JOURNAL OF PRODUCTION AND AUTOMATION

v.2,n.1,p.19-31, 2019.

Development of an autonomous vehicle at a 1:8 scale

Ariel Guerrero[1], Micaela Jara[2], Erid Pacheco[2], Ariel Bogado[2], Jesús Franco[2]

[1]Parque Tecnológico de Itaipu – Paraguay

[2]Universidad Católica "Nuestra Señora de la Asunción"

ABSTRACT

The development of an autonomous vehicle is the subject of extensive study by many researchers [1]. One of the approaches of study consists of the use of physical models at scale whose construction is the object of this article. For this purpose, a RC (remotely controlled) electric car with a 1: 8 scale is modified. A myRIO is used for data acquisition (odometer, accelerometer, magnetometer, gyroscope) and operation tasks. Another task is to merge the data of the sensors in such a way that the position and orientation information can be used for the navigation decision making. This data can be sent to a workstation, running an application in LabVIEW. This workstation provides a human-machine interface for the operator to display the navigation parameters.

Keywords: Autonomous Vehicle, IMU, Navigation Algorithm, Kalman.

INTRODUCTION

Autonomous vehicles have attracted a great deal of interest in research in recent years, as well as important industry development efforts. In 2007, DARPA has executed the Urban Grand Challenge [2], with entries from several universities documented in many different publications, for example [3, 4].

Many companies in the automotive sector have their own division dedicated to the investigation of autonomous vehicles and recently other companies that are not of the sector such as NVIDIA, APPLE, GOOGLE, YANDEX, BIADU have started to develop their own autonomous vehicle. Projects that have been widely reported in the news [5, 6]. Although important efforts have been made in this field, there are still many problems to be solved, among them, the problems of detection, the different types and levels of control and the interaction of autonomous vehicles with their environment.

An autonomous vehicle is a vehicle capable of imitating the human capacities of management and control. The driver may choose the destination but is not required to activate any mechanical operation of the vehicle. Autonomous vehicles perceive the environment through sensors such as laser, radar, lidar, global positioning system and computer vision. Advanced control systems interpret information to identify the appropriate route, as well as obstacles and relevant signage. Autonomous vehicles are generally capable to travel previously programmed roads and require a cartographic reproduction of the terrain, so if a route is not picked up by the system it's possible that it cannot advance coherently and normally.

v.2,n.1,p.19-31, 2019.

METHODOLOGY

The methodology used in this project was the traditional design. In figure 1, the flow diagram of the tasks performed is presented. The main problem was how to develop, in a short period of time, an autonomous scaled electric car. Based on this approach, possible solutions were analyzed, carrying out the necessary studies and evaluations. Once the objectives to be met were set, the hardware was prepared, in this case the adaptation of the components in the scaled electric car, and the design of the software. The software and hardware implementation were carried out. Subsequently, the necessary tests were carried out in order to verify problems, and if necessary, implement modifications and improvements to the system, whether they were in the software or hardware.

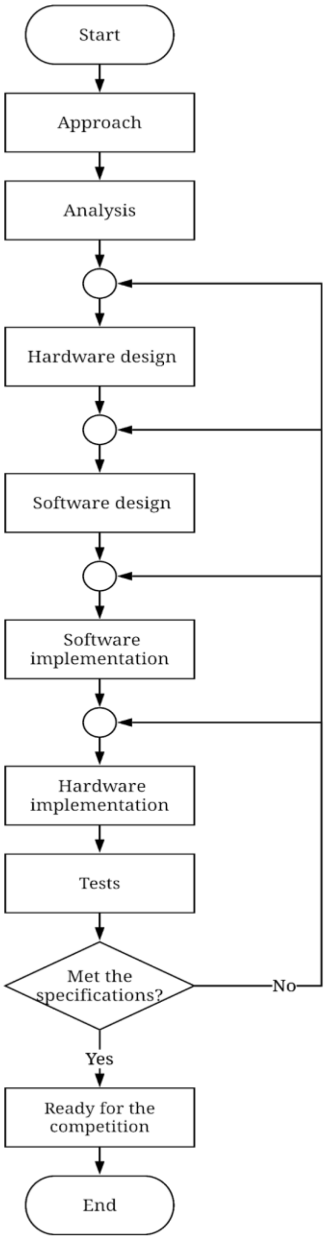

Figure 1. Project methodology flowchart.

DEVELOPMENT

System Hardware Architecture

Following the consulted literature [7] we needed to have a mathematical model that describes the behavior of the plant. As it would take more time to reach an accurate model taking into account the short time available to complete the project and the fact that we needed to have the control of both the traction and the direction of the plant, we proceeded to obtain the mathematical model through a transfer function using the "black box" method, which consists in the study of an element from its output behavior for a given entry without covering its internal functioning. The diagram can be seen in figure 1. Knowing the input and output, the transfer function of the "black box" was identified using the Matlab System Identification tool.

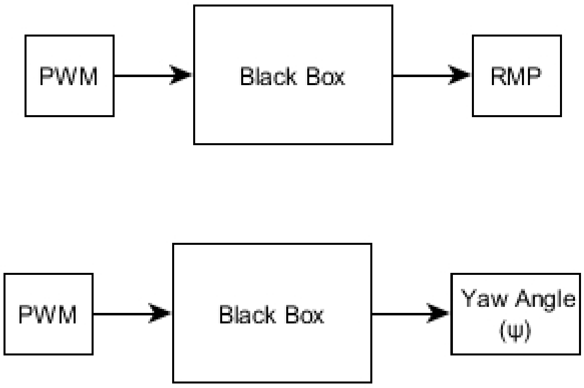

Figure 2. Powertrain and direction schematic

In the "black box" of the powertrain, there is a brushless motor (BLDC) powered by an electronic speed controller (ESC), which receives a PWM signal to control the speed of the motor. On the other hand, in the "black box" of the steering, there is a servo motor, which receives another PWM signal to control the Yaw Angle of the front wheels, and consequently control the direction of the car.
The result obtained through this method had an approximation of 70% for the powertrain and 87% for the direction, which was enough to be able to control the plant.

The system to be implemented consists of:

- Autonomous Vehicle at 1: 8 scale: Modified chassis of a miniaturized electric car in which a myRIO 1900 is adapted as well as the sensors and actuators required for inertial navigation (IMU + GPS). In this platform the algorithms that allow determining the position and orientation information based on the sensors will be executed. It will also allow the acquisition of data from the sensors, execute the selected navigation algorithm and determine the control signals to the actuators.
- Base Station: For configuration of the navigation parameters (waypoints) and visualization of the status of the sensors.

The hardware platform consists of the following elements:

v.2,n.1,p.19-31, 2019.

- Crius Crius AIOP v2.1: this electronic board type MARG (Magnetic, Angular rate and Gravity) has several built-in sensors such as a gyroscope /accelerometer MPU6050 6-axis, a high-precision altimeter MS5611-01BA01, and a magnetometer HMC5883L 3-axis. The integrated microcontroller is an ATMEGA 2560 8-bit, 16 MHz and communicates with external devices through pins and serial ports.
- Sensor: MPU6050 : Accelerometer and gyroscope

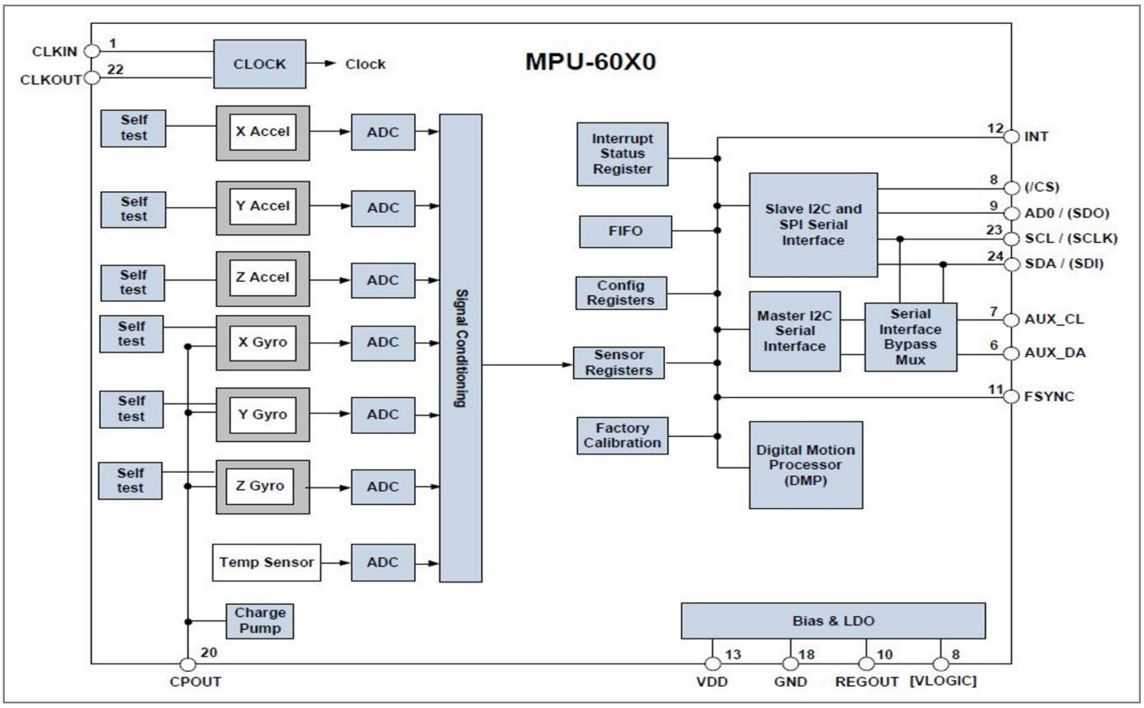

Figure 3. Architecture of the MPU 6050 [8]

Figure 3 shows the architecture of the MPU6050 [9], the sensor has digital analog converters for each of the axes and thus obtains the values simultaneously with a range of up to ± 2000° per second in the case of the gyroscope and ± 16g for the accelerometer. Then, the data is filtered according to the preset configuration and the factory calibration, to then go to the sensor register, the data can be accessed by the DMP or by the user. The DMP updates the FIFO data read at a certain frequency in order to avoid overflow. The communication serial interface of the MPU6050 is the I2C communication protocol.

- Sensor: HMCL5883L: Integrated magnetometer: the magnetometer integrated in the controller is the Honeywell HMC5883L, the triaxial sensor has an operation field of -8 to +8 gauss, which in this way gives the real direction of the geographic north considering the 15-degree inclination that exists in the zone. The serial interface of the HMC5883L complies with the I2C communication protocol at 400 kHz.
- Odometer and Speed Sensor Encoder FC03:

Operating voltage: 3.3V - 5V DC

Outputs: Analogica y Digital TTL

Sensor: MOCH22A

Board model: FC-03/FZ0888

Type of emitter: Photodiode IR

Detector type: Phototransistor

Wavelength of the emitter: 950 nm (infrared)

Weight: 8 g

Dimensions: 32*14*7 mm

Slot: 5 mm

Opamp comparator: LMS393

Power indicator LED

Pulse indicator LED

Output TTL ON: blocked sensor

Output TTL OFF: unlocked sensor

Knowing the position or speed of an engine is very important in robotics, for which there are several alternatives, one of the most common being the use of optical type encoders. The incremental optical encoders perform the measurement of movement with the use of an infrared beam that is interrupted by the slots of a disk coupled to the shaft. The number of slots per revolution will determine the encoder's accuracy, in this case 4 pulses per revolution.

- Actuator: Digital Servo HB-5514 14kg

Figure 4. Digital servo for direction.

It is an actuator device that has the ability to be located in any position within its operating range, and to remain stable in that position.

v.2,n.1,p.19-31, 2019.

- Actuator: Motor BLDC 2200KV (rmp/V): they are synchronous motors fed by DC through an inverter or switching power supply that produces an AC electric current to control each phase of the motor through a closed-circuit controller. The controller provides pulses of current to the motor windings that control the speed and torque of the motor.
- Actuator: Electronic Speed Controller ESC WP-8BL100, 100A: an electronic speed controller or ESC is an electronic circuit that controls and regulates the speed of an electric motor. It can also provide reversing of the motor and dynamic braking. ESCs are often used on motors essentially providing an electronically-generated three-phase electric power low voltage source of energy for the motor.
- Mechanical Plant: Chasis Haboo Hyper VS 1/8:

Figure 5. Chassis.

Dimension: 460 mm. x 306 mm. x 140 mm.

Distance between axis: 322 mm.

Weight: 4720 g

Battery for myRIO: LI-PO 2S 3000 mAh 7.4 v

Motor battery: LI-PO 4S 5400 mAh 14.8 v

Anodized aluminum chassis.

Front aluminum turret 4mm and rear 3mm
Reinforced suspension support

Battery holder with velcro

Big Bore 17mm shock absorbers

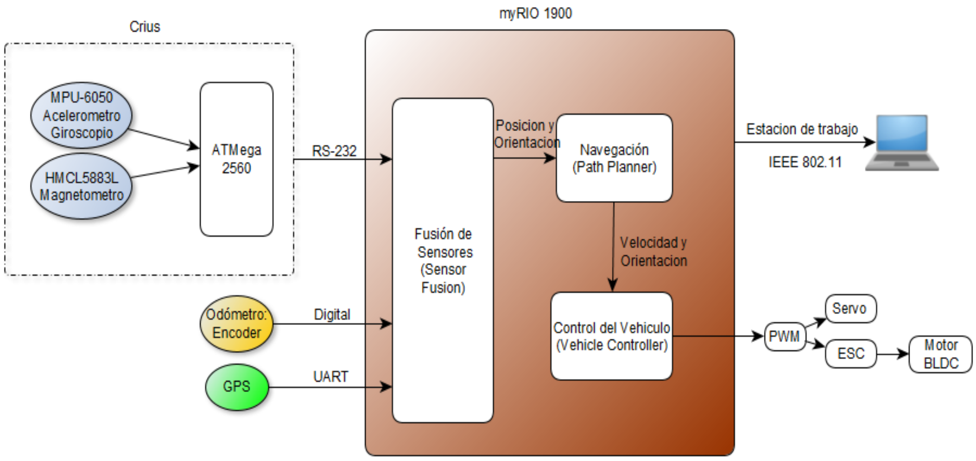

Figure 6. Hardware Architecture.

Software Architecture

In this section, we will focus more on the processing of sensor data to obtain the position and orientation data, and the use of them for the calculation of the path of travel. Before the fusion of sensors, the collected data go through a calibration process, where offset and gain errors are eliminated, this process can be found in [8], then the data goes through a process of changing the reference, passing from a fixed frame of reference to the autonomous vehicle, to which the inertial sensors belong, to a frame of reference fixed to the ground, known as the navigation reference frame.

- Sensor Fusion

Sensor signals were used to improve and correct the position measurement of the 4-wheel autonomous vehicle to obtain a more reliable position estimate. From this, we calculated the estimation of the position and reduced the systematic and non-systematic errors during the tests and we succeeded in estimating the deviation of the turn bias. The basic tool here is a Kalman filter.

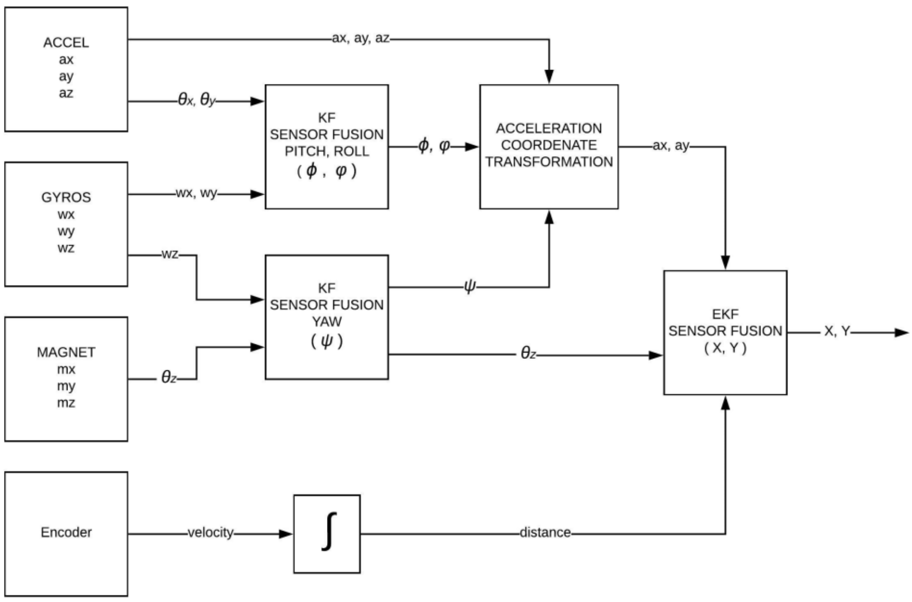

Figure 7. Overview of the fusion system.

v.2,n.1,p.19-31, 2019.

As shown in figure 7; Initially, a linear Kalman filter (KF) is used to merge the data of the accelerometer and the gyroscope, with this we obtain the pitch (φ) and roll (φ) angles, the angle yaw (ψ) is obtained from the fusion of the gyroscope with the magnetometer, also through a linear Kalman filter (KF) [14].

In figure 8, the result of the fusion of the gyroscope and the magnetometer is shown. In it, it can be seen that the magnetometer does not have good response at high frequencies, however, at low frequency, the response is good. On the other hand, the behavior of the gyroscope curve is smooth, but the cumulative error increases and there is no way to correct it without performing the fusion of the sensors.

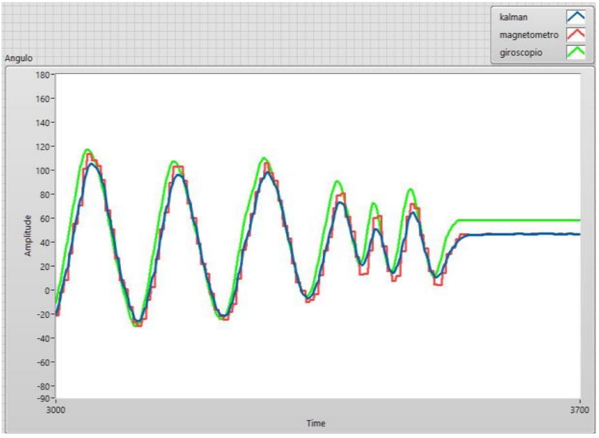

Figure 8. Kalman filter for orientation.

Finally, the encoder is added to the system, in order to have a better estimate of the distance traveled, an extended Kalman filter (EKF) is used, taking into account the non-linearity of the estimation by means of the odometry model [15].

This model is subject to cumulative errors that increase with time, because there is no external reference. These errors can be minimized by integrating a GPS (Global Position Systems) into the system, but this is beyond the scope of this work [16].

- Path Planner

This module is responsible for taking the autonomous vehicle from an initial position to a final, following a trajectory. The algorithm used is the so-called pure pursuit algorithm [17]. For practical purposes, the implementation of "Team 1712" [18] has been used and modified according to the requirement.

With this algorithm it is possible to determine the target speed of the autonomous vehicle depending on the curvature of the segment of the trajectory in which the autonomous vehicle is located, as well as to establish the direction to which it should go knowing its current position and a target point called "Look Ahead Point" (figure 10).

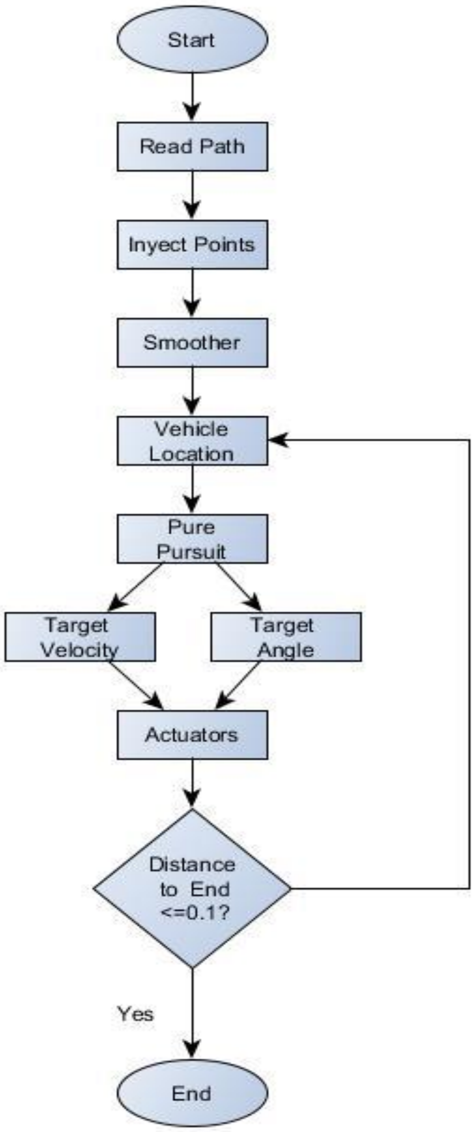

Figure 9. Route planner.

The "Look Ahead Point" is a fundamental parameter in the application of this algorithm, since by varying its value it is possible to vary: the response of the car to deviations from the wanted trajectory, and the stability in which the car follows its trajectory preventing oscillations. Its value can be static or dynamic, that is, static if its value is predetermined by the programmer before the car starts its trajectory, or dynamic when its value depending on characteristics such as the speed of the car and the curvature of the trajectory make its value to get the best response. In this project the static was applied due to its simplicity and rapid implementation.

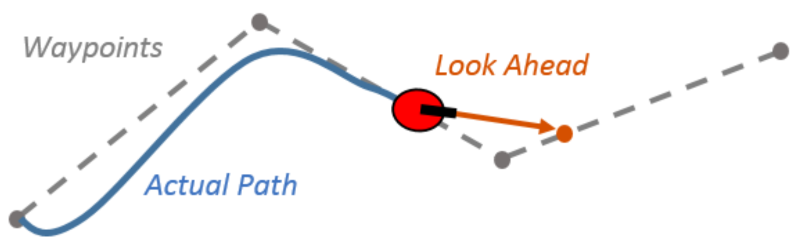

Figure 10. Look Ahead Distance.

v.2,n.1,p.19-31, 2019.

With regard to its value, the choice of a small number will cause the vehicle to quickly seek to approach the desired trajectory, however, as a consequence, the car begins to oscillate in search of the trajectory as shown in figure 11.

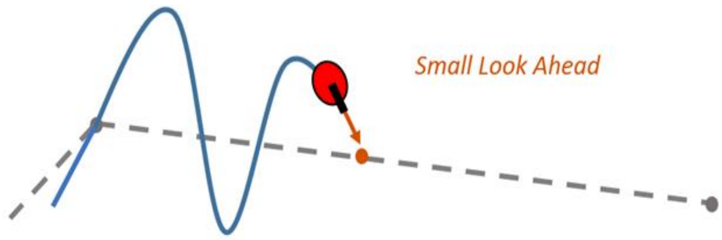

Figure 11. Small Look Ahead [19].

On the other hand, a choice of a large value will cause the car to stop oscillating, however the response to sudden variations in the trajectory becomes very slow, as does the curvature in which the car follows its trajectory (figure 12).

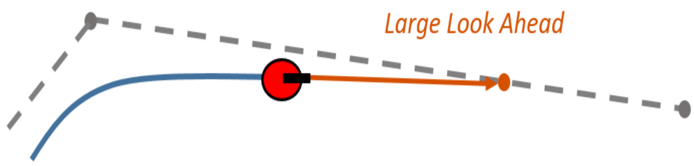

Figure 12. Large Look Ahead [19] .

To obtain greater precision in the calculation of the speed and curvature, points were injected into the original trajectory obtaining in this way closer points, to then pass them through a smoothing stage and achieve continuity in the path.

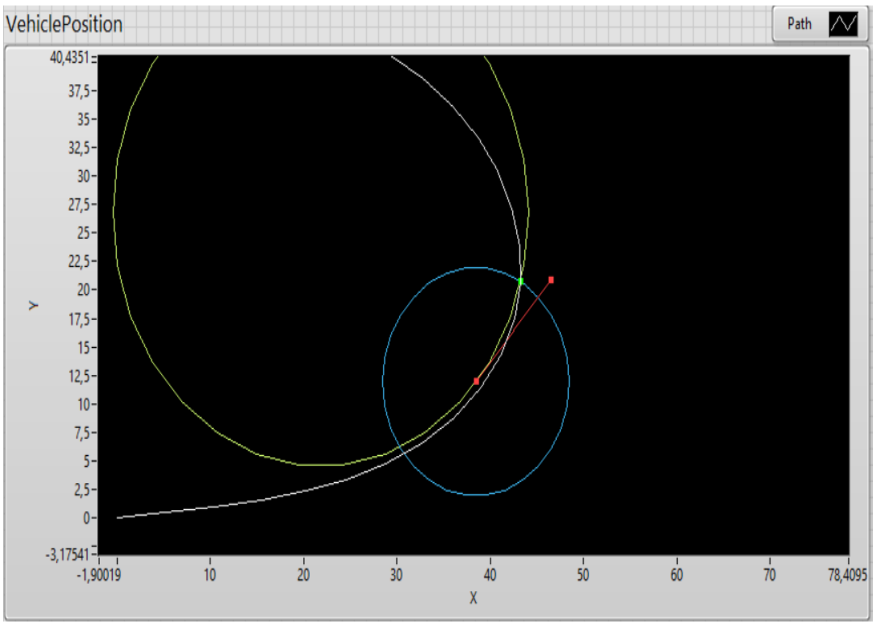

Figure 13. Simulation of Pure Pursuit Algorithm.

FINDINGS AND DISCUSSION

During the tests it was evidenced that the pure pursuit algorithm gives good results in the determination of follow-up of the established trajectory. Due to its robustness, this allows some errors in the acquisition of sensor data or in the tuning of the speed and direction control, however, as it takes into account feedback from a previous state, it allows to update the actuators with consistent values so that small errors of orientation and position are attenuated.

The choice of the best value for the Look Ahead Point was made through experimentation finding that for the speed 2 m/s, a value of Look Ahead Point equal to 1.8m, prevents oscillations and a good response is obtained for more curves closed.

CONCLUSIONS

Autonomous navigation was validated in the tests, demonstrating its effectiveness in tracking a defined trajectory. A set of tests was performed varying the initial position and the orientation to verify that, despite the different initial conditions, in the same way, the car manages to follow a predefined trajectory, as expected by the results of the simulations. Tests in a controlled environment have obtained satisfactory results for the desired purposes, taking into account that the scaled vehicle was used in a race of autonomous scaled cars. For this purpose, the cumulative errors inherent in the system were reduced so that the influence on the result was negligible. In addition, the inertial navigation algorithm turned out to be very effective compared to other types of navigation used for this purpose, achieving a higher response speed due to its low computational requirement.

However, in uncontrolled environments, it has not been very effective, due to the inability of the linear Kalman filter to eliminate electromagnetic distortions that affect the readings of the magnetometer, these readings with distortion considerably affect the calculation of the yaw angle, and consequently, they produce errors in the calculation of the x and y coordinates.

Bearing in mind that the duration of the race does not generate significant cumulative errors, the use of a magnetometer could be eliminated, and thus make the system less sensitive to disturbances in the magnetic field (with the cost that this entails in the absence of an absolute orientation, which is what the magnetometer offered).

ACKNOWLEDGEMENTS

We thank the support of the following institutions: Universidad Católica "Nuestra Señora de la Asunción", Centro de Investigación en Ciencias, Tecnología e Innovación Avanzada (CICTIA), Fundación Parque Tecnológico Itaipu – Paraguay and National Instruments Brazil.

v.2,n.1,p.19-31, 2019.

REFERENCES

[1] IYENGAR, D., & PETERS, D. L. (2015, October). Development of a miniaturized autonomous vehicle: Modification of a 1: 18 scale rc car for autonomous operation. In ASME 2015 Dynamic Systems and Control Conference (pp. V003T50A008-V003T50A008). American Society of Mechanical Engineers.

[2] MCBRIDE, J. (2007). Darpa urban challenge.

[3] URMSON, C., BAGNELL, J. A., BAKER, C. R., HEBERT, M., KELLY, A., RAJKUMAR, R., & TEAM, D. U. C. (2007). Tartan racing: A multi-modal approach to the DARPA

[4] MONTEMERLO, M., BECKER, J., BHAT, S., DAHLKAMP, H., DOLGOV, D., ETTINGER, S., & THRUN, S. (2008). Junior: The Stanford entry in the Urban Challenge. Journal of field Robotics, 25(9), 569-597

[5] THE ECONOMIST. (2013). Look, no hands. Recovered from The economist: http://www.economist.com/news/special-report/21576224-one-day-every-car-may-come-invisible-chauffeur-look-no-hands

[6] DOCKTERMAN, E. (2015). Google's self-driving car may come with airbags on the outside. Time Magazine. Recovered from Time: http://time.com/3758446/googles-self- driving-car-may-come-with-airbags-on-the-outside/

[7] THE DUCKIETOWN FOUNDATION. (2017). The Duckietown Project. Recovered from Duckietown: https://www.duckietown.org/

[8] BENÍTEZ, W., & BOGADO, Y. (2015). Desarrollo de un prototipo de VANT (Vehículo Aéreo No Tripulado) para inspección visual de líneas eléctricas aéreas (Tesis de Grado). Universidad Católica "Nuestra Señora de la Asunción" Campus Alto Paraná. Paraguay.

[9] INVENSENSE. (2013). MPU-6000 and MPU-6050 Product Specification Revision 3.4. Sunnyvale, California, United States of America. Recovered from Invensense: https://store.invensense.com/datasheets/invensense/MPU-6050_DataSheet_V3%204.pdf

[10] LEE, U., OH, J., SHIN, S., SHIM, I., CHOI, J., JUNG, Y., PARK, K., KIM, M., & JUNG, J. (2014). EureCar, KAIST Self-Driving car. Recovered from National Instruments: https://forums.ni.com/t5/Projects-Products/EureCar-KAIST-Self-Driving-car/ta-p/3517884

[11] KOK, M., HOL, J. D., & SCHÖN, T. B. (2017). Using inertial sensors for position and orientation estimation. *arXiv preprint arXiv:1704.06053*.

[12] OZYAGCILAR, T. (2012). Implementing a tilt-compensated eCompass using accelerometer and magnetometer sensors. *Freescale semiconductor, AN*, *4248*.

[13] ZUNAIDI, I., KATO, N., NOMURA, Y., & MATSUI, H. (2006). Positioning system for 4-wheel mobile robot: encoder, gyro and accelerometer data fusion with error model method. *CMU. Journal*, *5*(1).

[14] VIGOUROUX CAVOLINA, D. P. (2010). Implementación de unidad de mediciones inerciales (IMU) para robótica utilizando filtro de Kalman. Sartenejas, Venezuela.

v.2,n.1,p.19-31, 2019.

[15] FISCHER, T., NITSCHE, M. A., & PEDRE, S. (2014). Fusión de de encoders de cuadratura, sensores inerciales y magnéticos para la localización de robots móviles. Facultad de Ciencias Exactas y Naturales - UBA, Buenos Aires.

[16] MOHINDER S., G., & ANGUS P., A. (2008). Kalman Filtering: Theory and Practice Using MATLAB (Third ed.). Hoboken, New Jersey: JOHN WILEY & SONS, INC.

[17] COULTER, R. C. (1992). Implementation of the Pure Pursuit Path Tracking Algorithm.

[18] FRC TEAM 1712 (2018). Implementation of adaptive pure pursuit controller. Recovered from chiefdelphi: https://www.chiefdelphi.com/media/papers/3488.

[19] MATHWORKS (w/d). Pure Pursuit Controller. Recovered from MathWorks: https://www.mathworks.com/help/robotics/ug/pure-pursuit-controller.html

CONTACT INFORMATION

Gregorio Ariel Guerrero Moral (corresponding author)
ariel.guerrero@pti.org.py

Micaela Carolina Jara Ten Kathen
micaela.jara@uc.edu.py

Erid Eulogio Pacheco Viana
erid.pacheco@uc.edu.py

Ariel David Bogado Arce
ariel.bogado@uc.edu.py

Jesús Maria Franco Santacruz
jesus.franco@uc.edu.py