robots: A review. *Machines*, 10(10), 913. Retrieved from: <https://doi.org/10.3390/machines10100913>

Robot za navigacijo v posevkih koruze za dogodek »Field Robot 2023«

IZVLEČEK

Operacije, kot je avtonomna navigacija robotov med vrstami rastlin na koruznem polju, so ključne za razvoj robotov v kmetijstvu. Takšne operacije so lahko del številnih nalog, kot so škropljenje, spremljanje rasti in zdravja rastlin ter odkrivanje plevela in škodljivcev. Na dogodku »Field Robot Event 2023« (FRE) so univerze in raziskovalne skupine izzvane k razvoju naprednih algoritmov za kmetijske robote. Universidad Autónoma Chapingo razvija robota za različna kmetijska opravila, s ciljem zagotoviti cenovno dostopno rešitev za mehiške kmete v prihodnosti. Za dogodek FRE so ustvarili navigacijski algoritem, ki uporablja podatke iz odometrije, inercialne merilne enote (IMU), RPLIDAR (nizkocenovno LiDARsko tipalo) in kamer, kar omogoča avtonomno odločanje. Algoritem je bil razvit v Robotskem Operacijskem Sistemu (ROS Melodic) in je nalogo razdelil na več korakov, ki so bili preizkušeni za določitev dejanskih premikov robota. Navigacijski sistem upošteva interesna področja (ROI) in masno središče robota, kar omogoča krmiljenje robota med vrstami koruze. Za premikanje med vrstami uporablja meritve RPLIDAR, medtem ko za zavoje uporablja orientacijo robota prek IMU. Za zaznavanje rastlin za škropljenje so na vsaki strani vozila nameščeni laserski merilniki. Zaznavanje ovir temelji na algoritmu YOLOv5 (You Only Look Once) in laserju, medtem ko za vzvratno navigacijo robot uporablja zadnjo kamero. Med tekmovanjem se je robot soočal z izzivi, kot so ravnanje s travo, majhne rastline in potrebe po drugačnih energetskih virih, kar je vplivalo na njegovo delovanje.

Ključne besede: strojni vid, konvolucijske nevronske mreže (CNN), interesna področja (ROI), avtonomna navigacija