

Design of a Mobile Surveillance System using GPS Navigation and Trajectory Control via waypoints

Marcelo Jaime Quispe Ccachuco, Dr.¹, César Pio Castillo Cáceres, Dr.², Paulo Antonio Tejada Benavides, Br³ y Fabrizio Renato García Gutiérrez, Br⁴.

¹Universidad Católica de Santa María, Perú, mquispec@ucsm.edu.pe

²Universidad Católica de Santa María, Perú, ccastill@ucsm.edu.pe

³Universidad Católica de Santa María, Perú, ptejada.benavides@outlook.com

⁴Universidad Católica de Santa María, Perú, fab45garcia@gmail.com

Abstract— Object recognition is essential in any surveillance system. Being important to identify the entry of intruders to a home or parking in restricted places, are some actions that are performed during a routine. However, the performance of these systems turns out to be inefficient because they remain static and in many cases their viewing angle turns out to be very limited. Therefore, we believe that a mobile surveillance system would be more efficient in reducing citizen insecurity in some closed housing estates, shopping centers or condominiums. Since being a mobile monitoring system would not have the disadvantages that present security systems have.

Reason why, in this project is proposed the design and application of a patrol robot with autonomous navigation, which is able to monitor and travel a specified route. This path can be defined by the user, who when recording a waypoint the robot will move to that location in a relatively straight line, allowing the robot to navigate from one point to another pre-programmed. The navigation of robot will be done by GPS, but not only using this signal. Since, although it is true that this gives us information about the position or speed of a topological form, it does not show us if there are any obstacles in the way.

For this we seek to merge GPS navigation with a sensor-based one, which collects information and with the help of a microprocessor this information serves for the robot to make decisions, since it is sought to be completely autonomous. Finally, the robot will be able to capture images with the help of a camera, storing them in its internal memory. These images are then sent to the user in real time so that the user can analyze them and detect any anomaly act in the shortest possible time.

Keywords— GPS, Navigation, Safety, Trajectory Control, Waypoint.

Digital Object Identifier (DOI):

<http://dx.doi.org/10.18687/LACCEI2022.1.1.508>

ISBN: 978-628-95207-0-5 **ISSN:** 2414-6390

Diseño de un Sistema de Vigilancia Móvil mediante Navegación GPS y Control de la Trayectoria vía waypoints

Marcelo Jaime Quispe Ccachuco, Dr.¹, César Pio Castillo Cáceres, Dr.², Paulo Antonio Tejada Benavides, Br³ y Fabrizio Renato García Gutiérrez, Br⁴.

¹Universidad Católica de Santa María, Perú, mquispec@ucsm.edu.pe

²Universidad Católica de Santa María, Perú, ccastill@ucsm.edu.pe

³Universidad Católica de Santa María, Perú, ptejada.benavides@outlook.com

⁴Universidad Católica de Santa María, Perú, fab45garcia@gmail.com

Resumen— El reconocimiento de objetos es fundamental en cualquier sistema de vigilancia. Siendo importante identificar el ingreso de intrusos a una vivienda o estacionar en lugares restringidos, son algunas acciones que realizará el robot durante una rutina habitual. Sin embargo, el rendimiento de estos sistemas resulta ser poco eficiente, debido a que permanecen estáticos y en muchos casos su ángulo de visión resulta ser muy limitado. Por lo tanto, consideramos que un sistema de vigilancia móvil sería más eficiente en la reducción de la inseguridad ciudadana en algunas urbanizaciones cerradas, centros comerciales o condominios. Puesto que, al ser un sistema de monitoreo móvil no tendría las desventajas que poseen los sistemas de seguridad actuales.

Motivo el cual, en este proyecto se propone el diseño y aplicación de un robot móvil con navegación autónoma, el cual sea capaz de monitorear y recorrer una ruta especificada. Dicha trayectoria puede ser definida por el usuario, que al grabar un waypoint el robot se moverá a esa ubicación en una línea relativamente recta, permitiendo que el robot pueda navegar de un punto a otro preprogramado. La navegación del robot se realizará mediante GPS, pero no únicamente usando esta señal. Puesto que, si bien es cierto que este nos da información sobre la posición o velocidad de una forma topológica, no nos muestra si hay algún obstáculo en la vía.

Para esto buscamos fusionar la navegación GPS con una basada en sensores, la cual recopila información y con ayuda de un microprocesador esta información sirve para que el robot tome decisiones, ya que se busca que este sea completamente autónomo.

Finalmente, el robot podrá capturar imágenes con la ayuda de una cámara, almacenándolas en su memoria interior. Estas imágenes luego se envían al usuario en tiempo real para que el usuario pueda analizarlas y de detectar alguna anomalía actuar en el menor tiempo posible.

Palabras clave— Control de Trayectoria, GPS, Navegación, Seguridad, Waypoint.

Abstract— Object recognition is essential in any surveillance system. Being important to identify the entry of intruders to a home or parking in restricted places, are some actions that are performed during a routine. However, the performance of these systems turns out to be inefficient because they remain static and in many cases their viewing angle turns out to be very limited. Therefore, we believe that a mobile surveillance system would be more efficient in reducing citizen insecurity in some closed housing estates, shopping centers or condominiums. Since being a mobile monitoring system would not have the disadvantages that present security systems have.

Reason why, in this project is proposed the design and application of a patrol robot with autonomous navigation, which is able to monitor and travel a specified route. This path can be defined by the user, who when recording a waypoint the robot will move to that location in a relatively straight line, allowing the robot to navigate from one point to another pre-programmed. The navigation of robot will be done by GPS, but not only using this signal. Since, although it is true that this gives us information about the position or speed of a topological form, it does not show us if there are any obstacles in the way.

For this we seek to merge GPS navigation with a sensor-based one, which collects information and with the help of a microprocessor this information serves for the robot to make decisions, since it is sought to be completely autonomous. Finally, the robot will be able to capture images with the help of a camera, storing them in its internal memory. These images are then sent to the user in real time so that the user can analyze them and detect any anomaly act in the shortest possible time.

Keywords— GPS, Navigation, Safety, Trajectory Control, Waypoint.

I. INTRODUCCIÓN.

Actualmente las personas sienten inseguridad en todo momento, debido a la creciente ola de robos con distintas modalidades [12]. Razón la cual, los últimos años muchas personas han implementado sistemas de seguridad en sus viviendas, urbanizaciones o centros de trabajo [13]. En este proyecto se plantea el diseño e implementación de un sistema de vigilancia móvil, el cual sea capaz de monitorear un área recorriendo una ruta de la zona correspondiente.

Este robot podría capturar imágenes con la ayuda de una cámara, las cuales se envían a la nube donde el usuario en tiempo real analizará y observará en búsqueda de algún problema. Este sistema busca reducir la inseguridad ciudadana en alguna zona en específico, ya que, al ser un sistema móvil, no tendría las desventajas que poseen los sistemas de seguridad actuales convencionales.

A. La Navegación Autónoma

Un Sistema de Navegación Autónoma trata en todo momento conocer de forma continua el vector de posición

respecto al origen de un sistema coordenado. Para averiguar este vector de posición, se puede partir del vector velocidad y realizar una integración respecto al tiempo [10]. Otra opción sería partir del vector aceleración y tras dos integrales respecto al tiempo, hallar el vector posición. Ambos cálculos requieren el conocimiento de las condiciones iniciales.

Para hallar la posición, la condición inicial se hallará al encontrarse en el origen del sistema o en un punto de coordenadas conocidas. Por otro lado, para hallar la velocidad, la condición inicial es velocidad cero.

Uno de los problemas fundamentales de la cartografía ha sido conseguir de forma precisa determinar la posición en el espacio de los fenómenos a cartografiar. Tradicionalmente se han utilizado técnicas de topografía basadas en la triangulación. Hoy en día se cuenta con el sistema GPS que permite determinar las coordenadas de cualquier punto de la superficie terrestre con mayor precisión que los antiguos métodos topográficos.

B. La Tecnología del Sistema de Posicionamiento Global (GPS)

La tecnología GPS (Global Positioning System) fue diseñada originalmente con propósitos militares, pero rápidamente se vio su aplicabilidad en el mundo civil, fundamentalmente en navegación y topografía. El sistema consta de tres componentes [11]:

Componente espacial: Formado por 24 satélites situados a unos 20200 km de la Tierra y que pasan por el mismo lugar cada 12 horas.

Componente de control: Formado por una serie de estaciones de observación cercanas al ecuador encargados de controlar la posición orbital de los satélites y calibrar y sincronizar los relojes.

Usuarios con un receptor GPS: Entre las actividades la navegación marítima o terrestre, excursionismo, topografía, control de maquinaria, etc.

Dependiendo de las necesidades existen tres modos de utilización de un dispositivo GPS [9]: Navegación autónoma con un receptor simple, la precisión es de 20 metros para usuarios militares y 100 metros para usuarios civiles. Posicionamiento diferencial corregido (DGPS) con precisiones de 0.5 a 5 metros utilizado en SIG. Posicionamiento diferencial de fase con precisiones entre 0.5 y 20 mm.

El cálculo del posicionamiento se basa en la medición de la distancia desde la posición de cada satélite a Tierra. Puesto que la señal emitida por el satélite incluye la hora en que fue

emitida y el receptor conoce la hora de llegada, la distancia se puede calcular como: $d=c\Delta t$ donde c es la velocidad de la luz.

Se genera de este modo una pseudoesfera con centro en el satélite y radio igual a la distancia medida. Si se dispone de tres satélites se tienen 3 pseudoesferas cuya intersección genera un único punto que es la posición del receptor. Sin embargo, para obtener una medida hacen falta al menos cuatro satélites debido a las diferencias temporales en la recepción de las señales de los satélites [8].



Fig. 1 Descripción del funcionamiento del sistema GPS.[1]

La técnica DGPS permite solventar muchos de estos errores logrando una precisión de 2 a 3 metros. Se basa en la conexión del receptor a un receptor de referencia fijo. Este último, puesto que conoce su posición, puede calcular el error de la señal del GPS y transmitirlo al receptor móvil para que efectúe la corrección.

Esta corrección puede realizarse en tiempo real o en postproceso mediante programas que descargan por internet los ficheros con los errores y los tiempos en que se han medido estos errores para corregir las posiciones medidas por el receptor móvil y almacenado en otro fichero.

La técnica de Posicionamiento diferencial de fase requiere como mínimo dos receptores que deben utilizarse de forma simultánea tomando varias medidas para conseguir una reducción de errores mediante procedimientos estadísticos.

C. Características del Robot Móvil

Al realizar el modelamiento de la planta se tomaron en consideración varios factores, principalmente el terreno por donde transitará el robot puesto que de no considerar esto se correría con el riesgo de que las llantas se atasquen o simplemente no se genere un agarre adecuado [4]. Por otro lado, el robot fue creado para realizar tareas de vigilancia, debido a esto, al realizar el diseño del chasis se consideró que

sea suficientemente amplia como para que entren los componentes e inclusive la cámara.

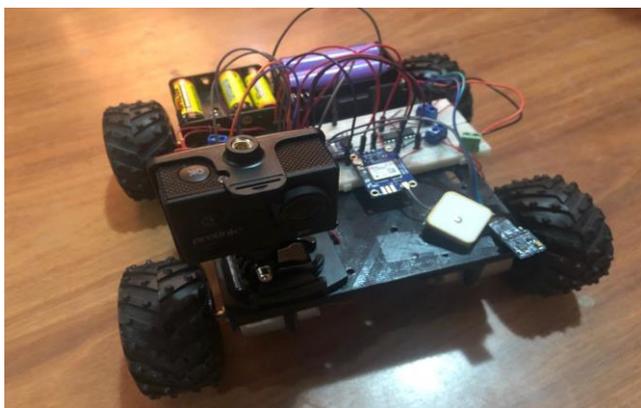


Fig. 2 Ensamble Final del Robot Autónomo GPS.

Así mismo se hizo el diseño de unos acoples para las ruedas hacia el motor, las cuales garantizarán la correcta posición de los motores y una correcta transmisión hacia las ruedas. Sin embargo; de nada serviría tener un buen acople hacia las ruedas para la transmisión si no consideramos que al momento de fijar las ruedas al chasis (donde se fijara la cámara), estas uniones no sean lo suficientemente estables y eviten generar demasiada vibración. Por último, también se consideró el diseño de un soporte con amortiguadores a los lados para la Pro-Link, reduciendo la vibración producida por el trayecto y evitando que la cámara tambalee logrando un buen video.

D. El Protocolo National Marine Electronics Association (NMEA)

La comunicación con el GPS se puede realizar mediante protocolo SiRF creado por SiRF Technology, Inc. o bien con los comandos NMEA (National Marine Electronics Association). Esta asociación se encarga de definir un estándar, el NMEA 0183, que consiste en una interfase eléctrica y un protocolo de datos para la comunicación entre instrumentos marinos. NMEA se establece como un grupo de trabajo para el desarrollo de nuevos estándares de comunicación de datos para dispositivos marinos a bordo de las embarcaciones. Este protocolo se lanza por primera vez en marzo 1983, y su última versión fue publicada en el 2001.

El protocolo NMEA es usado para la comunicación entre dispositivos de uso marino para transmitir datos. La salida NMEA es EIA-422A (Electronic Industries Alliance), pero para la mayoría de los propósitos puede considerarlo RS-232 compatible. Todos los datos son transmitidos a través de sentencias con caracteres ASCII, cada instrucción comienza por "\$" y termina con <CR><LF>. Los primeros dos caracteres después de "\$" son los que identifican el equipo, así, por ejemplo; "GP" que se usa para identificación de datos

GPS y los siguientes tres caracteres es el identificador del tipo de sentencia que está enviando. Los tres tipos de sentencias NMEA que existen son los de envío (Talker Sentences), Origen del equipo (Proprietary Sentences) y consulta (Query Sentences).

Los datos están delimitados por coma, deben incluirse todas las comas, ya que actúan como marcas. Una suma de verificación adicional es agregada opcionalmente; aunque para algunos tipos de instrumentos es obligatoria. Siendo los comandos soportados por GPS los que muestra la tabla I.

TABLA I
COMANDOS NMEA SOPORTADOS POR GPS

Comando NMEA	Descripción
GGA	Datos Fijos del Sistema Global de Posicionamiento.
GSA	Modo de operación de receptor GPS, Satélites empleados para navegación y valores DOP.
GLL	Posición Geográfica Latitud-Longitud.
GSV	Número de Satélites en vista, números de PRN, elevación, azimut & valores SNR.
RMC	Mínimo de Datos Específicos Recomendados.
VTG	Velocidad sobre la tierra y velocidad de la misma.

II. MATERIALES Y MÉTODOS.

A. Diseño del Robot Móvil

Para el diseño de la planta mecánica se consideró las condiciones en las cuales va a operar el vehículo. De manera que, el área de operación del robot móvil no es una pista del tipo asfáltica, más bien es del tipo adoquinada como se observa en la figura 3.



Fig. 3 Primera consideración para diseñar la planta.

Teniendo como referencia que el área de la pista es tipo adoquinada, se procedió a buscar neumáticos que vayan acorde con los requisitos del terreno, de las cuales se

seleccionó unas que contaban con una estructura que facilitaba que giren en la pista en la cual se operaría.

Asimismo, se procedió a seleccionar un motor que garantice el suficiente torque para poder movilizar todo el robot. Por lo que se optó usar un motorreductor de 200 rpm (velocidad moderada - torque alto). Estos motores son los que se muestran en la figura 4 y se alimentan con 12V en corriente continua.

Posteriormente, luego de seleccionar el tamaño de las llantas y el motor con un eje adecuado, se procedió a realizar el diseño de las ruedas usando el software Fusion 360 para que correspondan al tamaño requerido y pueda ser impreso en 3D.

Para el diseño de la planta se seleccionó un microcontrolador que debía contar con un número de entradas analógicas y digitales, 6 y 13 respectivamente. Además de ser compatible con la comunicación I2C y con la comunicación serial entre el mismo y el módulo wifi, ya que uno de los principales requisitos es que el vehículo opere de manera autónoma y se haga un monitoreo de su posición en todo momento. Se optó por utilizar la placa de desarrollo Arduino Uno, la cual cuenta con un microprocesador Atmega 328p y cumple con los requisitos previamente mencionados [6].

De acuerdo con lo mencionado, se planteó un control independiente de los motores mediante el uso de 2 drivers L293D, que en configuración de semi puente H cada uno controle dos motores. No obstante, esto significó que se necesitó de dos fuentes de alimentación, una de 5 V, la cual sirvió para alimentar todo el circuito de control, y las activaciones necesarias; y una segunda fuente de 12 voltios, la cual se usó para dar la potencia a los motores, ya que el Arduino solo se entregaba a la señal de control, tal como se muestra en la figura 4.

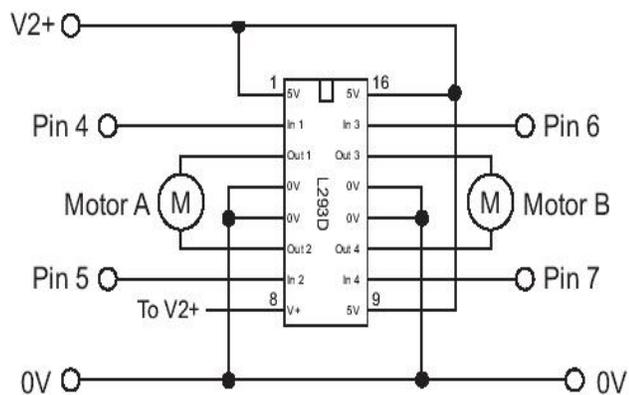


Fig. 4 Conexión del Drive L293D en configuración de Semipuente H. [2]

Para la primera fuente, se optó por una configuración de 4 baterías de 1.2 V y 2700 mAh conectadas en serie, logrando así una fuente de alimentación de 4.8 V y 2700 mAh de capacidad. Seguidamente para la alimentación de los motores

se requería un voltaje de alimentación de 12 V, por lo que se usó las baterías del tipo 18650 de 3.7 V y 2000 mAh. Esto significa que se necesitó una configuración de 3 de estas baterías conectadas en serie generando un voltaje de 11.1 V con una capacidad de 2000 mAh.

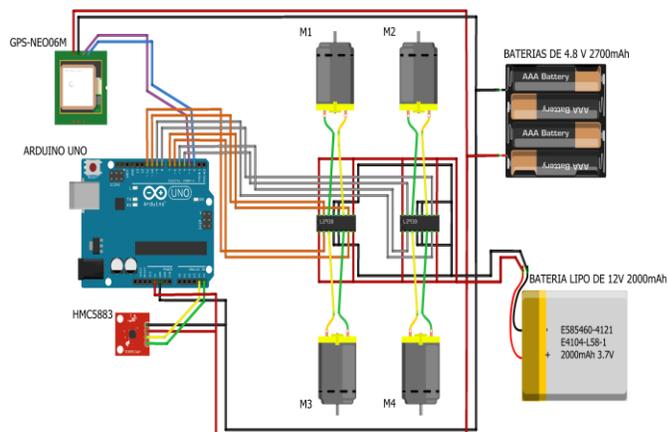


Fig. 5 Esquema de conexiones del robot.

Ya con la selección de las fuentes de alimentación y el circuito del driver diseñado se pudo estimar el área de trabajo necesario con el cual se diseñó el chasis y acoples de los motores del robot. Asimismo, se consideró dentro del diseño del chasis los agujeros en los que irían ensamblados los componentes para que sea más sencillo la ubicación de todas las partes. Después de que se realizó el diseño, las piezas fueron mandadas a imprimir, finalmente pasaron al proceso de ensamble de la estructura del vehículo, con los motores, llantas y su circuito de control, el resultado se muestra en las figuras 5 y 6.

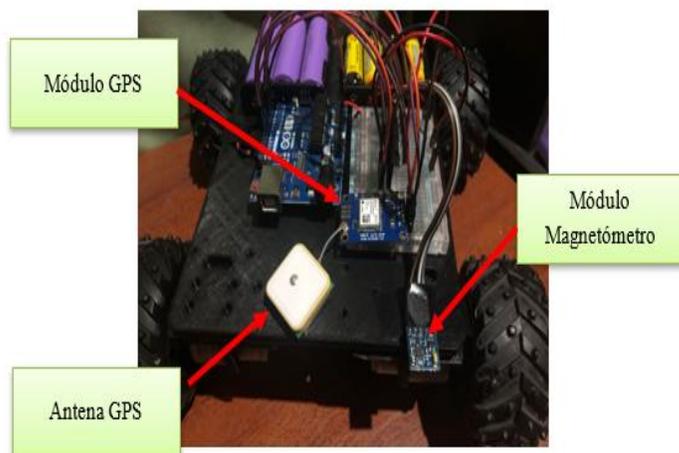


Fig. 6 Diseño y Montaje final de la Planta.

B. Obtención de las Coordenadas de Ubicación GPS

Para la obtención de las coordenadas en tiempo real se hizo uso de un módulo GPS para un modo de trabajo en espacio abierto, el cual viene con un módulo de serie U-Blox

NEO M8N equipado en el PCB (Printed Circuit Board) y una antena cerámica de amplio alcance.

Lo primero que se realizó fue un código para el Arduino, el cual pueda leer la información en bruto del GPS empleando la librería “SoftwareSerial” que proporciona Arduino. Con dicha librería se consiguió establecer puertos seriales aparte de los que tiene el Arduino por defecto (pin 0 - pin 1), los cuales se usaron para la comunicación entre el Arduino y el ordenador. Dado esto, se procedió a crear dos pines más para que el módulo GPS no presente ningún conflicto [6] con los antes mencionados. Obteniéndose el resultado que muestra la figura 7.

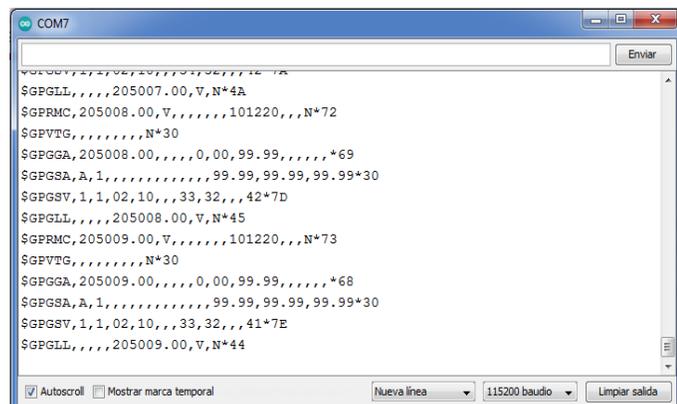


Fig. 7 Datos extraídos por el Módulo GPS en codificación NMEA.

Asimismo, en la figura 7 se puede observar la información tal cual la envía el módulo GPS, siendo estos valores difíciles de interpretar y no se parecen en absoluto a coordenadas de latitud o longitud. Esto se debió a que el módulo está codificado bajo el protocolo NMEA, las cuales son sentencias estándares internacionales para la recepción de datos GPS.

De la trama de datos que envió el módulo GPS se pudo obtener varias variables. Siendo las importantes para este proyecto, la información de posicionamiento, que se refiere a la latitud y la longitud. Así mismo; conociendo cada interpretación y característica de los estándares NMEA se pudo descifrar para obtener cada información solicitada. Además de la aplicación de una de las librerías más usadas cuando se trabaja con GPS, la cual es conocida como “Tiny GPS”.

Obteniéndose con este código la latitud, longitud, altitud, e incluso la fecha, hora y el día de la simulación. Tal como se muestra en la figura 8.

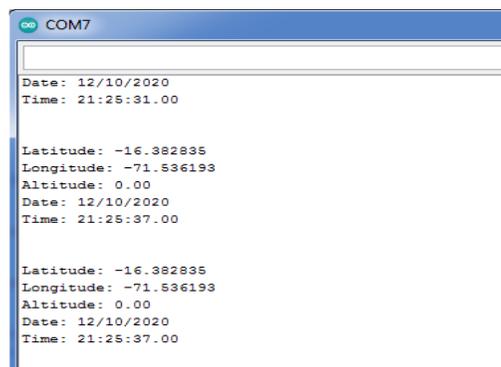


Fig. 8 Decodificación NMEA de la Coordenadas de Ubicación GPS.

C. Obtención de la Orientación del Robot

Para que el robot pueda orientarse en el área de trabajo, fue necesario utilizar un magnetómetro o también llamado brújula de 3 ejes. Con este se pudo leer los componentes del campo magnético presente en un lugar en específico de la tierra [5]. De esta forma conociendo la dirección del campo magnético terrestre se pudo calcular la orientación con respecto al norte magnético de la tierra, esto siempre y cuando el sensor no esté expuesto a algún campo magnético externo u algún objeto metálico que altere el campo magnético terrestre.

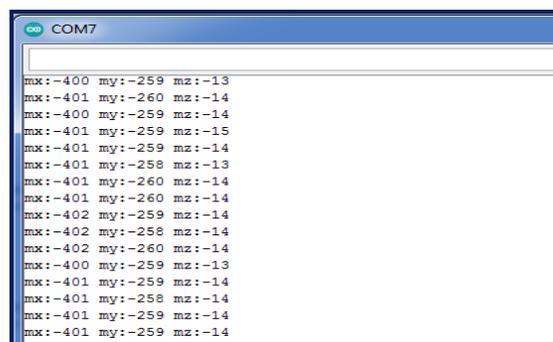


Fig. 9 Componentes del Campo Magnético correspondientes a la Orientación del Robot.

El protocolo de comunicación que presenta es I²C (Inter-Integrated Circuit), puesto que se necesitó una comunicación fluida sin interferencia de por medio y cuya dirección no se consiga cambiar, la cual es 0x1E. Debido a esta razón no se pudo conectar otro magnetómetro en el bus I²C del controlador [6], pero si se podía conectar otros dispositivos I²C que tengan una dirección distinta. Asimismo, se puede afirmar que este módulo tiene un funcionamiento en base a ganancias, puesto que se logró configurar el rango de medición en Gales ($9.81 \text{ m/s}^2 = 981 \text{ Ga}$) con una precisión desde $\pm 0.88\text{Ga}$ hasta $\pm 8.1\text{Ga}$; al que se le aplicó ganancias de 1370 a 230 respectivamente y de esta forma obtener un rango de salida de 12 bits.

Para obtener las coordenadas del campo magnético terrestre correspondientes a donde se está direccionando el vehículo, se realizó la codificación mediante la librería I2Cdev, la cual fue creada por Jeff Rowberg. Obteniéndose lo que se muestra en la figura 9. Es importante mencionar que estos valores son valores de ganancia, para tener el valor real hay que dividirlo entre 1090 (ganancia por defecto) y de esta forma obtener la medida en Gauss.

El campo magnético terrestre dependiendo del lugar puede variar entre 0,25-0,65 Ga, teniendo en cuenta esto es mejor trabajar con el rango por defecto si solo se va a medir el campo terrestre. Si queremos mayor precisión podemos bajar el rango a 0.88 Ga. pero se podría saturar la lectura si existe un campo magnético externo. Como se mencionó anteriormente estos son los componentes del campo magnético terrestre, sin embargo, para este proyecto solo se necesitó saber hacia dónde se está dirigiendo el vehículo. Esto se logró conseguir hallando la resultante de los componentes x – y (1), de manera que obtenga el ángulo que está haciendo el robot con respecto al Norte, donde este significa 0 grados.

$$\theta = \text{atan}\left(\frac{M_y}{M_x}\right) \quad (1)$$

Así teniendo en cuenta la ecuación anterior, se pudo calcular la dirección a la que está yendo el vehículo. Por el contrario, debido al impacto de la declinación magnética al momento de realizar las pruebas saltó a la vista que se producían pequeños errores, los cuales se corrigieron. Para esto, existe una página llamada: “Find the magnetic declination at your location” - <https://www.magnetic-declination.com/>, en la cual, usando las coordenadas obtenidas por el GPS se pudo obtener la declinación magnética de la zona de estudio.

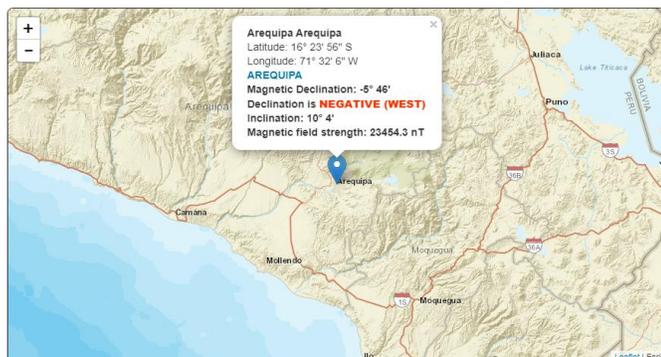


Fig. 10 Declinación magnética en la ciudad de Arequipa.

Se puede ver que se obtuvo una declinación magnética de -5°46'. Y para corregir el ángulo de la dirección del vehículo lo que se hizo fue que restar el ángulo menos la declinación magnética (2), aplicando la siguiente ecuación.

$$\text{angulo} = \text{angulo} - \text{declinación} \quad (2)$$

Después de modificar el código con las correcciones pertinentes, se obtuvo el resultado que muestra la figura 11. El ángulo marcado de color rojo es el equivalente, que se obtiene al cambiar el valor medido de -180 a 180 por uno de 0 a 360, y este es el que nos interesa [5], ya que con este se logró definir un valor vectorial desde el cual se pudo determinar la dirección en la que el vehículo está orientado.

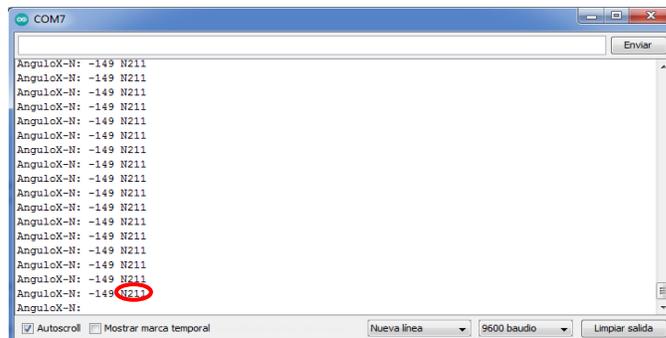


Fig. 11 Corrección del ángulo de dirección para determinar la orientación del robot.

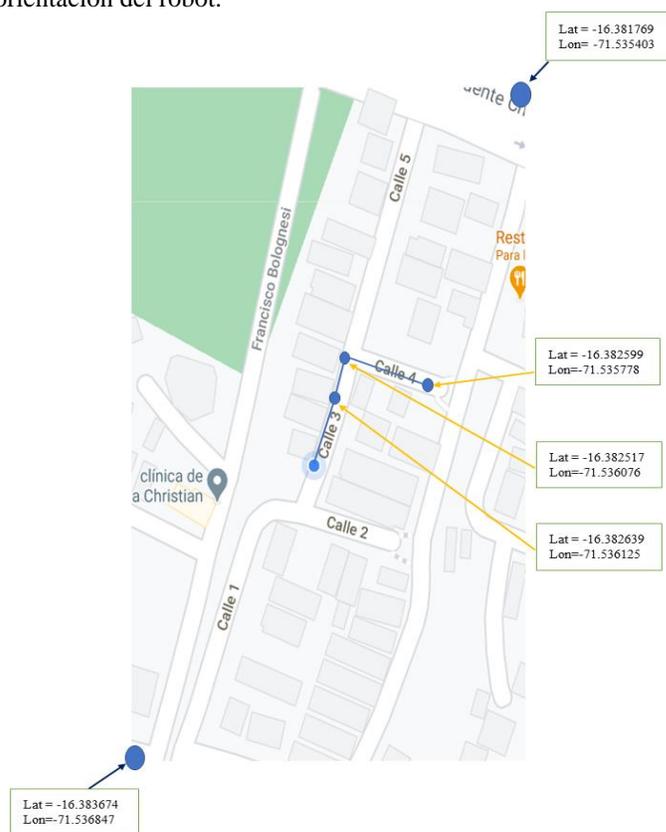


Fig. 12 Delimitación del área de trabajo y de los puntos de paso de la trayectoria del robot.

D. Definición de la Trayectoria

Teniendo la planta completa, se procedió a definir la trayectoria por la cual deberá pasar el robot. Por lo que

inicialmente se delimitó el área de trabajo, puesto que el vehículo estará destinado a patrullar en una zona en específico. Con ayuda de Google Maps, se extrajeron las coordenadas del área delimitada y posteriormente las coordenadas de los puntos de paso (waypoints) por los cuales se deseó que el robot realizase el patrullaje, tal como se muestra en la figura 12. Con esto se definió a los waypoints y el área de trabajo, los cuales fueron cruciales para determinar qué acciones deberá hacer el vehículo.

E. Control de movimiento del robot

Luego de determinar y conocer los waypoints, se definió qué tiene que hacer el robot, para lo cual se consideró el siguiente caso. Teniendo presente que el robot se encontró en su posición inicial pero no con una orientación inicial en dirección del primer waypoint, generando un pequeño ángulo entre ambos, tal como se muestra en la figura 13.

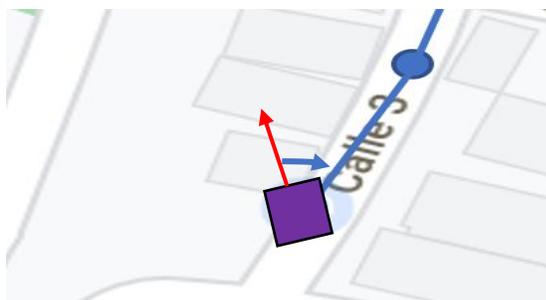


Fig. 13 Posición inicial del robot desfasado de la orientación inicial en dirección del primer waypoint.

El objetivo del control de movimiento era tratar de reducir este ángulo hasta que sea cercano a 0, para luego poder avanzar al waypoint. Esto significa que el robot tuvo que rotar, casi sin moverse de su posición de origen, por lo que se necesitó calcular este ángulo de error [3]. Esto se plasma de una mejor manera en las figuras 14 y 15. Considerando que se conoce algunas de estas variables, como son la latitud y la longitud actual; la orientación actual del robot; la latitud y la longitud deseadas, que fueron extraídas con ayuda de Google Maps. Lo que faltaría calcular era la orientación del waypoint y el ángulo de error.

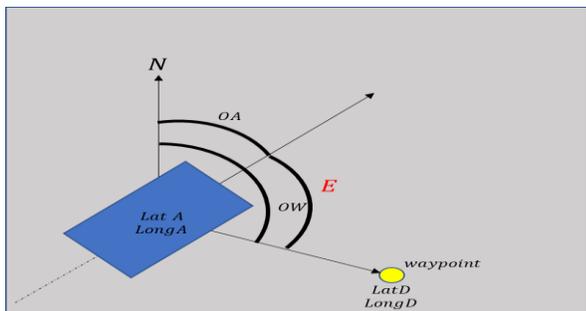


Fig. 14 Sistema de ubicación para hallar el error entre la orientación actual y la orientación del waypoint.

Donde:

- LatA: Latitud Actual
- LongA: Longitud Actual
- LatD: Latitud Deseada
- LongD: Longitud Deseada
- OA: Orientación Actual
- OW: Orientación del Waypoint
- E: Ángulo de Error

Encontrar este ángulo fue una de las mayores cuestiones a lo largo de todo el proyecto, hasta que se encontró el artículo: “Fórmula para encontrar el ángulo de rumbo o dirección entre dos puntos: Latitud, Longitud”, redactado por Akshay Upadhyay (2019) quien trabajó con la empresa Semi Government Company. En este artículo, Akshay logró el desarrollo de un algoritmo para encontrar el ángulo de dirección entre dos puntos, considerando como variables conocidas la ubicación de los dos puntos. Además, el ángulo resultante es el formado por la dirección del norte del primer punto y la posición del segundo, que es exactamente lo que se estaba buscando. De dicho artículo se puede extraer la siguiente ecuación, la cual permite hallar el ángulo de dirección entre dos puntos (3).

$$\beta = \text{atan2}(X, Y) \quad (3)$$

Donde, X - Y son dos cantidades y se pueden calcular mediante las siguientes expresiones (4) y (5).

$$X = \cos \theta b * \sin \Delta L \quad (4)$$

$$Y = \cos \theta a * \sin \theta b - \sin \theta a * \cos \theta b * \cos \Delta L \quad (5)$$

Así tenemos:

- θb puede ser interpretado como la latitud deseada.
- θa puede ser interpretado como la latitud actual.
- ΔL se interpreta como la diferencia entre la longitud deseada y la actual

De la misma forma para hallar la orientación del waypoint, se tuvo que considerar el microcontrolador a emplear. Puesto que, para el caso de Arduino, si se deseaba usar la función $\text{atan2}()$ se necesitaba que los argumentos estén en radianes y considerar que el sistema trabaje con un ángulo de 0 a 360 donde 0 es el norte. Por lo tanto, si es que el resultado saliese un ángulo negativo, habría que $2 * \pi$ sumarle para convertirlo de 0 a 360.

Con la obtención de la orientación del waypoint, se procedió a hallar el ángulo de error, que simplemente fue la diferencia entre la orientación del waypoint, y la orientación actual. Si bien se consiguió el ángulo de error y se sabe cuánto debe girar el robot [10], el módulo magnetómetro tiene un rango de precisión de $\pm 2^\circ$ que se consideró en la precisión de

la medición del error, así como se muestra en la siguiente expresión (6).

$$E = OW - OA, \quad E \pm 2^\circ \quad (6)$$

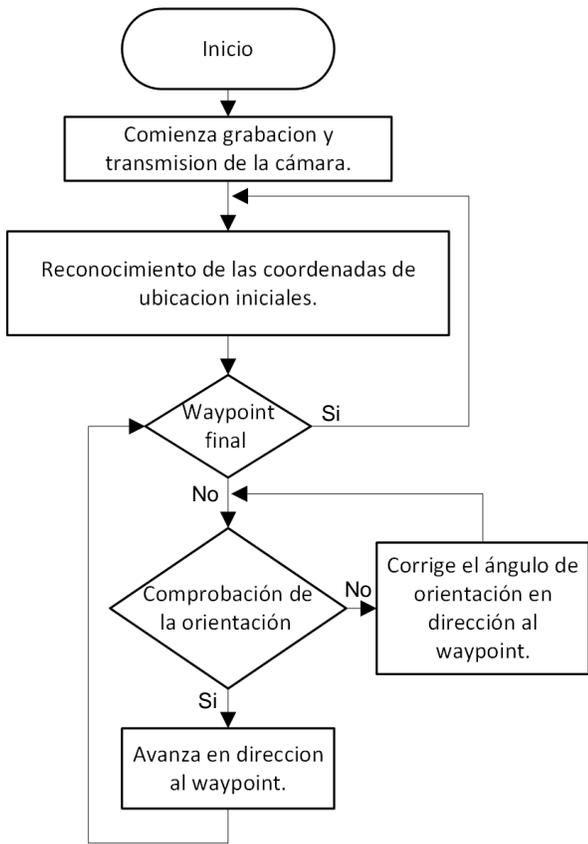


Fig. 15 Diagrama de flujo del control de movimiento del sistema de vigilancia móvil.

Ya una vez obtenido el ángulo que debe girar el robot, se buscó ahora lograr reducir este valor de manera que sea cercano a 0. Para esto, el robot tuvo que girar casi desde su centro de gravedad, lo que significó un problema ya que el robot no cuenta con un sistema diferencial. Sin embargo, se planteó un movimiento coordinado de las llantas, donde por ejemplo para girar hacia la derecha, la llanta trasera del lado izquierdo debió girar hacia adelante, y la llanta delantera del lado derecho hacia atrás y de forma contraria si se busca girar a la izquierda. Esto se puede apreciar de una mejor manera en la figura 16.

Aquí también se consideró que para que esto funcione, los motores deben de girar a la misma velocidad y deben tener un torque elevado, como en el presente trabajo. Una vez que el ángulo de error sea 0 o muy cercano a 0, el vehículo se desplazó en línea recta al waypoint deseado, con un rango de error mínimo, ya que es casi imposible que se posicione sobre

este, seguidamente se repitió el proceso para cada punto de paso.

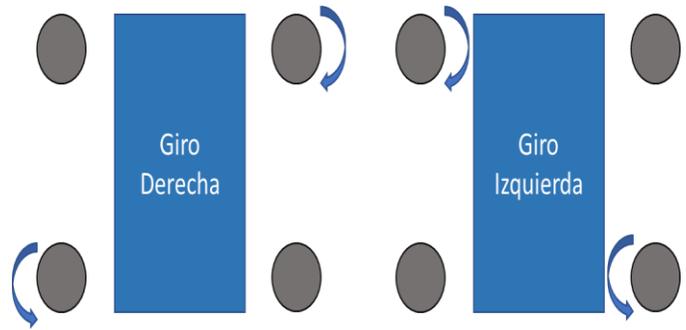
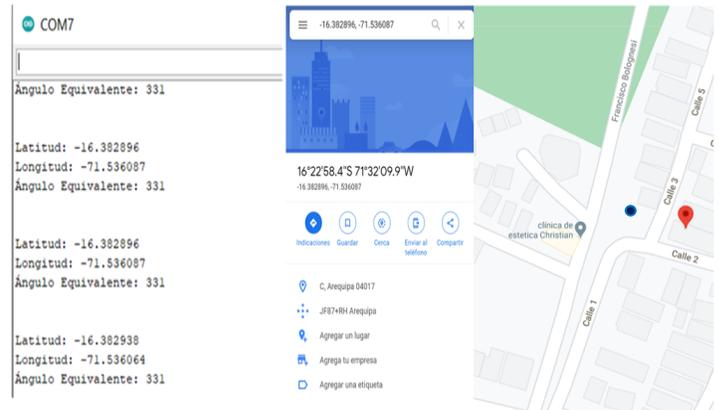


Fig. 16 Control del movimiento de las ruedas del robot.

III. RESULTADOS Y DISCUSIÓN.

En un primer momento, para facilitar el comportamiento del sistema del robot se incorporó lo que es la extracción de las coordenadas GPS y la extracción de la orientación del vehículo, todo en el mismo código. Además, se redujo el código del módulo GPS como para que solo extraiga los valores de interés que son la latitud y longitud, e ignore otra información como hora, día, etc.

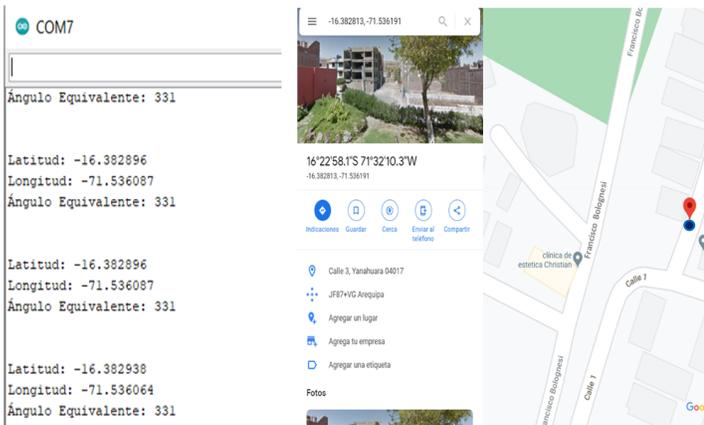
Ya obtenido esto se procedió a comprobar cuál es el error de precisión del módulo GPS. Para esto, con ayuda de Google Maps, graficaremos su latitud y longitud con la posición actual del vehículo. En este caso se puede ver que hay un error considerable entre la posición actual (círculo azul), y la posición entregada por el módulo (marca roja), figura 17; esto se debe a que esta medición fue realizada en un espacio interior (un edificio), lo que disminuye la señal y aumenta el error [7]. Luego de probar en un espacio abierto y con ayuda de Google Maps se obtuvo el siguiente resultado de la figura 18.



a) Lectura de la Posición GPS

b) Lectura de la Posición por Google

Fig. 17 Obtención de la posición y orientación bajo un espacio cerrado.



a) Lectura de la posición del Módulo GPS b) Lectura de la posición otorgada por Google Maps

Fig. 18 Obtención de la posición y orientación bajo un espacio abierto.

Por otra parte, se realizaron tres pruebas para la trayectoria definida anteriormente, que es la que se muestra en la siguiente figura 19. Para realizar las pruebas se estableció un protocolo de ejecución el cual consiste en lo siguiente; primeramente, el robot fue colocado en una posición inicial, teniendo en cuenta el ángulo existente que pueda producir la orientación del robot con respecto a la dirección de los waypoints, seguidamente se dio la corrección de la orientación acercándose lo más posible a un ángulo nulo, para luego avanzar hacia el primer punto de paso. Por último, se tuvo el desplazamiento en la trayectoria al último waypoint.



Fig. 19 Obtención de la posición y orientación bajo un espacio abierto.

No obstante, se pudo notar que debido a las irregularidades que presentaba la pista, la trayectoria se veía

un poco afectada. A pesar de ello, el control de orientación ayudó a mantener el curso deseado.

IV. CONCLUSIONES.

Se eligieron motores de un torque elevado de hasta 50kgf.cm, 200 rpm y 12 voltios, y en base a estos motores, se diseñaron lo que son los acoples de los motores al chasis, y los aros para las llantas.

Con esta consideración se prosiguió a modela el vehículo y obtener los planos, luego se dio inicio al diseño del circuito de control y navegación GPS. La mayoría de módulo GPS o GSM trabajan bajo el protocolo NMEA por sus siglas en inglés “National Marine Electronics Asociation”, las cuales son sentencias estándares internacionales para la recepción de datos GPS y GSM por parte de los satélites, por lo que se vio necesario el uso de librerías, en específico la librería Tiny GPS para la descriptación de esta información, y poder recibir lo que es latitud y longitud.

Por otro lado, el módulo Magnetómetro trabaja con un protocolo de comunicación I²C, con una dirección que no se puede cambiar la cual es 0x1E, por esta razón no se puede conectar otro del mismo tipo en el bus I²C del controlador, pero si podemos conectar otros dispositivos I²C que tengan una dirección distinta.

Así mismo se determinó que para conseguir una buena comunicación durante la Navegación Autónoma mediante GPS la selección del controlador es fundamental, ya que este debe contar con cierto número de entradas analógicas y digitales, asimismo, debe contar con compatibilidad para comunicación I²C, asimismo, debe de ser compatible con comunicación serial entre el mismo y módulos wifi, ya que uno de los principales requisitos es que el vehículo opere de manera autónoma y se haga un monitoreo de su posición en todo momento.

Para calcular la dirección a la que está yendo el vehículo se hizo uso de esta ecuación $\theta = \text{atan}(M_y/M_x)$. Además, debido al impacto de la declinación magnética al momento de realizar las pruebas saltó a la vista que se producían pequeños errores, los cuales se corrigieron. Para esto, existe una página llamada: “Find the magnetic declination at your location” - <https://www.magnetic-declination.com/>, en la cual, usando las coordenadas obtenidas por el GPS se pudo obtener la declinación magnética de la zona de estudio.

Encontrar el ángulo de error fue una de las mayores cuestiones a lo largo de todo el proyecto, hasta que se encontró el artículo: “Fórmula para encontrar el ángulo de rumbo o dirección entre dos puntos: Latitud, Longitud”, redactado por Akshay Upadhyay (2019). Donde dicho ángulo resultante es el formado por la dirección del norte del primer punto y la posición del segundo.

Además; para hallar la orientación del waypoint, se tuvo que considerar el microcontrolador a emplear. Puesto que, para el caso de Arduino, si se deseaba usar la función $\text{atan2}()$ se necesitaba que los argumentos estén en radianes y considerar que el sistema trabaje con un ángulo de 0 a 360

donde 0 es el norte. Por lo tanto, si es que el resultado saliese un ángulo negativo, habría que sumarle $2 * \pi$ para convertirlo de 0 a 360.

Una vez que se obtuvo la orientación del waypoint, se halló el ángulo de error, que simplemente es la diferencia entre la orientación del waypoint, y la orientación actual. Obteniéndose un ángulo equivalente de 331° con un error de $\pm 10^\circ$ en zona de exteriores, el cual debe girar el robot móvil, en un tiempo de 5 a 7 segundos para alinearse y desplazarse hacia el primer waypoint.

Finalmente, al momento de recibir la posición entregada por el módulo GPS, esta se comprobó usando Google Maps, y se pudo apreciar que, en espacios cerrados, por ejemplo, dentro de viviendas, el error aumenta considerablemente, sin embargo, en espacios exteriores, su precisión aumenta y se puede trabajar de forma correcta. Así mismo, la precisión dependió de muchos factores, entre los considerables está que momento del día era y la posición de los satélites en ese momento.

REFERENCIAS

- [1] Pacheco Germán, Cardozo Lenin “Sistema de Posicionamiento Global” Universidad de los Andes, Facultad de Ingeniería, 2014.
- [2] Compañía Nacional de Robótica “Jugando con el Driver del Motor L293D” 2021.
- [3] Liao, G.H. “Configuring Waypoints and Patterns for Autonomous Arduino Robot with GPS and Bluetooth Using an Android App” Lecture Notes in Networks and Systems 296, pp. 304-312, 2022.
- [4] Abrar, M.M., Islam, R., Shanto, M.A.H. “An Autonomous Delivery Robot to Prevent the Spread of Coronavirus in Product Delivery System” 2020 11th IEEE Annual Ubiquitous Computing, Electronics and Mobile Communication Conference, UEMCON 2020 9298108, pp. 0461-0466, 2020.
- [5] Zhao, X., Gao, Z., Zhang, Y., Chen, B.M. “A target tracking and positioning framework for video satellites based on SLAM” IEEE International Conference on Intelligent Robots and Systems 9341270, pp. 1887-1894, 2020.
- [6] Cherepanov, P.Y., Romanov, P.A., Zharikov, S.V., Mazhitov, T.R. “The control system for a mobile robot using Arduino mega 2560 with GPS and obstacle detection systems” Journal of Physics: Conference Series 1333(5), 052003, 2019.
- [7] Exposito Jimenez, V.J., Schwarzl, C., Martin, H. “Evaluation of an indoor localization system for a mobile robot” 2019 8th IEEE International Conference on Connected Vehicles and Expo, ICCVE 2019 – Proceedings 8965234, 2019.
- [8] Santos, E.R.S., Azpurua, H., Rezeck, P.A.F., (...), Freitas, G.M., Macharet, D.G. “Global localization of mobile robots using local position estimation in a geo tagged wireless node sensor network” Proceedings - 15th Latin American Robotics Symposium, 6th Brazilian Robotics Symposium and 9th Workshop on Robotics in Education, LARS/SBR/WRE 2018 8588524, pp. 8-12, 2018.
- [9] Chae, H., Choi, S., Yu, W., & Cho, J. “Autonomous navigation of mobile robot based on DGPS/INS sensor fusion by EKF in semi-outdoor structured environment” In 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (pp. 1222-1227). IEEE, (2010, October).
- [10] Hamid, M. H. A., Adom, A. H., Rahim, N. A., & Rahiman, M. H. F. “Navigation of mobile robot using Global Positioning System (GPS) and obstacle avoidance system with commanded loop daisy chaining application method” In 2009 5th International Colloquium on Signal Processing & Its Applications (pp. 176-181). IEEE, (2009, March).
- [11] Khan, M. Y. A., Mustafa, E., Nawaz, A., Saleem, N., & Illahi, U. “Sensor-fusion based navigation for mobile robot in outdoor environment” Mehran University Research Journal of Engineering and Technology, 38(1), 113-128, (2019).
- [12] Muggah Robert “The Rise of Citizen Security in Latin America and Caribbean” Alternative Pathways to Sustainable Development: Lessons from Latin America, (p. 291-322), (2017).
- [13] Instituto Nacional de Estadísticas e Informática “Estadísticas de Seguridad Ciudadana”, (2015, Enero – Junio)