Computación de Próxima Generación en el Contexto de Tendencias Emergentes en Inteligencia Artificial, Analítica Avanzada y Defensa Digital

María del Carmen Santiago Díaz

COMPUTACIÓN DE PRÓXIMA GENERACIÓN EN EL CONTEXTO DE TENDENCIAS EMERGENTES EN INTELIGENCIA ARTIFICIAL, ANALÍTICA AVANZADA Y DEFENSA DIGITAL

María del Carmen Santiago Díaz

Gustavo Trinidad Rubín Linares

Ana Claudia Zenteno Vázquez

Judith Pérez Marcial

(Editores**)**

María del Carmen Santiago Díaz

(Coordinadora)

María del Carmen Santiago Díaz, Gustavo Trinidad Rubín Linares, Ana Claudia Zenteno Vázquez, Judith Pérez Marcial
(editores BUAP)

María del Carmen Santiago Díaz
(coordinadora BUAP)

María del Carmen Santiago Díaz, Gustavo Trinidad Rubín Linares, Ana Claudia Zenteno Vázquez, Judith Pérez Marcial, Fernando Moreno Barbosa, José Hugo Barrón Zambrano, Salvador Eugenio Ayala Raggi, Raúl Castro Ortega, Gabriela Francisca Solís Magaña, Enrique González Amador, Luis Fernando Curi Quintal, Miguel Angel San Pablo Juárez, Gustavo Manuel Minquiz Xolo, Hiram Herrera Rivas, Javier Flores Méndez, Erik Zamora Gómez, Alfonso Martínez Cruz, Vittorio Zanella Palacios, Alejandro Peñuelas Angulo, Ricardo Alvarez González, Beatriz Alejandra Flores Rojas, Lauro Reyes, Victor Menendez Domínguez, Antonio Armando Aguileta Güemez, Elsa Rubio Espino, José Juan García Hernández, Darnes Vilariño Ayala, Ernest Cortes, Fabricio Otoniel Pérez Pérez, Gerardo Martínez Guzmán, Jesús Yaljá Montiel Pérez, Jonathan Axel Cruz Vázquez, José Miguel Hurtado Madrid, Mario Alejandro Vega Navarrete, Saulo Abraham Gante Díaz, Sergio Delgadillo Marín, Efrén Pablo Héctor González Videgaray, Katya Celina Del Toro Oblea, Georgina Eslava García, Oscar Gabriel Caballero Martínez, Adriana Dávila Santos, Federico Guillermo Bahena Gómez, José Antonio Coria Fernández, Ricardo Guzmán Fuentes, Christian Carlos Delgado Elizondo, Enrique Espinoza Loyola, Sergio Quiroz Almaráz.
(revisores)

Primera edición: 2025

ISBN: 978-607-8975-84-6

Montiel & Soriano Editores S.A. de C.V.

15 sur 1103-6 Col. Santiago Puebla, Pue.

BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

Rectora:
Dra. María Lilia Cedillo Ramírez

Secretario General:
Mtro. Damián Hernández Méndez

Vicerrector de Investigación y Estudios de Posgrado:
Dr. Ygnacio Martínez Laguna

Directora de la Facultad de Ciencias de la Computación:
M.I. María del Consuelo Molina García

Contenido

Prefacio……5

Implementación de un Robot Autónomo Evasor de Obstáculos para Navegación Interior y Mapeo Estimado con Python
Ricardo Álvarez González
Brian Manuel González Contreras
Alba Maribel Sánchez Gálvez
José Fermi Guerrero Castellanos……6

Sistema de Monitorización para el Invernadero de Cactáceas y Suculentas Helia Bravo-Hollis Utilizando Tecnología LoRa en Malla
Nicolás Quiroz Hernández
Lucero Montserrat Cuautle García
María Fernanda Agüero González
Carlos Emmanuel Bernal Estrella
Ana María Rodríguez Domínguez……16

Integración de Módulos en un Nanosatélite para Detección de Incendios Forestales
Ángel Castillo Muñoz
Ronaldo Rodríguez García
Nicolás Rosas Guerrero
María del Carmen Santiago Díaz
Ana Claudia Zenteno Vázquez
Gustavo Trinidad Rubín Linares……27

LoRa Performance Evaluation for Real-World IoT Applications
Berenice Cortes Cinto
Josefina Castañeda Camacho
Victor Rodolfo Gonzalez Diaz……37

Clasificación de la Polaridad en Opiniones Políticas mediante Modelos Preentrenados de Lenguaje
Angeles Belém Priego Sánchez……52

Caminemos Juntas
Cecilia América Barranco Herrera
María del Carmen Santiago Díaz
Ana Claudia Zenteno Vázquez
Gustavo Trinidad Rubín Linares
Raúl Antonio Aguilar Vera
Abraham Sánchez López ……65

Integración de Visión Artificial a un Brazo Robótico de 3 Grados
Angel Cruz Jimenez
Abel Alejandro Rubín Alvarado
Jonathan Axel Cruz Vazquez
Juan Humberto Sossa Azuela ……*75*

Calibración de Crosstalk para Sensores ToF Multi-zona con Cubiertas Protectoras
Javier Martínez Fernández
Nicolás Quiroz Hernández
Ana María Rodríguez Domínguez
Roberto Olmos Pimentel……*89*

Desarrollo de una red inteligente de monitoreo de Calidad del Aire para la Ciudad de Puebla
Ángel José González Pineda
Mariano Larios Gómez
Marco Antonio Mora Ramírez……*101*

Percepción Espacial en Tiempo Real para la Navegación Autónoma de un Robot Diferencial bajo una Ley iLQR
Ignacio Gómez Juárez
Emilio Zdenko Abarca Cruz
Abel Alejandro Rubín Alvarado
Saulo Abraham Gante Díaz
Juan Humberto Sossa Azuela ……*108*

Análisis y Visualización del Impacto del Estado Emocional en el Rendimiento Académico mediante Herramientas en la Nube y Algoritmos Predictivos
Roberto Carlos Silva Lozano
Judith Pérez Marcial
Beatriz Flores Rojas
María del Carmen Santiago Díaz
Ana Claudia Zenteno Vázquez
Gustavo Trinidad Rubín Linares……*123*

Sistema de Gestión de Vehículos basada en Visión por Computadora para el Reconocimiento de Placas
Joselyne Janette Flores Emilio
Ana Luisa Ballinas Hernández
Judith Pérez Marcial
María Claudia Denicia Carral
Hector Medina Cruz ……*133*

Prefacio

Hoy concebimos la computación de una manera diferente gracias al gran desarrollo tecnológico que se ha dado en los últimos años. Aquello que antes considerábamos innovación hoy se vuelve rutinario, permitiendo que los avances que parecían distantes se integren rápida a nuestras actividades cotidianas. La Inteligencia Artificial (IA), la Analítica Avanzada y la Defensa Digital han dejado de ser áreas especializadas para convertirse en pilares fundamentales de la sociedad contemporánea.

Agentes autónomos, sistemas generativos multimodales, sistemas que procesan grandes cantidades de datos en tiempo real y los modelos conversacionales, resultan ser ya fácilmente accesibles para millones de usuarios bajo un procesamiento cada vez más eficientes mediante modelos y sistemas híbridos simbólico-neuronales. En paralelo a la evolucionan de estas tecnologías surgen proyectos comunitarios y desarrollos científicos que enriquecen el ecosistema tecnológico con una velocidad sin precedentes. Y es que la computación contemporánea no solo se refiere al aprendizaje profundo. Hoy en día, tecnología emergente como la computación cuántica se está integrando en soluciones a problemáticas que ya desafían incluso a supercomputadoras, el edge computing indispensable para sistemas inteligentes distribuidos; y la analítica autónoma que permite a los sistemas tomar decisiones complejas sin supervisión humana constante. Adicionalmente, la ciberseguridad no solo reacciona ante amenazas mediante la defensa digital sino está incorporando el uso de IA, análisis de comportamiento, detección temprana y automatización estratégica.

En este nuevo panorama, la formación continua resulta ser una necesidad imperante. Las tecnologías de próxima generación requieren además de curiosidad y creatividad, un entendimiento importante de sus fundamentos, riesgos, límites e implicaciones éticas. Desde la academia y la investigación, asumimos el reto de guiar este proceso, construir conocimiento, sensibilizar a la reflexión crítica y atender a la construcción de soluciones que respondan a los desafíos actuales y futuros.

La construcción de este libro cumple con ese objetivo. Reúnir trabajos que exploran las dimensiones más relevantes de la computación emergente: Inteligencia Artificial aplicada, modelos y arquitecturas avanzadas, analítica predictiva y prescriptiva, tecnologías cuánticas, ciberdefensa inteligente y perspectivas innovadoras que dialogan entre sí. Cada capítulo ofrece una mirada actualizada, reflexiva y propositiva sobre los retos y oportunidades que presentan estas áreas en constante transformación. Hemos cuidado la calidad del libro desde la aceptación de trabajos que cumplan con el rigor, actualidad y claridad en cada contribución. Esperamos que esta obra sirva como guía, inspiración y referencia para estudiantes, docentes, investigadores y profesionales que buscan asimilar y explotar tecnologías que definirán la próxima generación del conocimiento computacional.

Invitamos al lector a recorrer estas páginas con la curiosidad que exige la innovación y con el espíritu crítico que demanda la ciencia. La computación del futuro ya está aquí; lo que hagamos hoy será determinante para imaginar y construir lo que sigue.

María del Carmen Santiago Díaz
Gustavo Trinidad Rubín Linares

Implementación de un Robot Autónomo Evasor de Obstáculos para Navegación Interior y Mapeo Estimado con Python

Implementation of an Autonomous Obstacle Avoidance Robot for Indoor Navigation and Estimated Mapping with Python

Ricardo Álvarez González[1], Brian Manuel González Contreras[2], Alba Maribel Sánchez Gálvez[3], José Fermi Guerrero Castellanos[1],

[1] Facultad de Ciencias de la Electrónica, Benemérita Universidad Autónoma de Puebla, Av. San Claudio y 18 sur Col. San Manuel. C.P. 72570. Puebla, Puebla, México.

[2]Facultad de Ciencias Básicas de Ingeniería y Tecnología, Universidad Autónoma de Tlaxcala, Carretera Apizaquito S/N, San Luis Apizaquito, C.P. 90401, Apizaco, Tlaxcala, México.

[3] Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla, Av. San Claudio y 14 sur Col. San Manuel. C.P. 72570. Puebla, Puebla, México.

{ricardo.alvarez, alba.sanchez, fermi.guerrero}@correo.buap.mx, brianmanuel.gonzalez@uatx.mx

Abstract

The development of a frugal prototype of an autonomous mobile robot with differential drive is presented, designed to operate in indoor environments and avoid obstacles through autonomous navigation. The system is built with low-cost commercial components and transmits data from its infrared incremental encoders via Bluetooth. This information is used to estimate the path traveled and calculate the area covered by the robot, using a Python program. Experiments show a relative accuracy of 77% in estimating the distance traveled.

Resumen

Se presenta el desarrollo de un prototipo frugal de robot móvil autónomo con tracción diferencial, diseñado para operar en entornos interiores y evitar obstáculos mediante navegación autónoma. El sistema está construido con componentes comerciales de bajo costo y transmite, vía Bluetooth, los datos de sus codificadores incrementales infrarrojos. Esta información se emplea para estimar la trayectoria recorrida y calcular el área cubierta por el robot, utilizando un programa de desarrollado en Python. Los experimentos realizados muestran una exactitud relativa del 77 % en la estimación de la distancia recorrida.

Keywords and phrases: Robot Móvil Autónomo, Tracción Diferencial, ESP32, Python.

1 Introducción

Los robots móviles autónomos con ruedas (RMAR) abordan una variedad de problemáticas en distintos sectores (industrial, agrícola, tareas de rescate, exploración remota, etc), principalmente

aquellas relacionadas con tareas repetitivas, riesgosas, ineficiencias operativas y limitaciones humanas. Por lo que la presencia y demanda de robots móviles autónomos se ha incrementado notablemente en los últimos años [1], [2]. La localización es uno de los problemas más difíciles y costosos de los RMAR [3] por lo que, para abonar en la solución de este problema, se presenta el prototipo frugal de un RMAR y la estimación de su trayectoria recorrida mediante una gráfica realizada en Python.

2 Marco Teórico y Estado del Arte

A continuación, se definen conceptos importantes para la comprensión de este artículo.

2.1 Localización relativa

La localización relativa se realiza mediante navegación estimada, es decir, por la medida del movimiento entre dos ubicaciones del RMAR, que se realiza repetidamente mientras el robot se mueve, y estas medidas se van acumulando para estimar la distancia recorrida desde el punto de partida. Debido a que las estimaciones individuales de las posiciones locales no son exactas, los errores se acumulan y el error absoluto del movimiento total se acumula con la distancia recorrida. Para un RMAR, el método de navegación estimada se denomina "odometría" y está basado en los datos obtenidos de los codificadores incrementales de las ruedas. La suposición básica de la odometría radica en que las revoluciones de las ruedas se pueden transformar en desplazamientos lineales relativos al piso, la cual no es muy válida debido al deslizamiento de las ruedas y otras causas. [4]

2.2 Análisis cinemático de la localización relativa

El estado del RMAR, es decir, su posición y orientación en el instante k se determina por la ecuación 1:

$$X(k)=[x(k),y(k), \Theta(k)]^T \quad (1)$$

La localización basada en odometría, consiste en estimar X(k+1) en el instante k+1, y los incrementos lineales y angulares $\Delta l(k)$ y $\Delta \Theta(k)$ respectivamente. En donde:

$$\Delta x(k)=x(k+1)-x(k) \quad (2)$$

$$\Delta y(k)=y(k+1)-y(k) \quad (3)$$

$$\Delta l(k)=[\ \Delta x^2(k)+ \Delta y^2(k)]^{1/2} \quad (4)$$

$$\Delta \Theta (k)= \Theta (k+1)- \Theta (k) \quad (5)$$

Si suponemos que $\Delta \Theta (k)$ es muy pequeño, se pueden hacer las aproximaciones:

$$X(k+1)=x(k)+ \Delta l(k)\cos(\Theta (k) + \Delta \Theta (k)/2) \quad (6)$$

$$y(k+1)=y(k)+ \Delta l(k)\text{sen}(\Theta (k) + \Delta \Theta (k)/2) \quad (7)$$

$$\Theta (k+1) = \Theta (k)+ \Delta \Theta (k) \quad (8)$$

Para determinar los incrementos de la posición y el ángulo a partir de los valores de los codificadores, debemos de usar las ecuaciones cinemáticas del RMAR de tracción diferencial, para lo cual son suficientes dos codificadores ópticos que proporcionan los incrementos $\Delta l_1(k)$ y $\Delta l_2(k)$ correspondientes a las ruedas izquierda y derecha. Si 2ª es la distancia entre las ruedas, y R(k) el radio de curvatura instantánea, tenemos:

$$\Delta l_1(k)=[R(k)-a]\ \Delta\ \Theta\ (k) \tag{9}$$

$$\Delta l_2(k)=[R(k)+a]\ \Delta\ \Theta\ (k) \tag{10}$$

De las expresiones anteriores se obtiene que:

$$\Delta\theta(k) = \frac{\Delta l_2(k)-\Delta l_1(k)}{2a} \tag{12}$$

$$\Delta l(k) = \frac{\Delta l_1(k)+\Delta l_2(k)}{2} \tag{13}$$

De tal manera que los incrementos de posición y ángulo están expresados en término de los valores de incremento de los codificadores. [4]

2.3 Cálculo del área de un polígono mediante la fórmula de Shoelace.

Para calcular el área que recorre el RMAR, se puede utilizar una fórmula general para calcular el área de un polígono a partir de sus vértices: si $p_1=(x_1, y_1)$, $p_2=(x_2, y_2)$, $p_3=(x_3, y_3)$,…, $p_n=(x_n, y_n)$ de tal manera que p_i y p_{i+1} son vértices adyacentes en la frontera de un polígono, y los primeros y últimos vértices son los mismos, es decir $p_1=p_n$ se calcula el área usando la fórmula de Shoelace: [5]

$$A = \frac{1}{2}\left|\sum_{i=1}^{n-1}(x_i y_{i+1} - x_{i+1}y_i)\right| \tag{14}$$

2.4 Trabajos relacionados

En [6] presentan un método aproximado de auto localización de un RMAR basado en características visuales del ambiente, que el robot adquiere mediante una cámara que captura imágenes a color, de las que se extraen características para entrenar una red neuronal artificial de 4 capas. En [7] presentan el problema de localización y la trayectoria que debe de seguir un RMAR de tracción diferencial, a partir de los datos de los codificadores incrementales de sus ruedas de 48 pulsos por revolución. Realizan el control en una PC usando el software Matlab/Simulink, muestran gráficamente la trayectoria del robot; usan control proporcional integral y un estimador mediante un filtro de Kalman, apoyándose en el modelo cinemático del robot. En [3] presentan un controlador para el seguimiento de trayectorias de robot móviles no- holonómicos utilizando retroalimentación visual sin medición directa de su posición, estiman la posición global del RMAR usando características visuales por un sistema de visión y miden su orientación y velocidad por odometría y sensores inerciales como brújula e IMU.

Como se puede observar en los trabajos relacionados mencionados, la localización y seguimiento de trayectorias de un RMAR es un tema muy importante y recurrente, que se puede abordar con diversas técnicas y la exactitud alcanzada depende directamente de la complejidad de los equipos e instrumentos utilizados. El trabajo que se reporta en este documento consiste en un sistema frugal

para graficar la trayectoria estimada de un RMAR evasor de obstáculos, que envía periódicamente por bluetooth la información de sus codificadores incrementales, usando un SoC ESP32 como computadora de abordo. La información se recibe en un teléfono inteligente y mediante un script en Python se grafica la trayectoria estimada recorrida por el robot.

3 Arquitectura del Sistema

Para realizar la localización relativa del RMAR, se desarrolló un sistema compuesto básicamente por el prototipo de robot que navega libremente evitando obstáculos y por el script en Python que procesa los datos enviados por el RMAR para trazar su trayectoria, por lo que a continuación se describen cada una de estas secciones.

3.1 Descripción del prototipo de RMAR

En la figura 1 se muestra el RMAR que realiza navegación libre evadiendo obstáculos y está construido con componentes comerciales de fácil adquisición los cuales se especifican en la tabla 1. La tracción del robot se obtiene mediante un par de moto-reductores que están conectados a cada una de las ruedas, con sendos codificadores ópticos, obedeciendo a una estructura diferencial.

Tabla 1. Componentes del RMAR.

Percepción del entorno	Mediante tres sensores ultrasónicos HC-SR04
Microcontrolador	ESP32-WROOM32
Tracción	Dos motorreductores
Retroalimentación de movimiento	Discos codificadores de 20 pulsos por revolución con sensores infrarrojos Sensores ultrasónicos HC-SR04
Etapa de potencia	Puente H L298N de dos canales
Alimentación	Batería LiPO de 7.5 v

Como se aprecia en la figura 1, se utilizan tres sensores ultrasónicos HC-SR04 para percibir las distancias que se tienen hacia la izquierda, frente y derecha del robot, para tomar la decisión de su desplazamiento al navegar libremente evitando obstáculos. La computadora de abordo es el sistema en chip (SoC) ESP32, desarrollado por la compañía *Espressif Systems*, el cual está basado en una arquitectura de 32 bits, mediante el procesador de doble núcleo *Xtensa LX6* que puede funcionar hasta 240 MHz. Tiene 520 KB de memoria SRAM y 4 MB de memoria Flash externa. Tiene conectividad inalámbrica WiFi, así como Bluetooth convencional y de baja energía (BLE), que le permite al RMAR enviar a un teléfono inteligente vinculado a él, los datos de los pulsos de los discos codificadores conectados a los ejes de los motores para percibir su rotación, así como los comandos para iniciar y detener su funcionamiento, como se observa en la figura 2.

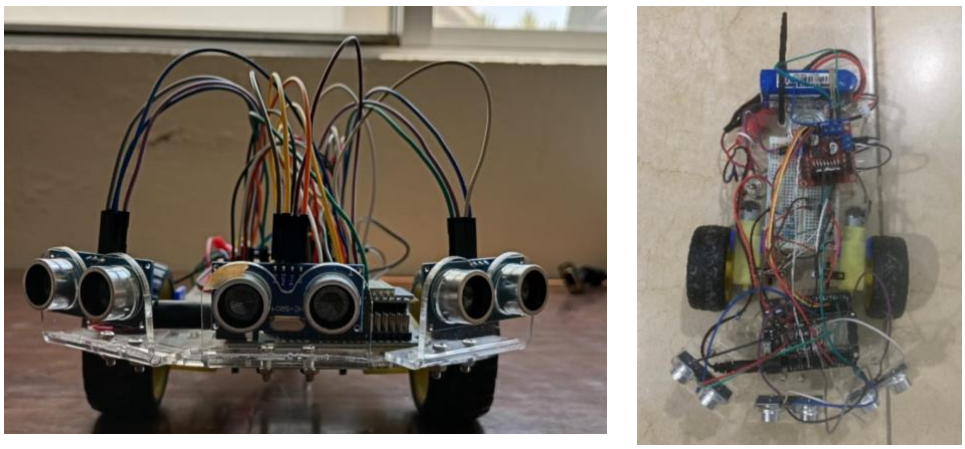
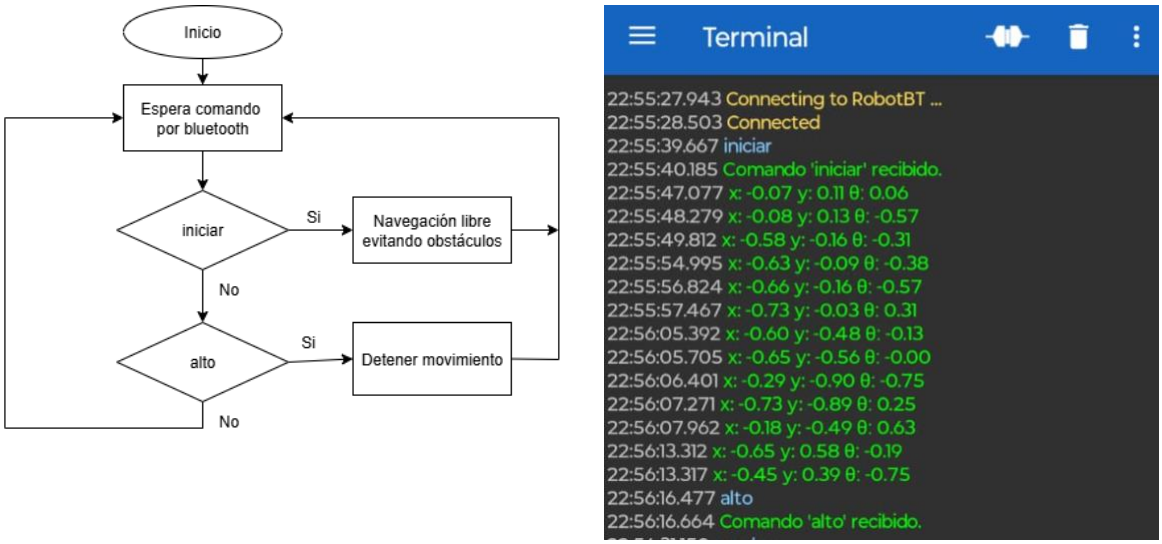


Figura 1. RMAR con tres sensores HC-SR04.



a) b)

Figura 2. a) Ciclo infinito del RMAR, b) Captura de pantalla del teléfono inteligente con el que se establece comunicación con el RMAR.

Con los codificadores de las ruedas se realiza la localización relativa del robot, a partir de sus parámetros cinemáticos mediante odometría de acuerdo con la ecuación 1, como se observa en la figura 3

```
// Parámetros cinemáticos
float r = 0.03; // radio de rueda en metros
float L = 0.15; // distancia entre ruedas en metros
int pulsosPorVuelta = 20;
float distancia_pulso = 2 * PI * r / pulsosPorVuelta;

// Pose del robot
float x = 0, y = 0, theta = 0;
```

Figura 3. Parámetros cinemáticos del RMA.

Se obtienen los incrementos lineales y angulares de acuerdo con las ecuaciones 2 y 3, para finalmente obtener la localización relativa del RMAR con los valores aproximados de x,y, Θ según como indican las ecuaciones 6,7 y 8. La generación de estos parámetros se observan en el fragmento del código fuente mostrado en la figura 4

```
// Actualización de odometría
long currentLeft = encoderLeft;
long currentRight = encoderRight;
long deltaLeft = currentLeft - prevLeft;
long deltaRight = currentRight - prevRight;
prevLeft = currentLeft;
prevRight = currentRight;

float dL = deltaLeft * distancia_pulso;
float dR = deltaRight * distancia_pulso;
float dC = (dL + dR) / 2.0;
float dTheta = (dR - dL) / L;

theta += dTheta;
x += dC * cos(theta);
y += dC * sin(theta);
```

Figura 4. Estimación de la posición del RMAR mediante odometría.

Al recibir el comando “iniciar” por bluetooth, el RMA realiza la tarea de navegación libre evadiendo obstáculos de acuerdo con el algoritmo mostrado en la figura 5, inicialmente cada sensor ultrasónico mide la distancia correspondiente, denominadas distancia izquierda *d_L*, distancia frontal *d_F* y distancia a la derecha *d_R*. El robot se moverá hacia al frente si es que existe un espacio mayor al umbral, el cual se fijó en 20 cm. Si no se cumple esta condición, se detiene el robot durante 100 ms y verifica si la distancia medida por el sensor de la derecha es mayor que el de la izquierda, pero comprobando a su vez si esta distancia es mayor que el umbral, si estas condiciones se cumplen, es porque hay espacio para que el robot se mueva a la derecha, por lo tanto, el robot gira hacia la derecha. En caso contrario, se verifica si la distancia medida por el sensor de la izquierda es mayor que el umbral, entonces girará hacia la izquierda. En caso de no satisfacerse estas condiciones, el robot se desplaza durante 300 ms en reversa y a continuación efectúa una vuelta a la derecha. Este algoritmo de desplazamiento del RMA le permite moverse en pasillos evitando obstáculos, e incluso desplazarse por laberintos.

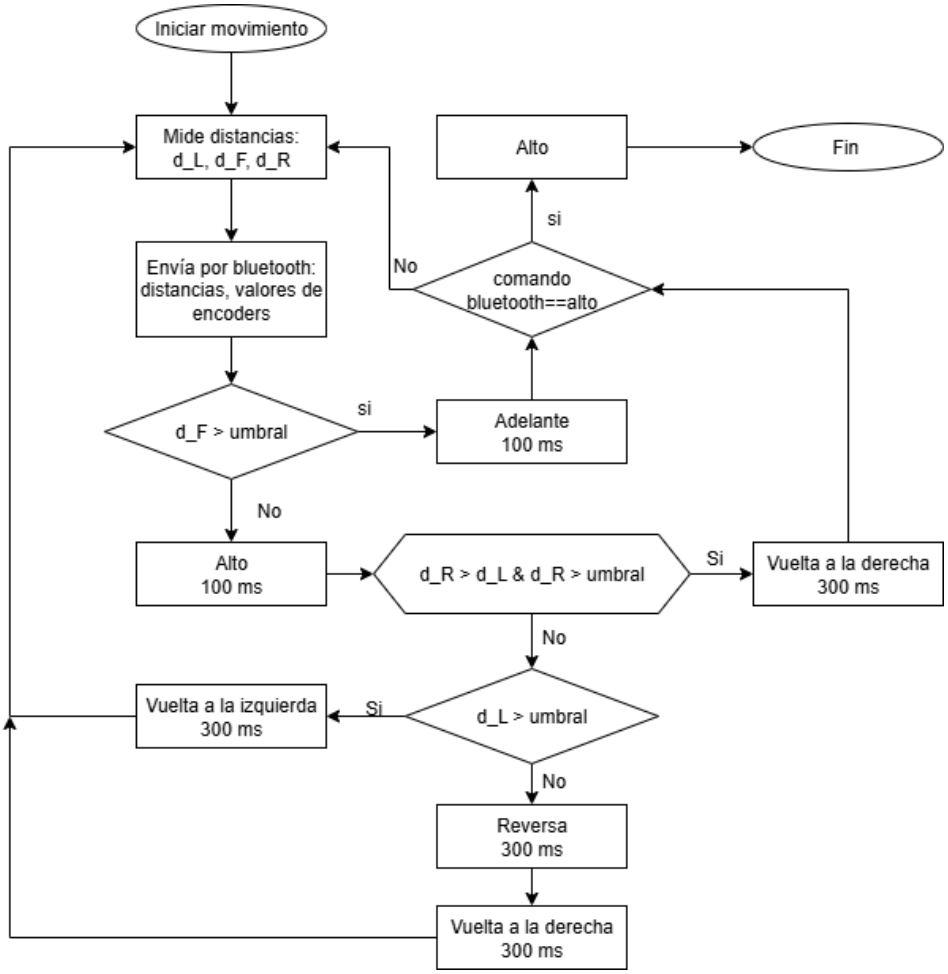


Figura 5. Diagrama de flujo de la rutina de navegación libre evitando obstáculos.

El RMAR disminuye proporcionalmente su velocidad conforme se va acercando a algún obstáculo, como se observa en la función *velocidadProporcional* mostrada en la figura 6. En donde *d_max* y *d_min* corresponden a 50 cm y 10 cm respectivamente.

```
int velocidadProporcional(float distancia) {
  if (distancia >= d_max) return vel_max;
  if (distancia <= d_min) return vel_min;
  float k = (vel_max - vel_min) / (d_max - d_min);
  return vel_min + k * (distancia - d_min);
}
```

Figura 6. Función proporcional de velocidad del RMAR

3.2 Descripción del algoritmo para graficar la trayectoria del RMAR

En la figura 7 se muestra el diagrama de bloques del script en Python que, a partir de los datos transmitidos mediante Bluetooth por el RMAR, es el encargado de graficar la trayectoria recorrida

a partir de los datos crudos generados por el robot, que inicia importando las funciones de biblioteca:

- *re*, para buscar datos en el texto a partir de expresiones regulares (regex)
- *numpy*, para las operaciones numéricas y vectorizadas
- *matpotlib,* para graficar la trayectoria del área recorrida por el RMAR

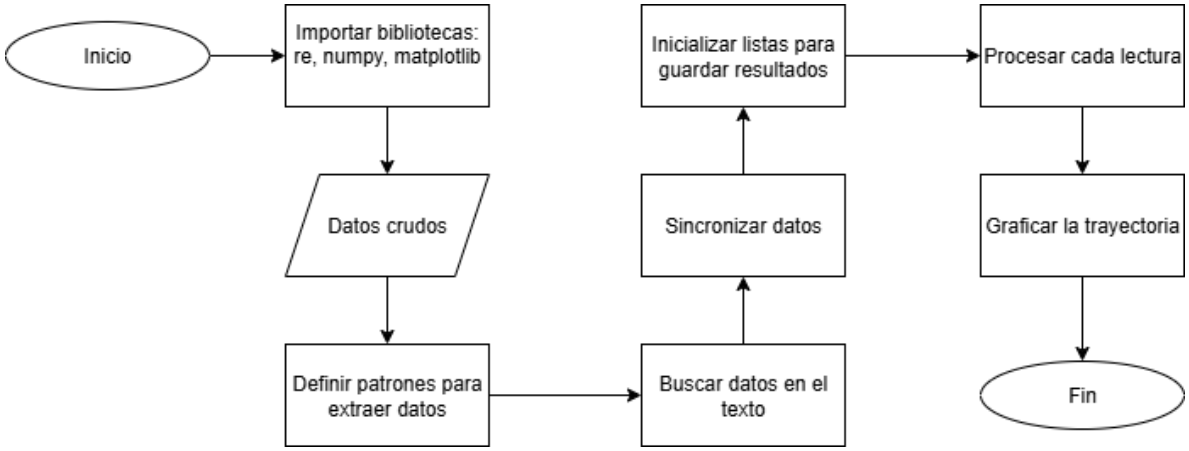


Figura 7. Diagrama de bloques del programa de procesamiento de los datos de posición del RMAR para graficar la trayectoria.

Los datos crudos transmitidos mediante bluetooth por el RMAR, están integrados por renglones como: *19:14:20.291 x: -18.10 y: 5.48 ϑ: 122.02*, que contienen la hora de transmisión y las coordenadas x,y,Θ en cm y radianes respectivamente de la posición relativa, solamente se usarán x,y para estimar el área. Mediante una expresión regular se buscan todas las líneas que contengan x,y, y Θ. Posteriormente se guardan los puntos como tuplas (x,y) en una lista. Se verifica si la trayectoria está cerrada, comparando el primer punto con el último, como se explicó en la sección 2.3, si están separados más de 5 cm, se considera que el RMAR no cerró el contorno, así que se agrega el punto inicial al final para cerrar artificialmente la trayectoria y calcular el área con la ecuación 14. Finalmente se calcula el área en cm^2, por lo que es necesario dividir el resultado entre 10 000 para convertirla a m^2 y se grafica la trayectoria del robot con línea azul.

Figura 8. Área de trabajo experimental de 1.2 m x 2.3 m.

4 Resultados

En la figura 8 se observa el espacio de trabajo de 1.2 m x 2.3 m, que ha sido confinado con la finalidad de mantener el desplazamiento del RMAR dentro del mismo y así poder cuantificar la exactitud de la estimación de la trayectoria reportada. En la mitad izquierda de la figura 9 se muestra la trayectoria del RMAR en navegación libre en el espacio de trabajo de la figura 8 sin obstáculo, cada punto representa la posición *x,y* reportada por el RMAR, en donde se aprecian puntos muy juntos, es debido a que al encontrarse cerca del límite del espacio experimental el RMAR disminuye gradualmente su velocidad para modificar su ruta, de acuerdo con el espacio libre disponible. En el eje de las abscisas observamos un desplazamiento aproximado del RMA de 1.2 m y en el de las ordenadas de 1 m, por trigonometría podemos calcular que la diagonal de la trayectoria corresponde a 1.56 m debido a que el RMA se detiene a 0.2m de la barrera del espacio de trabajo, estimamos entonces que la distancia accesible al RMAR es de 1.9 m, usando la ecuación 15 obtenemos una exactitud relativa del 82 %.

$$Exactitud = 1 - \frac{|valor\ estimado - valor\ real|}{valor\ real} \tag{15}$$

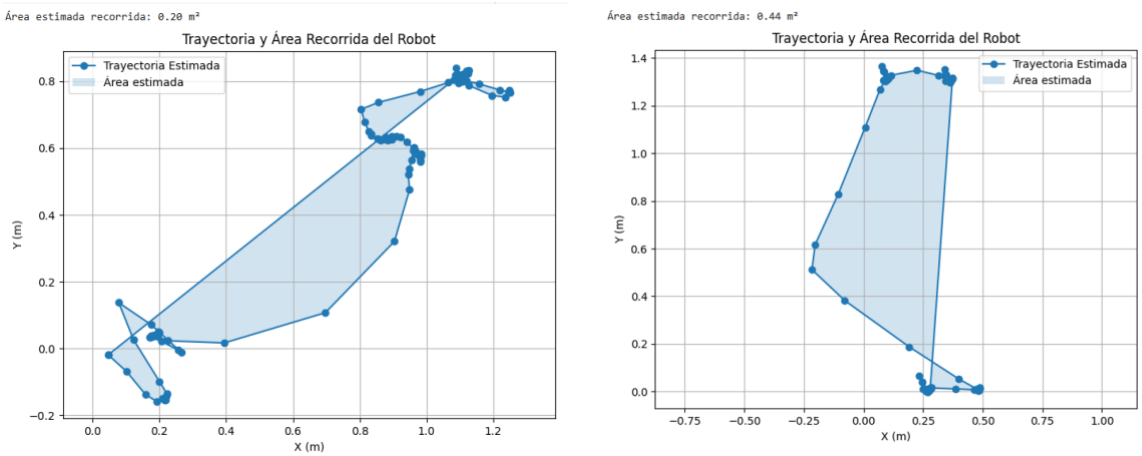


Figura 9. Gráfica de dos trayectorias recorridas por el RMAR.

En la mitad derecha de la figura 9, mostramos la trayectoria de otro experimento en donde se colocó un pequeño obstáculo en el centro del espacio de trabajo y se estima que el RMAR se desplazó de un extremo a otro del espacio del trabajo una distancia de 1.38 m, considerando que la distancia alcanzable por el Robot es de 1.9 m, aplicando nuevamente la ecuación 15, obtenemos una exactitud del 72 %. El promedio de la exactitud relativa obtenida en estos experimentos es del 77%.

5 Conclusiones y Trabajo a Futuro

Se mostró un prototipo funcional construido con componentes comerciales de bajo costo y un script en Python, para enfrentar uno de los retos más trascendentes de la robótica móvil: conocer la posición del RMAR. La exactitud es baja debido a que los codificadores ópticos son de baja resolución: 20 pulsos por revolución, además de los errores acumulados de los mismos para obtener la localización relativa, lo cual coincide con lo establecido por la teoría expuesta en la sección 2.1, pero considerando que se trató de un proyecto frugal ya que no se utilizaron sensores complejos que implicarían mayor consumo de energía (impactando directamente en la duración de la batería) ni software sofisticado que requiere el pago de licencias onerosas, los resultados son aceptables y motivan a realizar mejoras al prototipo.

En un trabajo futuro se fusionará la odometría con los datos de aceleración y velocidad angular proporcionados por una IMU, para obtener una estimación más precisa de la trayectoria del RMAR, así como también se realizará el montaje de los componentes en un circuito impreso, para eliminar los problemas de falso contacto que pueden ocasionar un comportamiento errático del prototipo.

Referencias

[1] H. Jian, “Path planning algorithm for mobile robots,” *Trans. Comput. Sci. Intell. Syst. Res.*, 2024.

[2] A. Loganathan and N. S. Ahmad, “A systematic review on recent advances in autonomous mobile robot navigation,” *Eng. Sci. Technol., Int. J.*, 2023. [Online]. Available: https://doi.org/10.1016/j.jestch.2023.101343

[3] K. Wang, Y. Liu, and L. Li, “Visual servoing trajectory tracking of nonholonomic mobile robots without direct position measurement,” *IEEE Trans. Robot.*, vol. 30, no. 4, pp. 1021–1035, Aug. 2014.

[4] S. G. Tzafestas, *Introduction to Mobile Robot Control*. Athens, Greece: Elsevier, 2014.

[5] A. Laaksonen, *Guide to Competitive Programming: Learning and Improving Algorithms Through Contests*, 3rd ed. Springer, 2024.

[6] F. A. Jafar, Y. Suzuki, K. Yokota, and T. Matsuoaka, “Autonomous mobile robot self-localization based on environmental visual features,” in *Proc. 4th Int. Conf. Autonomous Robots and Agents*, Wellington, New Zealand, 2009.

[7] S. Vera González, L. Chuquimarca Jiménez, and D. Plaza, “Kalman filter applied to a differential robot,” in *Proc. 1st Int. Conf. Circuits, Power and Intell. Syst.*, IEEE, 2023.

Sistema de Monitorización para el invernadero de cactáceas y suculentas Helia Bravo-Hollis utilizando tecnología LoRa en Malla

Monitoring System for the Helia Bravo-Hollis Cactus and Succulent Greenhouse using LoRa Technology on a Mesh

Nicolás Quiroz Hernández[1], Lucero Montserrat Cuautle García[2], María Fernanda Agüero González[1], Carlos Bernal Estrella[1], Ana María Rodríguez Domínguez[1]

[1]Facultad de Ciencias de la Electrónica, Benemérita Universidad Autónoma de Puebla, Puebla, México

[2]Facultad de Ciencias Biológicas, Benemérita Universidad Autónoma de Puebla, Puebla, México

{maría.aguerogo, carlos.bernale}@alumno.buap.mx, {nicolas.quirozh, lucero.cuautleg, anamaria.rodriguez}@correo.buap.mx

Abstract

This paper presents the development and implementation of an environmental monitoring system for the "Helia Bravo-Hollis" cactus and succulent greenhouse, located at the Benemérita Universidad Autónoma de Puebla (BUAP). The main objective is to automate the monitoring of critical variables—such as temperature, relative humidity, and substrate moisture—through a distributed sensor network based on LoRa technology using a mesh topology. The system consists of distributed sensor nodes, each equipped with DHT11 and FC-28 sensors, an ESP32-S3 microcontroller, and a LoRa RYLR998 communication module. The data collected by the nodes is transmitted through the mesh network to a gateway node, which then sends the information to a cloud database (Firebase) using the HTTPS protocol. Coverage tests were carried out within Ciudad Universitaria (BUAP), revealing that although LoRa modules offer a theoretical range of up to 10 km in line-of-sight conditions, their actual performance significantly decreases in urban environments with multiple structural obstacles. The tests showed successful transmission at up to 234 meters with two buildings in between, but signal failures when more than three obstacles were present. The system demonstrated its feasibility for reliable real-time monitoring inside the greenhouse and establishes the groundwork for future automation using irrigation control systems. There are plans to scale the solution to the BUAP Botanical Garden, where the extended coverage of mesh networks could be more effectively utilized.

Resumen

El presente trabajo describe el desarrollo e implementación de un sistema de monitorización ambiental para el invernadero de cactáceas y suculentas *"Helia Bravo-Hollis"*, ubicado en la Benemérita Universidad Autónoma de Puebla (BUAP). El objetivo principal es automatizar el registro de variables críticas —como temperatura, humedad relativa y humedad del sustrato— mediante una red de sensores basada en tecnología LoRa con topología de malla.

El sistema se compone de nodos sensoriales distribuidos, cada uno equipado con sensores DHT11 y FC-28, un microcontrolador ESP32-S3 y un módulo de comunicación LoRa RYLR998. Los datos recolectados son transmitidos entre nodos mediante algoritmos de enrutamiento y retransmisión hasta llegar a un nodo *gateway*. Este, a su vez, envía la información a una base de datos en la nube (Firebase) a través del protocolo HTTPS. Se realizaron pruebas de alcance dentro de Ciudad Universitaria (BUAP). Aunque los módulos LoRa ofrecen un rango teórico de hasta 10 km en línea de vista, el rendimiento real disminuyó en entornos urbanos con múltiples obstáculos. Los experimentos mostraron transmisión exitosa hasta 234 metros con dos edificios intermedios, pero se presentaron fallas cuando existieron más de tres barreras.

El sistema demostró su viabilidad para un monitoreo confiable en tiempo real dentro del invernadero y sienta las bases para su futura automatización mediante controladores de riego. Además, se plantea su escalamiento al Jardín Botánico de la BUAP, aprovechando la cobertura extendida que permiten las redes en malla. Desde la perspectiva de las ciencias de la computación, este proyecto implementa una arquitectura de software distribuido. La lógica de programación de los nodos sensoriales se enmarca en el campo de los sistemas embebidos, donde el microcontrolador ESP32-S3 ejecuta tareas específicas de adquisición, procesamiento y transmisión de datos. La comunicación entre nodos se gestiona mediante la red en malla LoRa, que requiere algoritmos de enrutamiento y retransmisión para garantizar la conectividad. En el nodo *gateway* se integran conceptos de desarrollo de software para redes y web, ya que este procesa los datos recibidos y establece una comunicación segura con la nube mediante HTTPS. Finalmente, la integración con Firebase aplica principios de computación en la nube y gestión de datos, asegurando el almacenamiento persistente, la consulta y la futura visualización en tiempo real de la información recolectada.

Keywords and phrases: LoRa, Malla, ESP32, DHT11, Invernadero.

1 Introducción

Las suculentas mexicanas conforman un grupo botánico diverso y fascinante que ha evolucionado en los ecosistemas desérticos y semiáridos del país. Su principal característica adaptativa es la capacidad de almacenar agua en sus tejidos, lo que les permite sobrevivir en condiciones de sequía extrema y suelos pobres [1]. Este rasgo fisiológico, junto con su alta especialización ecológica, las convierte en elementos clave de la biodiversidad vegetal mexicana, muchas veces asociadas a zonas de alto endemismo y fragilidad ambiental [2].

En este contexto, la Colección de Cactáceas y Suculentas “Helia Bravo-Hollis” de la Benemérita Universidad Autónoma de Puebla (BUAP) representa un patrimonio científico, ecológico y cultural de gran relevancia. Su importancia radica en la diversidad de especies que alberga —incluyendo especies endémicas y en peligro de extinción—, así como en su función como espacio para la investigación, conservación y educación ambiental enfocada en la flora xerófila de México.

Sin embargo, durante la contingencia sanitaria provocada por la pandemia de SARS-CoV-2, el acceso a las instalaciones fue restringido, lo que, sumado a otros factores administrativos y operativos,

derivó en el deterioro de las condiciones del invernadero y en la falta de mantenimiento especializado de la colección. Actualmente, el invernadero “Helia Bravo-Hollis” opera como una Unidad de Manejo para la Conservación de la Vida Silvestre (UMA), figura legal y técnica establecida por la Ley General de Vida Silvestre, cuyo objetivo es fomentar el aprovechamiento sustentable, la conservación de especies y hábitats silvestres, y el desarrollo de actividades productivas, educativas y turísticas compatibles con el entorno natural [3].

Ante este panorama, el presente trabajo aborda la modernización tecnológica para la automatización de la monitorización fitosanitario, y la implementación de tecnologías emergentes como el Internet de las Cosas (IoT) [4, 5].

La integración de sensores ambientales, redes de comunicación y sistemas de almacenamiento en la nube permitirá registrar y analizar en tiempo real variables críticas para la conservación, tales como temperatura, humedad relativa, intensidad lumínica, pH del sustrato y conductividad eléctrica. Esta infraestructura tecnológica no solo optimizará la toma de decisiones técnicas para el cuidado de las especies, sino que también consolidará una plataforma moderna para la investigación aplicada, la educación ambiental y la gestión sustentable de cactáceas y suculentas en riesgo [6][7].

Este artículo presenta el desarrollo de un sistema de monitorización ambiental enfocado en variables como humedad, temperatura y condiciones del sustrato en un invernadero de cactáceas y suculentas. El sistema está basado en una red en malla (mesh) utilizando tecnología LoRa (Long Range), que se caracteriza por su bajo consumo energético, gran alcance y robustez frente a interferencias. A diferencia de otras redes inalámbricas, una red LoRa Mesh permite que los nodos se comuniquen entre sí, reenviando información hasta llegar a un nodo con acceso a internet (puerta de enlace), extendiendo así la cobertura sin necesidad de infraestructura costosa ni redes móviles [8]. En la figura 1 se muestra el invernadero de cactáceas y suculentas del invernadero Helia Bravo-Hollis, la figura 1a muestra que se tienen cetáceas en macetas individuales y en figura 1b muestra recipientes donde se tienen varias cetáceas o suculentas. Para la monitorización implica varios retes tecnológicos para comunicar los diferentes nodos de monitorización.

a) b)

Figura 1. Invernadero Helia Bravo-Hollis, BUAP, a) Macetas individuales y b) maceteros compartidos.

2 Marco Teórico

Para la monitorización de las condiciones ambientales y edáficas del invernadero, se propone el uso de nodos sensoriales distribuidos, capaces de medir variables clave como la humedad del sustrato, la temperatura y la humedad relativa del aire, la cantidad de luz, etc. Esta implementación es especialmente importante en invernaderos dedicados al cultivo de cactáceas y suculentas, ya que estas especies son altamente sensibles a variaciones hídricas y térmicas [9].

En una segunda etapa del proyecto, se plantea no solo monitorear, sino también controlar activamente variables críticas como la humedad del sustrato y las condiciones ambientales internas del invernadero (temperatura y humedad relativa), mediante sistemas automatizados. Como se observa en la figura 1a, se requiere al menos un nodo por maceta para obtener datos precisos e individualizados. En el caso de la figura 1b, se propone una red de nodos distribuidos estratégicamente que permita representar de forma más generalizada las condiciones del sustrato en áreas compartidas.

Las tecnologías de red de área amplia y baja potencia (LPWAN, por sus siglas en inglés) se están adoptando rápidamente en aplicaciones industriales que requieren el sensado remoto, la monitorización continua y el control distribuido de sistemas a gran escala. Gracias a su bajo consumo energético y a su capacidad para soportar un número masivo de dispositivos a largas distancias, estas redes ofrecen una solución eficiente para entornos con recursos energéticos limitados. Su diseño basado en modulaciones robustas permite mantener comunicaciones estables incluso en condiciones adversas, haciendo de las LPWAN una tecnología atractiva para sectores como la agricultura de precisión, la industria 4.0 y los sistemas ambientales inteligentes [10]. En particular, en el contexto de los invernaderos agrícolas, las LPWAN permiten la monitorización eficiente y rentable de variables críticas como la temperatura, la humedad relativa, la humedad del sustrato y la intensidad lumínica. Su capacidad de operar de forma inalámbrica, su escalabilidad y su bajo costo operativo las posicionan como una alternativa superior frente a otras tecnologías de corto alcance.

Una de las tecnologías más destacadas dentro del ecosistema LPWAN es LoRa (Long Range), la cual ha ganado una notable aceptación tanto en el ámbito industrial como en el académico. Su modulación Chirp Spread Spectrum (CSS) le permite alcanzar largas distancias con alta robustez frente a interferencias, lo que la hace ideal para entornos ruidosos y con múltiples obstáculos físicos. Este trabajo investiga soluciones de acceso al medio aplicables a redes LoRa, como los que se presentan en invernaderos automatizados dedicados al cultivo de cactáceas y suculentas, donde la precisión, estabilidad y eficiencia energética del sistema son esenciales para una gestión agronómica sostenible.

Red en Malla (Mesh)

Se utilizó una red mesh para conectar múltiples nodos de sensores, ya que esta topología permite que los dispositivos se comuniquen entre sí y reenvíen datos hasta llegar a un nodo con acceso a internet [11]. Esto elimina la necesidad de que todos los nodos estén en contacto directo con un punto central, aumentando la cobertura del sistema y haciéndolo más robusto frente a fallos o bloqueos físicos. En un invernadero con estructuras que podrían interferir la señal, esta capacidad fue especialmente útil (figura 2).

El módulo RYLR998 fue seleccionado como el módulo de comunicación principal por su integración nativa con la tecnología LoRa [18]. Se utilizó un microcontrolador ESP32-S3 Super Mini [12][13], por su conectividad Wi-Fi integrada, bajo consumo, y su tamaño reducido, ideal para proyectos compactos. Su potencia de procesamiento permite recibir datos desde LoRa vía el protocolo RS-232, procesarlos y enviarlos de forma segura a la nube (Firebase) utilizando HTTPS [14]. El sensor DHT11 es digital y de bajo costo, fácil integración para la medición de temperatura y humedad relativa. Finalmente, para medir la humedad del sustrato se empleó el sensor FC-28 [15][16][17].

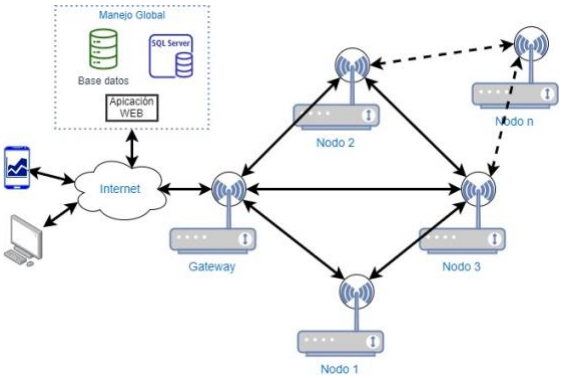


Figura 2. LoRa Mesh.

En resumen, los conceptos abordados en este apartado permiten comprender el funcionamiento y la lógica del sistema propuesto. La tecnología LoRa, junto con la estructura en red mesh, posibilita una comunicación eficiente entre nodos; mientras que los sensores seleccionados garantizan una recolección precisa de datos ambientales. Estos fundamentos técnicos serán esenciales para el diseño e implementación de la arquitectura del sistema.

La figura 3 muestra el nodo de la Red Mesh, que está compuesto por los dos sensores FC28 y el DHT11, además del transceptor LoRa. Dicho nodo se puede utilizar para transferir los datos obtenidos o también para recibirlos, dependiendo de la configuración dentro de la red. En la siguiente etapa se considera la integración de salida a hacia actuadores para poder controlar el riego de las plantas (figura 3).

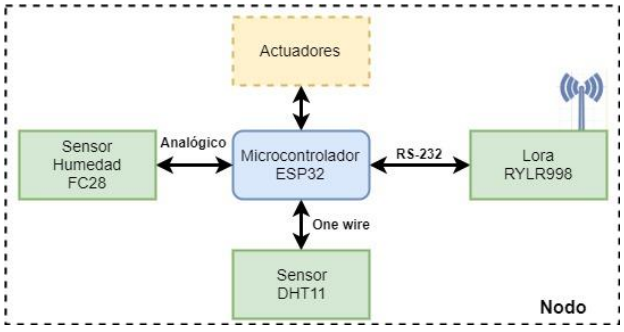


Figura 3. Nodo.

3 Arquitectura de la Aplicación

La arquitectura desarrollada para este sistema se basa en una red de comunicación de tipo Mesh, implementada con módulos LoRa RYLR998, que permiten la transmisión de datos de largo alcance con bajo consumo energético.

El nodo principal (Gateway), que actúa como receptor final, está equipado con un microcontrolador ESP32-S3, el cual se encarga de recibir los datos provenientes del módulo LoRa a través del protocolo RS-232. Una vez que los datos son recibidos, el microcontrolador los procesa y los envía mediante el protocolo HTTPS hacia una base de datos en la nube, específicamente Firebase, donde son almacenados para su análisis o visualización posterior.

El uso de una red Mesh basada en LoRa permite no solo mejorar la cobertura y confiabilidad del sistema, sino también reducir la dependencia de infraestructura de red convencional, como Wi-Fi o redes celulares. Esta arquitectura es especialmente útil en entornos donde la conectividad es limitada o se requiere una solución autónoma y eficiente para la monitorización remoto de variables.

4 Resultados

Para probar el funcionamiento del sistema desarrollado, se coloco uno de los nodos en el invernadero de cactáceas (transmisor), el recolecta los datos de temperatura y humedad relativa de ambiente y del sustrato. Los transmite para que el segundo módulo los reciba y los envía a la nube por internet.

Se eligieron varios puntos de prueba para determinar el alcance de los módulos LoRa. Las pruebas se desarrollaron dentro Ciudad universitaria, en San Manuel, durante el día. En la figura 4 se muestran en un mapa los puntos que sirvieron de prueba. En un inicio estaba pensado que el receptor quedaría ubicado en la Facultad de Ciencias de la Electrónica edificio 6 (FCE6), sin embargo, al momento de instalarlo se descubrió, la distancia en línea recta es de aproximadamente 324 m, que este alcance de lo que indica los módulos LoRa, sin embargo, – la distancia de los módulos LoRa indica de 10 km – al estar la torre de gestión en los dos puntos de comunicación, la señal no llega apropiadamente, ya que esta cruza al menos 4 edificios. Se probo también en el edificio FCE2 que se encuentra una distancia de 280 m y la señal cruza 3 edificios, pero también falla. Por lo tanto, se realizaron las pruebas en el edificio de la Facultad de Ciencias Biológicas, siendo una distancia aproximada de 61 m. Una prueba más se realizó en las mesas de Italian Coffee, teniendo buena recepción con una distancia de 234 m y cruzando 2 edificios. Se puede concluir que la distancia disminuye notablemente cuando se cruzan muros hasta unos cuantos cientos de metros (Tabla 1). La figura 5 a y b muestra la colocación del sistema diseñado, dentro del invernadero en uno de los puntos donde se realizaron las mediciones. Para evaluar el funcionamiento del sistema desarrollado, se instaló uno de los nodos transmisores dentro del invernadero de cactáceas. Este nodo recolecta datos de temperatura y humedad relativa tanto del ambiente como del sustrato, y los transmite de forma inalámbrica a un segundo módulo receptor, el cual a su vez envía la información a la nube a través de internet para su almacenamiento y análisis en tiempo real. Con el fin de determinar el alcance efectivo de los módulos LoRa, se seleccionaron varios puntos de prueba dentro de Ciudad Universitaria (BUAP), ubicada en la colonia San Manuel, Puebla. Las pruebas se realizaron durante

el día, y los puntos de prueba se muestran en la figura 4, la cual muestra un mapa con las ubicaciones de los puntos de prueba.

Inicialmente se planeó instalar el receptor en el edificio 6 de la Facultad de Ciencias de la Electrónica (FCE6). Sin embargo, al momento de la instalación se detectaron fallos en la comunicación, pese a que la distancia en línea recta es de aproximadamente 324 metros, valor muy por debajo del rango teórico de hasta 10 km especificado por los módulos LoRa. La falla se atribuye a la presencia de múltiples obstáculos físicos: la señal debía atravesar al menos cuatro edificios, incluyendo la torre de gestión. Se realizaron pruebas adicionales ubicando el receptor en el edificio FCE2, a una distancia aproximada de 280 metros, donde la señal debía cruzar tres edificios. En este caso también se presentaron interrupciones en la recepción de datos. Por lo tanto, se optó por ubicar el receptor en el edificio de la Facultad de Ciencias Biológicas, a una distancia más corta de aproximadamente 61 metros, obteniendo una transmisión estable y sin pérdida de datos. Una prueba adicional se llevó a cabo en las mesas exteriores del Italian Coffee, ubicadas a 234 metros del transmisor, con presencia de dos edificios intermedios. En este caso, la señal fue recibida exitosamente, confirmando que el desempeño de los módulos LoRa disminuye considerablemente cuando existen múltiples barreras físicas, especialmente muros de concreto o estructuras metálicas. La tabla 1 resume las distancias y condiciones de cada punto de prueba. En general, se concluye que, aunque los módulos LoRa ofrecen cobertura de largo alcance en espacios abiertos, su rendimiento se ve significativamente afectado en entornos urbanos o interiores con múltiples obstáculos. La figura 6a y 6b muestran la instalación del sistema desarrollado dentro del invernadero, específicamente en uno de los puntos donde se tomaron las mediciones.

Figura 4. Pruebas de alcance de los módulos LoRa.

Tabla 1. Pruebas de alcance de los módulos LoRa.

Sitio	Distancia (m)	Obstáculos (n)	Recepción	LoS	SF/BW/CR	Ptx (dBm)
FCB	61	1	Sí	No	12/125/4/8	22
IC	234	2	Sí	No	12/125/4/8	22
FCE2	280	3	No	No	12/125/4/8	22
FCE6	324	4	No	No	12/125/4/8	22

FCB Facultad de Ciencias Biológicas
IC Italian Coffee CU
FCE2 Facultad de Ciencias de la Electrónica Edificio FCE2
FCE6 Facultad de Ciencias de la Electrónica Edificio FCE6

LoS Línea de Vista
SF Factor de Ensanchamiento
CR Tasa de Codificación
BW Ancho de Banda
PL Tiempo de sincronización
Ptx Potencia de Transmisión

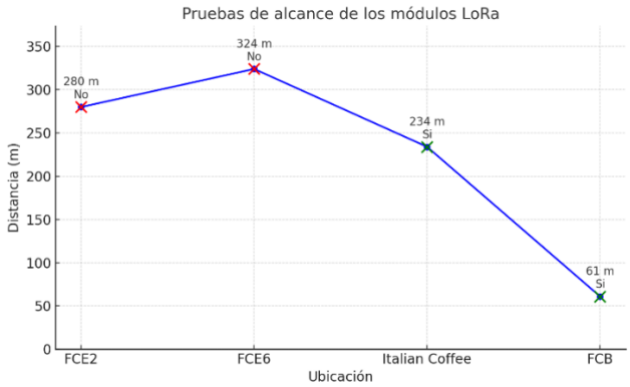


Figura 5. Alcance de los módulos Lora.

A continuación, también se presentan los datos e imágenes obtenidas mediante el sistema de monitoreo ambiental desarrollado. Estos resultados reflejan las mediciones de variables clave como la humedad del sustrato, la temperatura y otros factores ambientales registrados dentro del invernadero. La recopilación de estos datos se realizó a través de sensores distribuidos en una red en malla (Mesh) basada en tecnología LoRa, lo cual permitió una transmisión eficiente y confiable hacia la nube. Su análisis permite evaluar el comportamiento de las condiciones ambientales en tiempo real, y representa un paso fundamental para la toma de decisiones más precisas y la futura automatización del sistema, la figura 7 muestra los datos obtenidos: La fecha, hora, longitud y latitud, más los valores de la humedad relativa del sustrato y temperatura y humedad del ambiente dentro del invernadero.

a) b)

Figura 6. Invernadero y módulo ambiental. a) vista lateral y b) vista de frente.

Figura 7. Datos obtenidos de la monitorización del invernadero.

5 Conclusiones y Trabajo a Futuro

Los resultados experimentales obtenidos en las pruebas de comunicación LoRa evidencian una diferencia significativa entre el alcance teórico de los módulos y su comportamiento real en entornos urbanos densos y estructuralmente complejos, como los de Ciudad Universitaria. Aunque los fabricantes reportan un rango efectivo de hasta 10 km en línea de vista (LoS), las pruebas demostraron que la presencia de obstáculos físicos —particularmente edificios de concreto— degrada notablemente la calidad de la señal y reduce el alcance útil a unos pocos cientos de metros. En particular, los intentos de comunicación a distancias superiores a 280 metros con más de tres obstáculos intermedios presentaron fallos o pérdida total de conectividad. En contraste, se observó una transmisión estable y confiable a distancias inferiores a 250 metros, siempre que el número de obstáculos fuera limitado (uno o dos edificios como máximo). Este patrón concuerda con lo reportado en estudios previos sobre el desempeño de LoRa en entornos urbanos, donde la propagación de la señal se ve fuertemente afectada por la atenuación en muros, interferencias y efectos de multipath (Placidi et al., 2021). La ubicación en la Facultad de Ciencias Biológicas (61 m, sin obstáculos críticos) y las mesas de Italian Coffee (234 m, dos edificios intermedios) resultaron puntos óptimos de recepción, destacando la necesidad de realizar una planeación topológica cuidadosa para la instalación de redes LoRa en invernaderos dentro de contextos urbanos o institucionales. Este proyecto aporta a las ciencias de la computación al aplicar conceptos de sistemas distribuidos y comunicación en malla con LoRa, mostrando cómo los algoritmos de enrutamiento y la integración con la nube permiten diseñar soluciones robustas en entornos urbanos complejos.

Agradecimientos

Agradecemos el apoyo de la Facultad de Ciencias Biológicas por las facilidades necesarias para el desarrollo de este proyecto, en especial por el acceso al invernadero de cactáceas. De igual manera, a la Mtra. Lucero Cuautle García por su disposición y apoyo durante el proceso.

Referencias

[1] Muscarella, R., et al. (2023). Tissue succulence in plants: carrying water for climate change. *ScienceDirect*.

[2] Ogburn, R. M., & Edwards, E. J. (2010). The ecological water-use strategies of succulent plants. *Journal of Experimental Botany, 68*(9), 2121–2136.

[3] Secretaría de Medio Ambiente y Recursos Naturales (SEMARNAT). (2023). *Unidades de Manejo para la Conservación de la Vida Silvestre (UMA)*. Gobierno de México.

[4] Ronquillo, B. (2022). *Learn IoT programming using Node-RED: Begin to code full stack IoT apps and edge devices with Raspberry Pi, NodeJS and Grafana*. Apress.

[5] Beltramelli, Luca et al. "LoRa Beyond ALOHA: An Investigation of Alternative Random Access Protocols." *IEEE transactions on industrial informatics* 17.5 (2021): 3544–3554. Web.

[6] Lakhiar, I.A.; Gao, J.; Syed, T.N.; Chandio, F.A.; Buttar, N.A. Modern plant cultivation technologies in agriculture under controlled environment: A review on aeroponics. J. Plant Interact. **2018**, 13, 338–352. [CrossRef]

[7] Maraveas, C.; Bartzanas, T. Application of Internet of Things (IoT) for Optimized Greenhouse Environments. *AgriEngineering* **2021**, *3*, 954–970.

[8] Davidefa. (2018, December 5). ESP32 LoRa Mesh – 1: The basics. Hackster.io. https://www.hackster.io/davidefa/esp32-lora-mesh-1-the-basics-3a0920

[9] Juneidi, S.J.. Smart greenhouses using internet of things: case study on tomatoes. International Journal on Smart Sensing and Intelligent Systems, Sciendo, Vol. 15 (Issue 1), (2022) https://doi.org/10.2478/ijssis-2022-0019

[10] Heroucha, Abdelhak, Salim Bitam, and Abdelhamid Mellouk. "Next-Gen IoT Architecture: Long-Range, Low-Bandwidth Video Transmission Using GenAI." *2025 International Symposium on iNnovative Informatics of Biskra (ISNIB)*. IEEE, 2025. 1–6. Web.

[11] Veloso, Artur Felipe da Silva et al. "HyDSMaaS: A Hybrid Communication Infrastructure with LoRaWAN and LoraMesh for the Demand Side Management as a Service." *Future internet* 13.11 (2021): 271. Web.

[12] Litayem, Nabil. "Scalable Smart Home Management with ESP32-S3: A Low-Cost Solution for Accessible Home Automation." *2024 International Conference on Computer and Applications (ICCA)*. IEEE, 2024. 1–7. Web.

[13] Oner, Vedat Ozan. *Developing IoT Projects with ESP32 : Unlock the Full Potential of ESP32 in IoT Development to Create Production-Grade Smart Devices.* 2nd ed. Birmingham: Packt Publishing, Limited, 2023. Print.

[14] De Carli, Sullivan. *Build Mobile Apps with SwiftUI and Firebase: Learn SwiftUI and Firebase by Building Real-World Applications Communicating with a Backend*. 1st ed. Berkeley, CA: Apress, 2023. Web.

[15] “FC28 Multisite Pacing in Heart Failure — Longterm Results — Part II.” *Europace (London, England)* 2.Supplement_2 (2001): B58–B59. Web.

[16] Santos, W. S. dos, Brito, G. R. de, Soares, B. D., Soares, T. M., & de Souza, M. R. (2021). *Confidence analysis and calibration of a FC-28 soil moisture sensor mounted on a microcontroller platform*. Revista Nativa, 9(2), 198–204. https://doi.org/10.31413/nativa.v9i2.11737

[17] Llamas, L. (2018, abril 12). *Medir humedad del suelo con Arduino y sensor FC-28*. LuisLlamas.es. https://www.luisllamas.es/arduino-humedad-suelo-fc-28/

[18] REYAX LAUNCHES REYAX RYLR998 FEATURING SEMTECH LORA. vol. 35. Boston: Worldwide Videotex, 2022. Print.

Integración de Módulos en un Nanosatélite para Detección de Incendios Forestales

Integration of Modules in a Nanosatellite for Forest Fire Detection

Ángel Castillo Muñoz, Ronaldo Rodríguez García, Nicolás Rosas Guerrero, María del Carmen Santiago Díaz, Ana Claudia Zenteno Vázquez, Gustavo Trinidad Rubín Linares

Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla, México.

{cm202320330, rg202353660, rg202355015}@alm.buap.mx {marycarmen.santiago, ana.zenteno, gustavo.rubin}@correo.buap.mx

Abstract

We developed a CanSat system designed for early wildfire detection and environmental monitoring in hard-to-reach areas. The device integrates a suite sensor measuring temperature, pressure, humidity, gas concentrations, and GPS coordinates, alongside an onboard camera for visual terrain reconnaissance. Data is transmitted via a 433 MHz LoRa radio link to a ground station for real-time processing and visualization. Using the PDCA (Plan-Do-Check-Act) methodology, we implemented iterative improvements across design, implementation, and testing phases. The results demonstrate compliance with the stability, telemetry, and safe recovery criteria established by the CanSat competition framework.

Resumen

Diseñamos un sistema CanSat orientado a la detección temprana de incendios forestales y al monitoreo ambiental en zonas de difícil acceso. El dispositivo integra sensores de temperatura, presión, humedad, gases y posición GPS, además de una cámara a bordo para reconocimiento visual. La información capturada durante el descenso se transmite por radiofrecuencia a 433 MHz LoRa hacia una estación terrena, donde se procesa y visualiza en tiempo real. Aplicamos la metodología PHVA (Planificar-Hacer-Verificar-Actuar) para asegurar mejoras iterativas en diseño, implementación y pruebas. Los resultados demuestran que el sistema cumple con los criterios de estabilidad, transmisión de datos y recuperación segura establecidos para la competencia CanSat.

Keywords and phrases: CanSat, Monitoreo Ambiental, Detección Temprana de Incendios, LoRa SX1278, ESP32, Telemetría, Paracaídas, Estación Terrena.

1 Introducción

Los incendios forestales representan una de las amenazas ambientales más críticas en México y en múltiples regiones del mundo, debido a su impacto directo en la biodiversidad, la calidad del aire, la disponibilidad de agua y la economía local. Más allá de la pérdida de cobertura vegetal, los incendios generan emisiones de gases y partículas que deterioran la salud pública, reducen la visibilidad y aceleran procesos de degradación del suelo. En este contexto, la detección temprana y el monitoreo continuo del entorno son factores clave para disminuir el tiempo de respuesta, optimizar recursos de brigadas y mitigar daños antes de que el fuego se propague de forma incontrolable [1].

Sin embargo, los métodos tradicionales de monitoreo presentan limitaciones importantes. Las soluciones satelitales ofrecen cobertura amplia, pero suelen tener restricciones en resolución espacial/temporal, dependencia de condiciones atmosféricas y costos de acceso a infraestructura y procesamiento especializado [2]. Por otro lado, los sistemas terrestres (torres de vigilancia, sensores fijos o recorridos de inspección) pueden ser más precisos localmente, pero son difíciles de desplegar en zonas remotas**,** dependen de mantenimiento constante y no siempre permiten una observación rápida de áreas extensas en eventos dinámicos. Esto abre la puerta a alternativas tecnológicas de bajo costo que puedan acercar la instrumentación y la telemetría a regiones de difícil acceso [3].

Bajo este escenario, los sistemas CanSat surgen como una plataforma educativa y experimental con alto potencial para misiones reales de recolección de datos a baja altitud [4]. Un CanSat es un "satélite en miniatura" con restricciones de volumen y masa similares a una lata de refresco, diseñado para descender controladamente mientras adquiere mediciones del entorno y las transmite a una estación terrena. Su valor no solo radica en el costo reducido, sino en su capacidad para integrar electrónica, comunicación inalámbrica, software de adquisición y estructuras mecánicas en un mismo prototipo, replicando principios de diseño aeroespacial a escala [5].

En este trabajo se presenta el desarrollo del CanSat "BigBoy", propuesto en el marco de la misión "Ampliando Visiones", con el objetivo de capturar telemetría ambiental durante el descenso y transmitirla en tiempo real a una estación terrena para su visualización y análisis inmediato [6]. El sistema integra sensores de temperatura, humedad, presión, gases y posicionamiento GPS, además de una cámara a bordo para reconocimiento visual del terreno. La telemetría se comunica mediante LoRa en 433 MHz, buscando estabilidad de enlace y alcance útil en escenarios rurales, y se procesa en una plataforma de estación terrena basada en Python, MySQL y una interfaz web en Laravel.

Para asegurar un desarrollo ordenado, reproducible y con mejora continua, se aplicó la metodología PHVA (Planificar–Hacer–Verificar–Actuar**)** en las etapas de diseño, integración y pruebas. Esta estrategia permitió evaluar iterativamente la estructura, el sistema energético, la adquisición de datos y el despliegue de paracaídas, priorizando criterios críticos para la competencia: estabilidad durante el descenso, continuidad de transmisión, integridad del prototipo y recuperación segura [7][8]. En conjunto, el proyecto busca validar que una plataforma CanSat puede funcionar como herramienta de monitoreo ambiental y aproximación a detección temprana de incendios, además

de fortalecer competencias en electrónica, programación, comunicaciones y diseño mecánico dentro de un entorno académico.

2 Diseño y Desarrollo del Sistema CanSat

2.1 Requerimientos generales

En general un sistema CanSat, debe ajustarse a las siguientes restricciones:

- **Volumen máximo:** 66 mm de diámetro × 115 mm de altura.
- **Masa máxima:** 355 g.
- **Altura de liberación:** 100 m.
- **Descenso controlado:** ≤ 8 m/s mediante sistema de amortiguación.
- **Transmisión de datos en tiempo real** y recuperación intacta tras el aterrizaje.

Para cumplir estas especificaciones, dividimos el sistema en cinco subsistemas principales:

1. Estructura y mecánica.
2. Sensores y electrónica.
3. Energía.
4. Comunicación y estación terrena.
5. Descenso y recuperación.

El cumplimiento estricto de estos requerimientos garantiza la equidad competitiva y la viabilidad técnica del lanzamiento bajo los estándares nacionales.

2.2 Diseño estructural

El cuerpo del CanSat se fabrica con placas de aluminio de bajo espesor y refuerzos internos de tubos solidos de una alineación de metal, para mantener rigidez sin incrementar la masa. Se seleccionó aluminio debido a su alta relación resistencia-peso y su capacidad para disipar calor, superando al PLA que podría deformarse bajo las altas temperaturas internas generadas por la electrónica y la exposición sola. Además de una capacidad de moldeado increíble, así como la accesibilidad para su obtención.
En el interior se utiliza espuma de poliuretano de 1 cm de grosor que actúa como aislante térmico y absorbente de impactos.

La distribución interna se organiza en tres niveles:

1. **Nivel inferior:** batería $LiFePO_4$ de 7.2 V y regulador de voltaje.
2. **Nivel inferior-medio:** sensores ambientales (BME680, MiCS5524, MPU6050]) [9].
3. **Nivel superior-medio:** microcontrolador ESP32-Wrover-S (con cámara integrada)
4. **Nivel superior:** módulo GPS NEO-6M y módulo LoRa SX1278.

La parte superior aloja el compartimiento del paracaídas y el servomotor de despliegue. El balance de masa se calculó en **290 g**, manteniendo el centro de gravedad cerca del eje central para asegurar estabilidad durante el descenso.

2.3 Sistema de descenso y paracaídas

El mecanismo de descenso fue uno de los mayores retos del diseño. Se optó por un paracaídas circular de nylon ripstop, elegido por su alta resistencia al desgarre y bajo peso. Las ocho cuerdas de suspensión de hilo de pesca calibre 0.60 se fijaron a puntos equidistantes en la carcasa superior para distribuir uniformemente las tensiones.

Durante el vuelo, el MPU6050 detecta un incremento abrupto en la aceleración al momento de la liberación (modo caída libre). Esta señal activa el servomotor, que libera el paracaídas almacenado. El despliegue completo ocurre en menos de un segundo, estabilizando el descenso a una velocidad media estimada de 7.2 m/s.

Previo al vuelo se realizaron múltiples pruebas de caída desde 1.5 m, 5 m y 10 m de altura, verificando la estabilidad y la integridad del sistema. En todas ellas el paracaídas logró una desaceleración progresiva y una recuperación segura sin daños estructurales.

3 Subsistema Electrónico y Sensorial

El corazón del *CanSat BigBoy* es el microcontrolador ESP32-Wrover-S, elegido por su bajo consumo, doble núcleo de procesamiento y capacidad de ejecutar simultáneamente la adquisición de datos, el control de descenso y la transmisión LoRa. Para cumplir los objetivos de la misión integramos los siguientes sensores:

- **BME680**: mide temperatura (–40 a 85 °C, ±1 °C), presión (30 a 1100 hPa), humedad (0–100 %) y VOC.
- **MiCS5524**: detecta gases combustibles y tóxicos (CO, H_2, CH_4, NH_3, EtOH) [10].
- **MPU6050**: giroscopio y acelerómetro para calcular orientación y detectar la caída libre [11].
- **GPS NEO-6M**: proporciona coordenadas de localización con precisión de ±2.5 m [12].

La comunicación entre sensores y microcontrolador se realiza mediante protocolos I²C, SPI y UART, optimizando la velocidad de transmisión. El acondicionamiento eléctrico incluye capacitores de desacoplo de 100 nF y 220 µF para filtrar ruido y evitar reinicios durante el envío de datos.

4 Software de Vuelo y Telemetría

Desarrollamos el firmware en **Arduino IDE** usando las bibliotecas *esp32 by Espressif Systems v2.0.11* y *esp32Servo by Kevin H* [13][14][15].

El código controla el flujo de la misión en tres fases:

1. **Inicialización:** configuración de sensores y GPS.
2. **Vuelo:** lectura continua → formato JSON → envío LoRa a 433 MHz.
3. **Descenso:** detección de caída libre → activación del servomotor → despliegue del paracaídas.

La cadena JSON contiene variables como *temp, hum, press, gas, CO, H_2, lat, lon, alt, accX, accY, accZ*, lo que permite compatibilidad directa con Python y Laravel en la estación terrena [16].

5 Sistema de Comunicación y Estación Terrena

La comunicación entre el CanSat y la estación terrena es un elemento crítico de la misión, ya que garantiza la transmisión continua de telemetría durante el descenso y permite el monitoreo en tiempo real del estado del sistema y del entorno.

Para este propósito se empleó un enlace de radiofrecuencia basado en tecnología LoRa (Long Range), utilizando el módulo SX1278 operando en la banda de 433 MHz, seleccionada por su buena propagación en entornos rurales y su mayor penetración frente a obstáculos en comparación con bandas de mayor frecuencia [17][18].

La estación terrena consta de un microcontrolador receptor y una laptop que ejecuta un programa en Python, encargado de decodificar los paquetes, almacenarlos en una base de datos MySQL y visualizarlos mediante una interfaz desarrollada en Laravel (MVC).

En la plataforma web se muestran:

- Lecturas en tiempo real de los sensores.
- Posición GPS en mapa.
- Modelo 3D interactivo del CanSat.
- Historial de telemetría descargable.

6 Sistema Energético

Para garantizar autonomía durante la misión utilizamos una **batería $LiFePO_4$ de 7.2 V y 1800 mAh**, que alimenta el sistema durante más de 3 horas. El consumo promedio de los módulos se presenta en la tabla 1.

Tabla 1. Consumo promedio de cada módulo.

Módulo	Voltaje (V)	Corriente (mA)	Consumo (3 h, mAh)
LoRa SX1278	3.3	12	36
MiCS5524	5	35	105
MPU6050	3.3	5.1	15.3
BME680	3.3	12	36
GPS NEO-6M	3.3	45	135
ESP32	5	80	240
ESP32-CAM	5	98	294
Total			**≈861 mAh**

Con un margen del 80 % de seguridad, el consumo total estimado es 1076 mAh. El sistema incorpora un interruptor general y un regulador step-down para proteger los módulos sensibles.

7 Integración de Subsistemas

El ensamblaje siguió un enfoque modular:

1. **Estructura mecánica**: colocación de refuerzos de aluminio y espuma.
2. **Energía**: fijación de baterías y reguladores.
3. **Control**: instalación del ESP32 y LoRa.
4. **Sensores**: montaje perimetral para lecturas ambientales precisas.
5. **Paracaídas**: inserción en compartimiento superior con servomotor.

Cada subsistema fue probado de forma independiente antes de integrarlo al prototipo final, reduciendo riesgos de incompatibilidad eléctrica o mecánica.

8 Pruebas y Resultados

Para validar el desempeño del *CanSat BigBoy*, realizamos una serie de pruebas estructurales, térmicas, de vibración, vacío y caída.

8.1 Pruebas de resistencia ambiental

- **Prueba de calor:** se sometió la unidad a una temperatura constante de 50 °C durante 2 h, verificando lectura estable de sensores.
- **Prueba de frío:** exposición a –20 °C en cámara improvisada de hielo seco y unicel; los sensores mantuvieron comunicación normal.
- **Prueba de vibraciones:** con una pulidora orbital para simular vibraciones en los tres ejes; no se detectaron desconexiones.
- **Prueba de vacío:** dentro de una cámara sellada para comprobar integridad estructural; sin fugas ni deformaciones visibles.

8.2 Pruebas de caída

Se efectuaron lanzamientos desde 1.5 m, 5 m y 10 m, midiendo tiempos de descenso y observando la apertura del paracaídas.
El sistema logró velocidades promedio de 7.2 m/s, cumpliendo el criterio de seguridad estructural y permitiendo recuperación inmediata.

9 Análisis de Fallas y Mejora Continua

Aplicamos la metodología **AMEF (Análisis Modal de Efectos y Fallos)** para detectar posibles puntos críticos.
Los fallos más relevantes y las acciones preventivas se muestran en la tabla 2.

Tabla 2. Fallas y acciones preventivas.

Componente	Modo de fallo	Efecto	NPR	Acción recomendada
Batería	No suministra corriente	Falla total	120	Revisar conexiones y aislamiento
Servomotor	Sobrecorriente	Paracaídas no se despliega	405	Protección de corriente y revisión física
LoRa SX1278	Sin transmisión	Pérdida de datos	150	Recalibrar antena y posición
Sensores	Sobrecalentamiento	Datos erróneos	432	Aislar térmicamente
GPS NEO-6M	Pérdida de señal	Sin geolocalización	120	Reubicar antena externa

Además, aplicamos la metodología PHVA, asegurando mejora continua:

1. **Planificar:** definir pruebas y condiciones.
2. **Hacer:** ejecutar y registrar resultados.
3. **Verificar:** comparar valores esperados y reales.
4. **Actuar:** aplicar correcciones y repetir pruebas.

10 Conclusiones

El *CanSat BigBoy* demostró ser un sistema funcional y confiable para la adquisición y transmisión de datos ambientales. El uso de materiales ligeros y resistentes, junto con la integración modular de hardware y software, permitió alcanzar los objetivos establecidos. Las pruebas confirmaron estabilidad estructural, precisión sensorial y efectividad del sistema de descenso. Además, el proyecto fomenta el desarrollo de competencias en electrónica, programación, mecánica y gestión de proyectos aeroespaciales dentro de un entorno académico.

Referencias

[1] U.S. Environmental Protection Agency (EPA), "Technical Assistance Document for the Reporting of Daily Air Quality – the Air Quality Index (AQI)," EPA-454/B-18-007, 2018.

[2] National Geospatial-Intelligence Agency (NGA), "Department of Defense World Geodetic System 1984 (WGS 84): Its Definition and Relationships with Local Geodetic Systems," NGA.STND.0036_1.0.0_WGS84, 2014.

[3] F. Yuan et al. “Deep Learning for Wildfire Smoke Detection,” IEEE Access, vol. 9, 2021.

[4] American Astronautical Society (AAS), "CanSat Competition Design Guide 2024-2025," Washington D.C., 2024.

[5] CanSat Competition, *CanSat Design and Construction Guide*, 2025.

[6] ESERO Luxembourg, *CanSat Technical Guide 2020–2021*, 2025.

[7] T. W. Knacke, "Parachute Recovery Systems Design Manual," Naval Weapons Center, China Lake, CA, NWC TP 6575, 1992.

[8] National Oceanic and Atmospheric Administration (NOAA), "U.S. Standard Atmosphere, 1976," Washington, D.C., 1976.

[9] Bosch Sensortec GmbH, *BME680 Datasheet*, 2025.

[10] SGX Sensortech, *MiCS-5524 Gas Sensor Datasheet*, 2025.

[11] Semtech, *LoRa SX1278 Technical Documentation*, 2025.

[12] u-blox, "NEO-6 Series - GPS Modules Data Sheet," Rev. C, 2011.

[13] Espressif Systems, “ESP32 Series Datasheet,” Espressif Systems, 2025.

[14] Espressif Systems, “ESP32-WROOM-32 & ESP32-WROVER Datasheet,” Espressif Systems, 2025.

[15] Espressif Systems, “ESP32-CAM Development Board User Guide,” Espressif Systems, 2024.

[16] G. Welch and G. Bishop, "An Introduction to the Kalman Filter," University of North Carolina at Chapel Hill, Department of Computer Science, TR 95-041, 2006.

[17] Semtech Corporation, "SX1276/77/78/79 LoRa Transceiver Datasheet," 2024.

[18] C. A. Balanis, Antenna Theory: Analysis and Design, 4th ed., Wiley, 2016.

Evaluación Real del Desempeño de LoRa para IoT

LoRa Performance Evaluation for Real-World IoT Applications

Berenice Cortes Cinto, Josefina Castañeda Camacho, Víctor Rodolfo González Díaz

Facultad de Ciencias de la Electrónica, Benemérita Universidad Autónoma de Puebla, México.

cc223470442@alm.buap.mx, josefina.castaneda@correo.buap.mx, vicrodolfo.gonzalez@correo.buap.mx

Abstract

This paper presents a comparative study that conducted an empirical evaluation of the performance of LoRa technology under real-world conditions, contrasted with simulated measurements. LoRa operated in the 915 MHz band using the LilyGO T3 v1.6.1 board. Two scenarios were analyzed: an urban environment and a rural environment, considering fundamental metrics such as RSSI and SNR.

To carry out the analysis, Rayleigh, Rician, and AWGN channel fading models were employed; the same module parameter configuration was used, and a distance of 2 km was analyzed. The results showed that, in rural environments, an RMSE of 1.6 dB was obtained, providing a reliable communication system, unlike the performance in urban areas, where multiple factors caused the RMSE to reach 4.5 dB.

It is concluded that, while simulations provide an overview of LoRa's behavior, in urban areas it is necessary to characterize the system through field validations to ensure its reliability.

Resumen

Este trabajo presenta un estudio comparativo que realizó una evaluación empírica del desempeño de la tecnología LoRa bajo condiciones del mundo real, contrastadas con mediciones simuladas. LoRa operó en la banda de 915 MHz, utilizando la tarjeta LilyGO T3 v1.6.1. Se analizan dos escenarios, un entorno urbano y uno rural, considerando métricas fundamentales como RSSI y SNR.

Para llevar a cabo el análisis, se emplearon modelos de desvanecimientos Rayleigh, Rician, y canal AWGN; se utilizó la misma configuración de los parámetros para los módulos y se analizó una distancia de 2 Km. Los resultados mostraron que, en entornos rurales, se obtiene un RMSE de 1.6 dB, lo que ofrece un sistema de comunicación confiable, a diferencia del desempeño en las áreas urbanas, debido a que ahí se tienen múltiples factores, alcanzado un RMSE de 4.5 dB.

Se concluye que, si bien las simulaciones ofrecen un panorama acerca del comportamiento de LoRa, en el caso de áreas urbanas es necesario caracterizar el sistema bajo validaciones de campo para garantizar la confiabilidad del sistema.

Keywords and phrases: IoT, LPWAN, LoRa, MATLAB.

1 Introducción

En el mundo actual, la manera de comunicarnos evoluciona constantemente, siempre con el fin de mejorar nuestra calidad de vida. En 1982 en la Universidad Carnegie Mellon conectaron una máquina expendedora de Coca-Cola a Internet, ese se considera el primer ejemplo de Internet de las Cosas (IoT, por sus siglas en inglés), posteriormente Kevin Ashton en 1999, define un nuevo sistema donde los objetos físicos se conectan a internet para proporcionar información de interés a los usuarios como el monitoreo y control de sistemas distribuidos en tiempo real [1]. El IoT se caracteriza por su bajo consumo, largo alcance y bajo costo, pero para que esta nueva tecnología transcendiera fue necesario usar una red dedicada al IoT que es una Red de Área Amplia de Bajo Consumo (LPWAN, por sus siglas en inglés) , por tal motivo entre el año 2010 y 2015 ya se tenía una variedad de tecnologías como Wi-Fi, Bluetooth, Zigbee y Long Range (LoRa) por mencionar algunas, para fines de este documento profundizaremos en la tecnología LoRa que destaca por su equilibrio entre bajo consumo de energía y comunicación a larga distancia, que la convierte en una solución ideal para aplicaciones IoT en áreas urbanas y rurales, observe la Figura. 1.

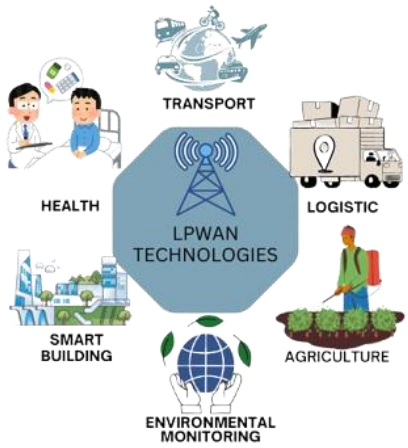


Figura 1. Campos de aplicación de las redes LPWAN. Fuente: Elaboración propia.

Sin embargo, a pesar de que LoRa se caracteriza por tener un largo alcance, puede funcionar en la banda Industrial, Científica y Médica (ISM, por sus siglas en inglés), tiene su propia modulación de espectro ensanchado (CSS), es robusto a interferencias, usa muy poca energía y se puede ajustar la velocidad de datos dependiendo de la aplicación está aún se ve afectada por las condiciones del entorno físico, como la densidad de edificios, árboles, vehículos, la interferencia electromagnética y otros tipos de tecnologías inalámbricas. Estos factores dificultan la estimación precisa de la calidad de señal real cuando se envía pequeños mensajes de un sistema IoT basado en LoRa únicamente a través de simulaciones. Por ello, es fundamental validar los modelos teóricos mediante pruebas experimentales en campo, lo que permite visualizar mejor las limitaciones y el potencial de esta tecnología en condiciones reales.

Este artículo presenta un estudio comparativo entre resultados simulados y reales de una red LoRa operando en la banda de 915 MHz, utilizando la tarjeta LilyGO T3 v1.6.1 que contiene el chip SX1276 el cual crea la comunicación LoRa. Se analizan dos escenarios contrastantes: un entorno urbano

(obstáculos múltiples) y uno rural (espacio abierto), considerando métricas clave como la Intensidad de señal recibida (RSSI) y la relación señal/ruido (SNR). Para las simulaciones se utilizó MATLAB, integrando modelos de pérdida por trayectoria, desvanecimiento por multitrayecto y ruido blanco gaussiano aditivo (AWGN, por sus siglas en inglés). Las pruebas reales se llevaron a cabo con hardware embebido, recolectando datos en campo mediante un protocolo de eco y marcas de tiempo.

El objetivo principal de este trabajo es evaluar en qué medida los modelos simulados representan el comportamiento real del sistema en diferentes entornos y así ofrecer recomendaciones prácticas para el diseño y despliegue de redes IoT con tecnología LoRa.

2 Marco Teórico y Estado del Arte

A continuación, se definen conceptos técnicos y teóricos importantes para la comprensión de este artículo.

2.1 Marco teórico

2.1.1 Internet de las cosas

El Internet de las cosas, según Maras [1] está definido como la interconexión de dispositivos físicos por medio de Internet que obtienen información de cosas sin intervención humana para compartirla al usuario con el fin ofrecer un servicio, además es capaz de monitorear a personas, animales, plantas y/u objetos para cuestiones de salud entre otras aplicaciones.

2.1.2 LPWAN

Red de área amplia de baja potencia [2] se define como una red que se caracteriza principalmente por tener un largo alcance de transmisión, un bajo consumo de energía en sus nodos finales, por su capacidad de escalabilidad y su bajo costo.

Este tipo de red tiene arquitecturas similares y se compone de cuatro elementos:

- Nodos finales
- Puertas de enlace o estación base.
- Servidores de red
- Servidores de aplicaciones

Las tecnologías orientadas a la comunicación inalámbrica IoT se divide en dos categorías:

- Redes de área local de baja potencia con un alcance inferior a 1000 metros. Esta categoría incluye IEEE 802.15.4, IEEE P802.1ah, Bluetooth/LE, etc.
- Redes de área amplia de baja potencia, con un alcance superior a 1000 metros. Esta categoría incluye LoRaWAN, pero también protocolos, como Sigfox, DASH7, etc.

a) Tecnología LoRa

LoRa (Long Range) [1] es la capa física que se basa en transmitir información por modulación de espectro ensanchado (CSS, por sus siglas en inglés). La modulación usa la técnica de Pulso de Radar Comprimido de Alta Intensidad (CHIRP, por sus siglas en inglés) que es cuando la señal transmitida aumenta o disminuye con el paso del tiempo, como se observa en la Figura.2.

Figura 2. Modulación CSS (Chirp Spread Spectrum). Fuente: Semtech Corporation, 2024.

Principales características de LoRa:

- Bajo consumo de energía
- Largo alcance
- Robustez frente al ruido
- Uso de banda ISM
- Geolocalización
- Velocidad de transmisión
- Flexibilidad
- Implementaciones públicas y privadas
- Seguridad AES-128
- Actualizaciones de firmware inalámbricas
- Roaming
- Bajo costo

En la Figura. 3. Podemos observar que LoRa destaca en equilibrio general, frente a otras tecnologías LPWAN.

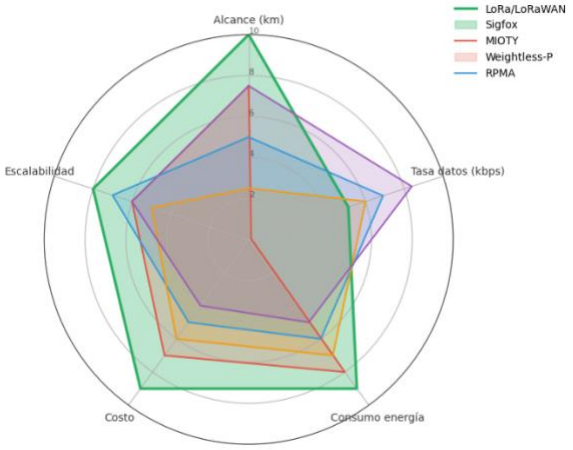


Figura 3. Comparativa de tecnologías LPWAN no celulares. Fuente: Elaboración propia.

Las principales aplicaciones de LoRa son en ciudades inteligentes, agricultura de precisión y monitoreo industrial.

La eficiencia de LoRa está en la configuración de factores como:

- Spreading Factor (SF): Determina la velocidad con la que se manda un número de muestras y que va a implicar compensaciones entre la velocidad de datos, el alcance y la resistencia a las interferencias con el fin de optimizar el rendimiento del sistema.
- Banwidth (BW): Define el ancho de banda de la señal.
- Coding Rate (CR): Relación entre bits de información y redundancia añadida por corrección de errores.
- Potencia de transmisión: Afecta principalmente el alcance la comunicación inalámbrica.

La señal dependiendo del entorno físico puede verse afectada, sin embargo, para mejorar la estimación precisa es necesario adaptar métricas como las que a continuación se mencionan.

b) Métricas de evaluación

Las métricas seleccionadas para la comparación entre un ambiente real y simulado fueron:

- RSSI (Received Signal Strength Indicator): Mide la intensidad de la señal recibida, en dBm.
- Relación señal a ruido (SNR): Mide la calidad de la señal, [6] en un tipo de comunicación inalámbrica que transmite pequeños datos a largas distancias de manera eficiente se mide la relación entre la señal útil y ruido, en dB. Se calcula mediante la Ecuación (1).

$$\mathrm{SNR(dB)}=10\cdot\log_{10}\left(\frac{P_{\text{señal}}}{P_{\text{ruido}}}\right) \tag{1}$$

- Latencia de transmisión: La latencia es el lapso de tiempo necesario para estimular todos los bits del paquete a través del medio físico de salida, es decir cuando tarda en transmitirse un paquete completo [7]. Se expresa, con (2), donde T representa el tamaño del paquete y R la tasa de transmisión.

$$L_{TX}=\frac{T}{R} \tag{2}$$

Estas métricas son utilizadas para caracterizar el comportamiento en estudios de propagación y diseño de redes IoT.

c) Canales de propagación inalámbrica

El rendimiento del sistema depende de las características del canal en donde LoRa principalmente se ve influenciado por factores como:

- Pérdida por trayectoria (Path Loss)

Es una métrica, que mide la atenuación de la potencia de la señal conforme la distancia aumenta entre el transmisor y receptor (ver Ecuación 3) debido a fenómenos como la difracción, la refracción y la dispersión [5]. Depende de la distancia d y la frecuencia f.

$$\mathrm{PL(dB)}=20\log_{10}(\mathrm{d})+20\log_{10}(\mathrm{f})+32.45 \tag{3}$$

- Desvanecimiento por multitrayecto
  - Rayleigh: Representa entornos como zonas urbanas densas sin línea de vista (NLoS).
  - Rician: Aplicable en áreas rurales o semiurbanas con línea de vista parcial (LoS).
- Canal AWGN (Ruido blanco Gaussiano Aditivo): Modela el ruido térmico presente en sistemas electrónicos, afectando la relación señal a ruido.

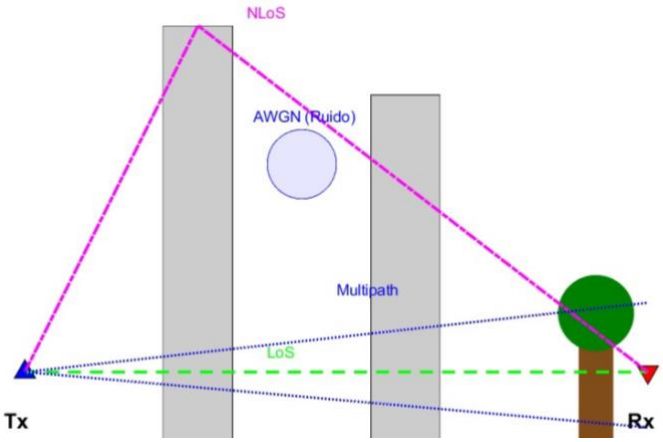


Figura 4. Modelos de propagación inalámbrica. Elaboración propia.

Estos modelos se implementaron en Matlab para simular condiciones térmicas y el comportamiento del canal en los escenarios de interés.

2.2 Estado del arte

2.2.1 Estudios previos sobre comunicaciones LoRa

La empresa francesa Cycleo desarrolló la tecnología LoRa alrededor del año 2009 aunque está paso a manos de Semtech Corporation para ser comercializada y estandarizada a nivel global.

Desde entonces, múltiples investigadores han evaluado el comportamiento de LoRa en distintos escenarios, con el fin de garantizar su fiabilidad y adaptabilidad frente a los requisitos específicos de cada aplicación [10], [11],[12]. Estas investigaciones han abordado aspectos como la calidad del enlace, el consumo energético, la validación del alcance real de la señal, la escalabilidad del sistema y la seguridad mediante pruebas de ataque y mecanismos de defensa principalmente.

2.2.2 Validación entre simulación y pruebas reales

La simulación del rendimiento de las redes LoRa es una herramienta útil y de bajo costo computacional, que nos ayuda a visualizar cómo será su comportamiento en escenarios reales. Sin embargo, no siempre ofrece una representación precisa de la realidad, incluso cuando se consideran múltiples factores. Por ejemplo, [8],[9] validaron modelos de simulación en NS-3, concluyendo que falta aproximar el comportamiento en áreas con obstáculos múltiples.

Este trabajo aborda precisamente esa brecha, integrando simulaciones en MATLAB con experimentos reales utilizando hardware accesible (tarjeta LilyGO T3 v1.6.1). Además, se incluye el análisis de mapas de calor georreferenciados, una herramienta poco explorada en estudios previos.

2.2.3 Limitaciones actuales en modelado de canales para LoRa

Este artículo propone la combinación de modelos de pérdida por trayectoria y desvanecimiento con datos reales obtenidos en campo. Esto permite validar las capacidades predictivas de los modelos actuales en entornos específicos para conocer el desempeño del sistema para futuras aplicaciones.

Como se sabe existen diversos modelados para canales sin embargo el modelo normal logarítmico ha mostrado que los RMSE desencadenan problemas significativos en el sistema de comunicación como en [15]. En este trabajo se contempla que la tierra es plana y por tanto solo nos apoyamos del GPS para su ubicación, pero otra de las limitaciones es que en las zonas montañosas o rurales resaltan que las irregularidades geográficas pueden impactar en la cobertura del sistema, por lo tanto, es necesario caracterizar el área de acuerdo a sus condiciones específicas [16].

Estudios recientes han demostrado que las soluciones basadas en LoRa pueden mitigar estas limitaciones. Algunos de los cuales son los presentados por, [17] que analizó la cobertura y calidad de la señal de red LoRaWAN en tiempo real en el monitoreo del agua del río Cikumpa en Depok City, destacando su efectividad en entornos urbanos.

Asimismo, [18] implementó un sistema LoRaWAN para monitorear la calidad del agua en áreas urbanas, abordando problemas de conectividad y desafíos en la implementación.

3 Arquitectura de la Aplicación

El objetivo es recolectar datos y analizar los parámetros fundamentales de LoRa (RSSI, SNR y Latencia) y comparar su comportamiento real con los modelos teóricos de desvanecimiento Rayleigh y Rician, en entornos tanto urbanos como rurales usando MATLAB. Este proceso siguió la siguiente secuencia:

3.1 Selección del área de estudio

La adquisición de datos se realizó en dos zonas de Puebla seleccionadas estratégicamente con línea de vista (LoS) y sin línea de vista (NLos) entre transmisor y receptor. La distribución geográfica de los puntos de medición se muestra en la Figura 5.

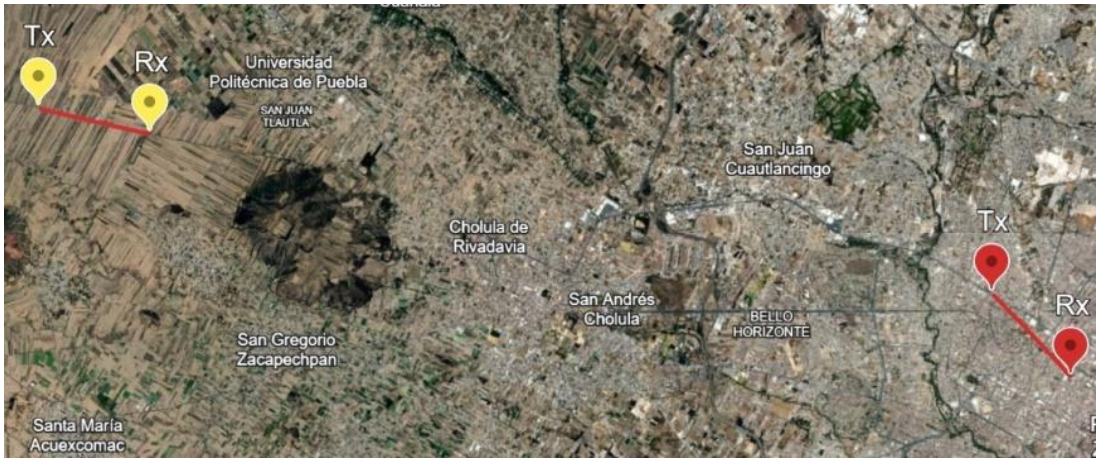


Figura 5. Comparación espacial entre (A) una zona LoS y (B) una zona NLos (Puebla de Zaragoza). Fuente: Google Earth (2025).

Figura 6. Entorno rural (Camino real).

Tx 19°05'31"N 98°23'28"W a Rx 19°05'16"N 98°22'21"W

Figura 7. Entorno urbano (Puebla de Zaragoza).

Tx 19°03'44"N 98°13'50"W a Rx 19°02'56"N 98°13'02"W

Para la simulación y análisis del desempeño de la red LoRa se utilizaron herramientas tanto de software como de hardware.

3.2 Sistema de comunicación

La arquitectura del sistema se basa en una red punto a punto empleando tecnología LoRa a 915 MHz. La metodología tiene la siguiente secuencia:

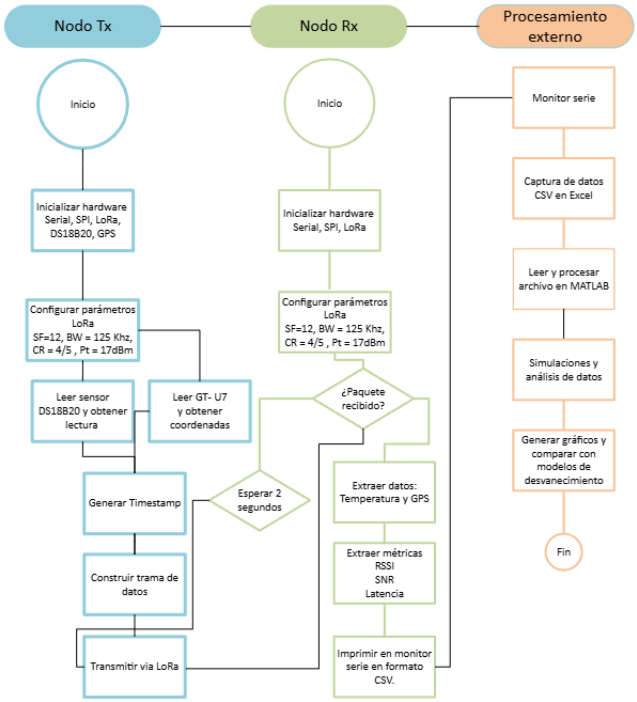


Figura 8 Diagrama de flujo del sistema IoT: nodo transmisor, receptor y procesamiento de datos. Elaboración propia.

- Nodo Tx:

Se uso como microcontrolador un ESP32 PICO D4 y el chip transceptor LoRa SX1276, que está integrado dentro de la tarjeta LilyGO T3 v1.6.1, este módulo es de fácil adquisición y es más económico que usar un ESP32 con algún otro módulo de LoRa. Este nodo se encarga de enviar la temperatura del sensor DS18B20 cada 2 segundos con un identificador, las métricas mencionadas anteriormente y la información de un módulo GPS GT-U7 para georreferenciar las coordenadas geográficas.

- Nodo Rx:

También basado en la LilyGO T3, recibe la temperatura, calcula la latencia de ida y vuelta a partir del timestamp recibido y extrae la RSSI y SNR. Cabe destacar que este módulo cuenta con una antena omnidireccional de 5dBi lo que permite captar señales desde cualquier dirección y además cuenta con 4 protocolos de comunicación Wi-Fi, BT, BLE y LoRa. Además, cabe mencionar que ambas tarjetas se alimentan con baterías de Litio de 3.7 v a 2000mAh.

La programación de estas tarjetas se realizó con el software Arduino IDE 2.3.4 en donde la comunicación entre el microcontrolador y el transceptor es por medio de el interfaz SPI por tanto usamos la librería SPI.h, para la configuración de los parámetros SF=12, BW= 125 KHz, CR= 4/5 así como Pt= 17 dBm se utilizó la librería LoRa.h. El valor de los parámetros se eligió con el criterio de

tener mayor alcance con menos tasa de transferencia, aunque el usar los 17 dBm no garantiza un ahorro de energía.

Para el sensor ambiental utilizamos la librería OneWire.h y DallasTemperature.h para la precisión de sensado y para la georreferenciación usamos la biblioteca de TinyGPS++.h

- Procedimiento de Medición:

Los datos obtenidos se almacenan primero en una hoja de Excel, para ello fue necesario indicar desde la programación en Arduino que mostrará los datos en el monitor serie en formato CSV, posteriormente desde MATLAB mandamos a llamar los datos obtenidos, donde se realizan las simulaciones correspondientes bajo las condiciones controladas y se genera la comparación con los datos reales.

En ambos casos, la distancias entre los nodos variaron hasta llegar a 2 kilómetros aproximadamente, permitiendo construir un mapa con la atenuación y evaluar el rendimiento en función de la distancia.

4 Resultados

Para analizar y modelar los datos adquiridos, se empleó la expresión del modelo de propagación Rayleigh para una zona con obstáculos y Rician para un espacio abierto. La gráfica comparativa de la distribución se muestra a continuación.

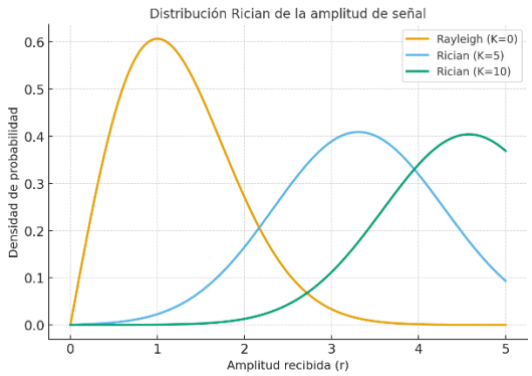


Figura 9. Distribución Rician de la amplitud de señal para distintos valores del factor.

El modelo Rician predice la distribución de la potencia que se envía de manera dispersa por múltiples trayectos de una señal dominante con linea de vista por medio de la estadística. Su factor suele expresarse como:

$$K = \frac{\text{Potencia LoS}}{\text{Potencia multipath}} \tag{4}$$

Con la RSSI y SNR obtenida se obtiene que el valor de K = 6.13 dB . Puesto que se tiene:

RSSI Mínimo = -114.0

Máximo = -104.0 dBm

	Media = -111.3 dBm
	Desviación estándar = 1.5 dB
SNR	Mínimo = -15.3 dB
	Máximo = 5.5 dB
	Media = 0.2 dB
	Desviación estándar = 3.2 dB

Correlación RSSI – SNR ⇒ 1.5 dB – 3.2 dB⇒ 0.434

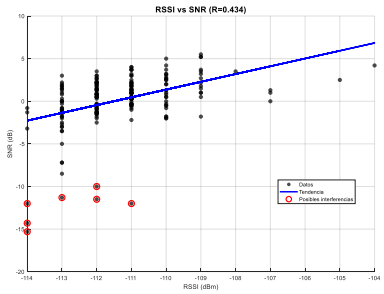


Figura 10. Rician con LoS.

El sistema encontró 7 posibles interferencias. La RSSI muestra que la señal es débil pero estable, pero el SNR nos muestra que afecta el ruido ambiental, por lo que el enlace es confiable con unas ligeras perdidas.

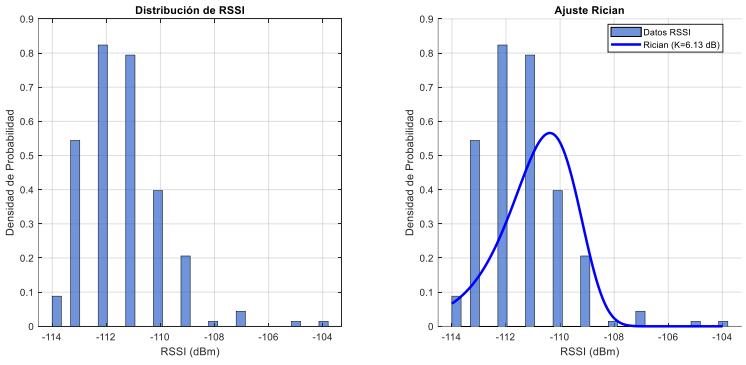


Figura 11. Distribución y ajuste Rician.

Rayleigh

A diferencia de Rician, describe como varia una señal tomando en cuanta las señales recibidas de todos los reflejos, difracciones y dispersión multitrayecto.

La expresión de Rayleigh expresada en dBm es la siguiente:

$$p_{dBm}=10\log_{10}\left(\frac{R^2}{1mW}\right)=20\log_{10}\left(\frac{R}{1mW^{1/2}}\right) \quad (5)$$

Con la RSSI y SNR obtenida se obtiene que el valor de K = 10.81dB . Puesto que se tiene:

RSSI	Mínimo = -117.0 dBm
	Máximo = -30.0 dBm
	Media = -95.3 dBm
	Desviación estándar = 16.8 dB
SNR	Mínimo = - 2.5 dB
	Máximo = 12.0 dB
	Media = 8.7 dB
	Desviación estándar = 1.8 dB

Correlación RSSI – SNR ⇒ 16.8 dB – 1.8 dB⇒ 0.586

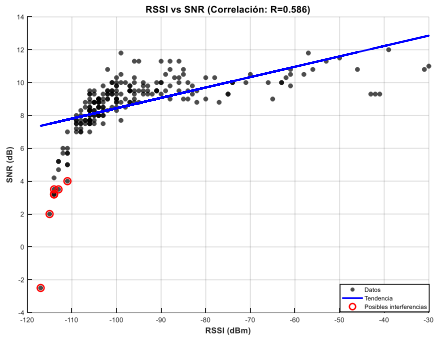


Figura 12. Rayleigh.

Por lo que el sistema encontró 7 puntos de posibles interferencias, con los parámetros LoRa configurados como se en la Figura

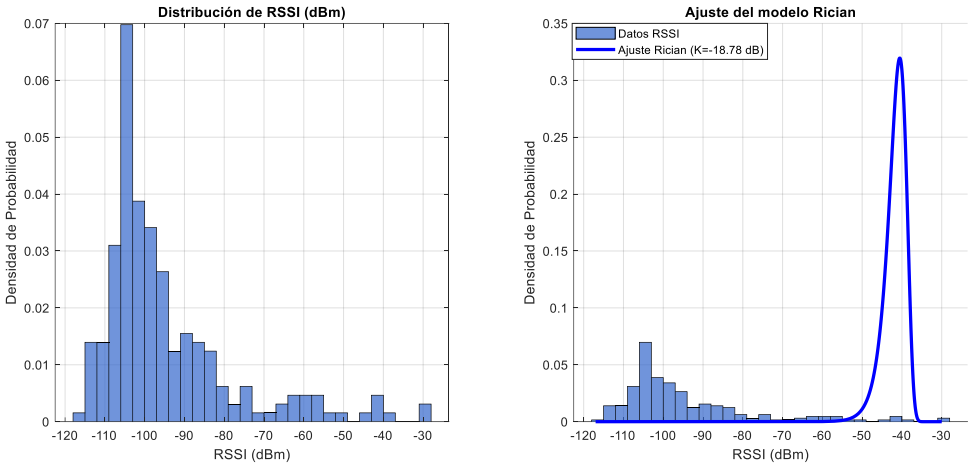


Figura 13. Distribución y ajuste del modelo Rayleigh.

Al graficar los datos se observa que es un canal Rician muy cercano a Rayleigh, esto se quizá se debe a que el sensor al ser de metal aumenta la dispersión multitrayecto. Ahora bien, al usar los valores de la Tabla 1 y 2 con el canal de AWGN se obtiene;

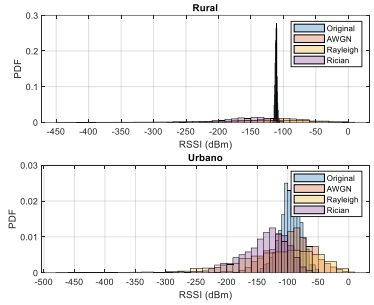


Figura 14. AGWN de los modelos de desvanecimiento.

Utilizando la RMSE se puede observar que en la Figura 14. se desvía la señal tras pasar por los modelos de desvanecimiento.

RMSE Tabla 1: AWGN=36.26, Rayleigh=76.34, Rician=54.65

RMSE Tabla 2: AWGN=30.10, Rayleigh=67.38, Rician=50.96

5 Conclusiones y Trabajo a Futuro

En el área con línea de vista se observó que los efectos Rician fueron menos pronunciados y el RMSE con respecto a la RSSI promedio de -111.3dBm y SNR promedio de 0.2 dB muestra que el canal AWGN introduce un RMSE de 1.6 dB que afecta menos a la señal y hace que sea más confiable la comunicación. Por otro lado, en el canal Rayleigh se generó un RMSE de 4.5dB por que entorno tiene más obstáculos y la señal se dispersa y hace que el canal sea complejo para recibir datos.

Los resultados obtenidos muestran una diferencia significativa entre el comportamiento ideal y el real, en donde en las áreas urbanas es más notoria esta variación. Esto confirma que los modelos simulados, aunque son útiles deben de ser caracterizados dependiendo de la aplicación y ubicación especialmente en zonas con múltiples factores. El trabajo que se desarrolló es una aportación a la tecnología LoRa que se puede usar para monitorear la agricultura o bien para aplicaciones en ciudades inteligentes.

Como trabajo futuro, se propone mejorar los modelos de simulación e integrar técnicas de ajuste dinámico de parámetros usando inteligencia artificial.

Agradecimientos

Los autores de este artículo agradecen al Consejo Nacional de Ciencia y Tecnología (CONACYT) y al Tecnológico Nacional de México Campus Orizaba por el apoyo otorgado en la realización de este trabajo.

Referencias

[1] M.-H. Maras, “Internet of Things,” in Encyclopedia of Security and Emergency Management, Cham: Springer International Publishing, 2021, pp. 473–474. doi: 10.1007/978-3-319-70488-3_71.

[2] A. Auteur, B. Auteur, y C. Auteur, “A systematic and comprehensive review on low power wide area network: characteristics, architecture, applications and research challenges,” Internet of Things, vol. 2025, Springer, Cham, 2025.

[5] “Path Loss,” in Mobile Channel Characteristics, International Series in Engineering and Computer Science, vol. 555, Springer, Boston, MA, USA, 2002, pp. 1–13.

[6] IEEE Standard 802.11-2020, “Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements: Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications,” IEEE, 2020.

[7] S. Bradner, “Benchmarking Terminology for Network Interconnection Devices,” RFC 1242, Internet Engineering Task Force (IETF), Jul. 1991. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc1242

[8] D. Dinev, A. Haka, V. Aleksieva, and H. Valchanov, “Analysis of LoRa RSSI Data Using Simulations and Real Devices,” Jun. 2023, doi: 10.1109/elma58392.2023.10202359.

[9] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, “A Study of LoRa: Long Range & Low Power Networks for the Internet of Things,” Sensors, vol. 16, no. 9, p. 1466, Sep. 2016, doi: 10.3390/S16091466.

[10] D. Dinev, A. Haka, V. Aleksieva, and H. Valchanov, “Analysis of LoRa RSSI Data Using Simulations and Real Devices,” Jun. 2023, doi: 10.1109/elma58392.2023.10202359.

[11] F. M. Ortiz, T. T. de Almeida, A. E. Ferreira, and L. H. M. K. Costa, “Caracterização de Desempenho de uma Rede LoRa em Ambientes Urbanos: Simulação vs. Prática,” Sep. 2019, doi: 10.5753/COURB.2019.7476.

[12] S. Poovalingam, R. Sangalge, N. Pareekutty, and A. Panda, “LoRA Communication Testing for LOS and NLOS Operations,” pp. 1–7, Dec. 2023, doi: 10.1109/gcitc60406.2023.10426113.

[13] Simon, M. K., & Alouini, M. S. (2004). Digital communication over fading channels. John Wiley & Sons.

[14] Shankar, P. M. (2017). Fading and shadowing in wireless systems. Springer.

[15] Olivera-Reyna, R., Olivera-Reyna, R., Muñoz-Minjares, J., Flores–Troncoso, J., & Vite-Chávez, O. (2022). Análisis del modelo de sombreado normal-logarítmico aplicado a un enlace lora punto a punto. Ciencia Latina Revista Científica Multidisciplinar, 6(6), 5153-5166. https://doi.org/10.37811/cl_rcm.v6i6.3802

[16] Castillo, P., Ojeda, M. A. V., Puglla, S. D. M., Rodriguez, J. C. G., & Peñaloza, D. A. R. (2024). Diseño y desarrollo de un dispositivo con tecnología lora para el monitoreo del estrés abiótico en cultivos de lechugas. Ciencia Latina Revista Científica Multidisciplinar, 8(2), 6827-6845. https://doi.org/10.37811/cl_rcm.v8i2.11096

[17] Ariana, H., Prameswono, L. P., Faizah, S., Putri, A. O., Jaenul, A., Dionova, B. W., Al-Humairi, S. N. S., & Mohammed, M. N. (2024). Analysis of LoRaWAN network signal coverage and quality parameters in real-time: Case study of Cikumpa River water quality monitoring, Depok City. *Teknika: Jurnal Sains dan Teknologi*, 13(3), 53–61.

[18] Garcia, P., & Rodriguez, L. (2016). LoRaWAN for water quality monitoring in urban areas. *Journal of Water and Environment Technology*, 9(3), 56–64. https://doi.org/10.xxxxx

Clasificación de la Polaridad en Opiniones Políticas mediante Modelos Preentrenados de Lenguaje

Polarity Classification in Political Opinions through Pretrained Language Models

Angeles Belém Priego Sánchez

Universidad Autónoma Metropolitana Unidad Azcapotzalco
Ciudad de México, México.

abps@azc.uam.mx

Abstract

In the digital age, the need for social interaction has evolved thanks to the rise of technology and social media, which allow people to share ideas, emotions, and opinions on a global scale. This ability to communicate not only facilitates personal connection but has also become a key tool for activities such as marketing, business, and political campaigns. However, the growing volume of information circulating online presents the challenge of accurately interpreting user-generated content, especially on social media platforms like Twitter. To address this need, the field of Natural Language Processing (NLP) has emerged as a powerful branch of artificial intelligence that enables machines to understand and interpret human language. One of its most impactful applications is sentiment analysis, which can identify the polarity—positive, negative, or neutral—of a message. This technique is essential for understanding public perceptions, evaluating the impact of political discourse, and anticipating social trends. This article presents the development and evaluation of two models based on BERT (Bidirectional Encoder Representations from Transformers), aimed at classifying the polarity of political comments on Twitter. The results show that the model trained in Spanish achieved an accuracy of 86.5%, outperforming the English-based model, which reached 81.4%. This difference is attributed to variations in vocabulary and linguistic context. These findings highlight the potential of BERT for sentiment analysis and open the door to extending its application to other thematic areas beyond the political sphere.

Resumen

En la era digital, la necesidad de interacción social ha evolucionado gracias al auge de la tecnología y las redes sociales, las cuales permiten compartir ideas, emociones y opiniones a nivel global. Esta capacidad de comunicación facilita la conexión personal y se ha convertido en una herramienta clave para actividades como el marketing, los negocios y las campañas políticas. Sin embargo, el creciente volumen de información que circula en internet plantea el reto de interpretar adecuadamente el contenido generado por los usuarios, especialmente en redes sociales como Twitter. Para responder a esta necesidad, ha emergido con fuerza el campo del Procesamiento de Lenguaje Natural (PLN), una rama de la inteligencia artificial que permite a las máquinas comprender e interpretar el lenguaje humano. Una de sus aplicaciones más potentes es el análisis de

sentimientos, capaz de identificar la polaridad —positiva, negativa o neutral— de un mensaje. Esta técnica es clave para comprender percepciones ciudadanas, evaluar el impacto de discursos políticos o anticipar tendencias sociales. Este trabajo presenta el desarrollo y evaluación de dos modelos basados en BERT (Bidirectional Encoder Representations from Transformers), enfocados en clasificar la polaridad de comentarios políticos en Twitter. Los resultados muestran que el modelo entrenado en español obtuvo una precisión del 86.5%, superando al modelo en inglés (81.4%), debido a las diferencias en vocabulario y contexto lingüístico. Estos resultados muestran el potencial del uso de BERT en análisis de sentimientos y abren oportunidades para extender su aplicación a otras áreas temáticas más allá del ámbito político.

Palabras clave y frases: Procesamiento de Lenguaje Natural, Análisis de Sentimientos, Discurso Político, BERT.

1 Introducción

La naturaleza social del ser humano se ha visto potenciada en la era digital, donde las tecnologías y redes sociales han multiplicado las oportunidades de interacción y conexión entre individuos. Plataformas como Twitter han transformado la manera en que se intercambian ideas, permitiendo la expresión inmediata de opiniones sobre temas relevantes, especialmente políticos. Esta red, por su inmediatez y carácter textual, se ha convertido en un espacio donde se reflejan las emociones y percepciones colectivas sobre decisiones gubernamentales, figuras públicas y políticas públicas.

El flujo constante y masivo de información generado en estos entornos representa una fuente invaluable de datos, pero también un desafío para su análisis e interpretación. Ante esta necesidad, surge el Procesamiento de Lenguaje Natural (PLN), una disciplina que combina inteligencia artificial y lingüística computacional para permitir que las máquinas comprendan y generen lenguaje humano. Una de sus aplicaciones más relevantes es el análisis de sentimientos, que busca identificar la polaridad emocional de los mensajes, clasificándolos como positivos, negativos o neutrales [10].

Esta técnica es especialmente útil en el contexto político, ya que permite extraer percepciones ciudadanas sobre temas sensibles y detectar tendencias en tiempo real [4]. Twitter se ha consolidado como una fuente clave para estos análisis por su naturaleza pública, breve y directa, aunque el lenguaje utilizado suele presentar desafíos como abreviaturas, ironía o ambigüedad semántica. En este escenario, los modelos de lenguaje preentrenados como BERT (Bidirectional Encoder Representations from Transformers) han marcado un hito en el PLN [3]. Su arquitectura bidireccional permite comprender el contexto completo de cada palabra dentro de una oración, mejorando notablemente el desempeño en tareas como clasificación de texto, análisis de sentimientos [16] o extracción de entidades [7].

Este trabajo se inscribe dentro de esa línea de investigación, al desarrollar y comparar dos modelos basados en BERT —uno entrenado en español y otro en inglés— aplicados a un conjunto de tuits sobre temas políticos. El objetivo es evaluar la capacidad de estos modelos para clasificar la polaridad de opiniones políticas, y aportar evidencia empírica sobre la influencia del idioma en el desempeño de modelos de PLN. En este sentido, la investigación refuerza la utilidad de BERT como

herramienta para el análisis del discurso político en redes sociales, y también abre nuevas perspectivas para su aplicación en otros dominios sensibles al lenguaje, como la salud pública, la educación, el consumo o la gestión de crisis, donde interpretar con precisión las emociones y opiniones expresadas por los usuarios es clave para tomar decisiones más informadas y oportunas.

En la actualidad, las redes sociales han transformado profundamente la forma en que los ciudadanos expresan sus opiniones y emociones, especialmente en contextos políticos. Twitter, por su naturaleza pública, breve e inmediata, se ha consolidado como una plataforma clave para el análisis de percepciones sociales. Sin embargo, el volumen masivo y la complejidad lingüística de los mensajes —caracterizados por el uso de abreviaturas, ironía, ambigüedad y lenguaje informal— representan un reto significativo para su análisis automático.

En este contexto, surge el problema central de esta investigación: evaluar cómo el idioma de entrenamiento de los modelos BERT influye en su capacidad para clasificar sentimientos en tuits políticos en español. La existencia de modelos preentrenados en diferentes idiomas (como el BERT multilingüe o BETO, entrenado específicamente en español) abre la pregunta sobre cuál de ellos ofrece un mejor rendimiento en tareas sensibles al idioma, como el análisis de discursos políticos. La contribución, de este trabajo de investigación, permite generalizar técnicas de PLN a otros campos donde la interpretación precisa del lenguaje emocional y contextual es fundamental para la toma de decisiones informadas.

2 Estado del Arte

La política ha sido siempre un tema controvertido que genera diversas opiniones sobre sistemas de gobierno, ideologías y leyes. Con el auge de redes sociales como Twitter, las personas pueden expresar y debatir sus ideas en tiempo real, convirtiendo a esta plataforma en una valiosa fuente para analizar el discurso ciudadano. Por su inmediatez y alcance, Twitter ha sido ampliamente estudiado en el análisis automático de opiniones [1], destacando la minería de sentimientos como una herramienta clave para identificar emociones y valoraciones expresadas por los usuarios. En los últimos años, el análisis de sentimientos ha evolucionado significativamente gracias al desarrollo de modelos de lenguaje profundo [11], siendo BERT uno de los más influyentes e innovadores.

BERT, propuesto por [3], introdujo un cambio de paradigma al utilizar una arquitectura basada en transformers y un entrenamiento bidireccional del lenguaje, lo que permite al modelo comprender el contexto de una palabra a partir de sus palabras adyacentes, tanto anteriores como posteriores. Esta característica mejora notablemente la capacidad de interpretación semántica y contextual del lenguaje, en comparación con modelos tradicionales basados en enfoques unidireccionales o métodos de bag of words. Investigaciones recientes han explorado con éxito su aplicación en entornos sociales digitales [17], como Twitter, evidenciando su capacidad para interpretar con mayor precisión el lenguaje natural cargado de emociones, ironía o contexto político. En [6] se presenta a VADER (Valence Aware Dictionary and sEntiment Reasoner), un modelo de análisis de sentimientos basado en reglas diseñado específicamente para textos provenientes de redes sociales.

En [9] se utilizó a BERT para analizar discursos sobre vacunación en Estados Unidos, mientras que en [18] se propuso una arquitectura heurística basada en BERT para mejorar la clasificación emocional de tuits. En el contexto hispanohablante, en [15] se analizaron publicaciones de partidos

políticos en Twitter utilizando modelos preentrenados como BETO y RoBERTuito, lo que demuestra la relevancia de adaptar el análisis lingüístico a características culturales y lingüísticas específicas. Asimismo, en [13] se desarrolló un enfoque híbrido con BERT que alcanzó altos niveles de precisión en la categorización de sentimientos, y en [12] se aplico este modelo para evaluar reputación digital. Estas investigaciones evidencian que los modelos preentrenados como BERT, especialmente cuando se personalizan lingüística y temáticamente, representan una herramienta poderosa para el análisis automatizado de la opinión pública en plataformas digitales, ofreciendo nuevas oportunidades para la comprensión de discursos políticos, toma de decisiones informadas y diseño de estrategias comunicativas más efectivas.

BERT ha sido ampliamente adoptado en tareas de PLN, tales como la clasificación de textos, la detección de entidades, la respuesta a preguntas y el análisis de sentimientos [19]. En particular, su implementación en el análisis de opiniones ha mostrado un rendimiento sobresaliente al clasificar la polaridad emocional de mensajes breves, como los publicados en Twitter. Esto se debe a que BERT no solo considera el significado literal de las palabras, sino también su intención contextual, lo cual es crucial en plataformas sociales donde el lenguaje tiende a ser informal, ambiguo y cargado de matices. Dado su éxito, han surgido múltiples variantes de BERT especializadas por idioma y dominio. Tal es el caso de BETO, una versión entrenada específicamente en corpora en español, lo que ha mejorado el rendimiento en tareas lingüísticas propias de este idioma [2]. Estudios comparativos han demostrado que los modelos monolingües como BETO superan frecuentemente a BERT multilingüe o a modelos entrenados en inglés, especialmente en la clasificación de textos altamente contextuales como los comentarios políticos, donde el vocabulario, la cultura y el tono tienen un papel determinante [5]. En el ámbito político, el uso de BERT ha permitido realizar análisis más precisos de la opinión pública expresada en redes sociales [15].

Recientes aplicaciones de BERT incluyen la detección de polarización ideológica, el monitoreo de campañas electorales, la identificación de discursos de odio y la evaluación de percepciones hacia líderes políticos [8, 14]. Estas investigaciones evidencian su capacidad para mejorar la exactitud del análisis emocional y ofrecer una comprensión más profunda del lenguaje utilizado en los debates públicos. Gracias a su capacidad multilingüe y manejo de la ambigüedad, BERT se consolida como una herramienta clave para el análisis automatizado del lenguaje. Este trabajo se enmarca en esta línea investigativa, explorando su uso para clasificar la polaridad de comentarios políticos en Twitter, mediante la comparación de modelos entrenados en inglés y en español. El objetivo es validar su eficacia en contextos hispanohablantes y contribuir al desarrollo de herramientas de PLN más inclusivas, culturalmente adaptadas y precisas.

3 Metodología Propuesta

La metodología propuesta se desarrolló con el objetivo de construir y comparar modelos de clasificación de sentimientos en comentarios políticos de Twitter, utilizando arquitecturas de lenguaje preentrenadas basadas en BERT. Este enfoque permite evaluar el rendimiento técnico de los modelos y muestra el comportamiento del lenguaje político en redes sociales, desde una perspectiva computacional y lingüística. La metodología se dividió en cinco etapas principales:

recolección de datos, preprocesamiento, etiquetado y tokenización, entrenamiento y evaluación. En la figura 1 se pueden visualizar de manera secuencial cada una de dichas etapas, las cuales forman un flujo integral orientado a capturar, procesar y analizar el discurso político digital. A continuación, se describe cada una de estas etapas.

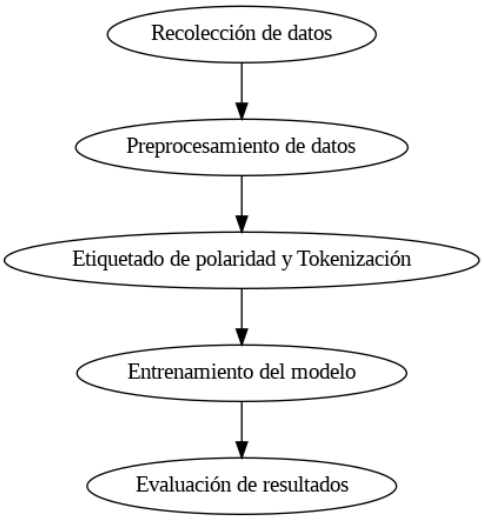


Figura 1. Comparativa de las métricas en modelos BERT base en español e inglés.

3.1 Recolección de datos

La recopilación de datos se llevó a cabo utilizando la Twitter API v2, específicamente mediante un punto de acceso de búsqueda avanzada, lo cual permitió acceder a tuits históricos y filtrarlos por criterios específicos. Para focalizar el análisis en el dominio político, se definió un conjunto de palabras clave relevantes a este ámbito, entre las que se incluyeron términos como: “gobierno”, “presidente”, “senado”, “elecciones”, “corrupción”, “partido político”, “candidato”, “democracia” y “protestas”. Estas palabras fueron seleccionadas a partir de un análisis preliminar de tendencias y del vocabulario político frecuente en países hispanohablantes.

Se estableció como periodo de captura un intervalo de tres meses comprendido entre mayo y julio de 2024, con el fin de contar con representatividad en distintos eventos y contextos políticos (campañas electorales, reformas legales, crisis institucionales). Los tuits fueron recolectados en su idioma original, restringiendo la búsqueda al idioma español. Además, se filtraron únicamente los tuits originales, excluyendo retuits y respuestas, con el fin de garantizar la independencia semántica del texto. También se descartaron mensajes duplicados y aquellos generados por bots, identificados mediante patrones repetitivos de publicación y metadatos anómalos.

Una vez recolectados, los tuits fueron almacenados en texto plano, preservando tanto el texto como los metadatos relevantes (fecha, ID del tuit, número de likes y retuits, nombre de usuario anonimizado), lo cual facilitó etapas posteriores de análisis y posible replicabilidad del estudio. En total se obtuvieron aproximadamente 50,000 tuits —utilizados en cada uno de los experimentos— , constituyendo un corpus representativo de opiniones políticas generadas por usuarios de distintas regiones hispanohablantes, como México, España, Argentina, Colombia, entre otros. Esta fase

resultó fundamental para determinar la validez y la relevancia del conjunto de datos, que constituye la base sobre la cual se entrenaron y evaluaron los modelos de clasificación de polaridad.

3.2 Preprocesamiento de datos

Una vez recolectado el conjunto de tuits en texto plano, se llevó a cabo un riguroso proceso de preprocesamiento con el objetivo de optimizar la calidad del texto para su posterior análisis y entrenamiento en modelos de lenguaje. Esta fase es esencial para reducir ruido, estandarizar los datos y maximizar la eficacia del modelo BERT en la tarea de clasificación de polaridad. El proceso incluyó las siguientes etapas:

- Extracción del texto: Se extrajo únicamente el contenido textual del tuit.
- Eliminación de ruido: Se removieron elementos que no aportaban información semántica útil, como URLs, menciones a otros usuarios (@usuario), hashtags no informativos, signos de puntuación excesivos, emojis y caracteres especiales. Esto se realizó mediante expresiones regulares personalizadas.
- Normalización del texto: Todo el contenido fue convertido a minúsculas para evitar duplicidad de tokens por diferencias de capitalización (por ejemplo, “Gobierno” vs “gobierno”). Asimismo, se eliminaron espacios en blanco redundantes.
- Eliminación de stopwords: Se eliminaron palabras vacías (stopwords) del español, como “el”, “la”, “que”, “de”, utilizando listas estándar del lenguaje provistas por bibliotecas como NLTK y spaCy, con ajustes personalizados para conservar ciertas palabras con carga política contextual.
- Lematización: Se redijeron las palabras flexionadas a su forma base o "lema" (su forma de diccionario), basándose en el significado de la palabra y su contexto para obtener una forma correcta al texto utilizando el modelo en español de spaCy. Por ejemplo, “votando”, “votar” y “votó” fueron reducidas a su lema “votar”. Esta etapa es fundamental para mejorar la generalización del modelo y reducir la dimensionalidad del vocabulario.
- Filtrado de tuits vacíos o irrelevantes: Tras el preprocesamiento, se descartaron aquellos tuits cuyo contenido quedó vacío o resultó ser genérico, irrelevante o ambiguo para fines de análisis político.

El conjunto final preprocesado mantuvo una representación textual coherente y limpia, lo que facilitó su tokenización posterior para ser ingresado en los modelos BERT. Esta limpieza y estandarización de datos permitió que el modelo aprendiera patrones significativos relacionados con la polaridad emocional del discurso político, minimizando interferencias derivadas de ruido textual.

3.3 Etiquetado de polaridad y tokenización con BERT

Una vez preprocesado el conjunto de datos, se procedió al etiquetado de la polaridad con el objetivo de convertir los textos en ejemplos supervisados para entrenar los modelos de clasificación. El etiquetado consistió en asignar a cada tuit una de tres categorías posibles: positiva, negativa o

neutral, de acuerdo con el sentimiento político expresado en el mensaje.El proceso de etiquetado combinó técnicas manuales y automáticas.

En una primera etapa, se seleccionó aleatoriamente una muestra representativa del corpus, la cual fue etiquetada manualmente por tres evaluadores independientes con conocimientos en análisis político y lingüístico. Se calculó el coeficiente Kappa de Cohen para determinar la concordancia interanotador y validar la calidad del etiquetado inicial. Esta muestra etiquetada se utilizó como base para entrenar un clasificador inicial basado en reglas semánticas y puntuaciones con herramientas de análisis de sentimientos como TextBlob y VADER, adaptadas al español. Posteriormente, este clasificador automatizado fue aplicado al resto del corpus, y se realizó una validación cruzada para corregir posibles errores. Una vez etiquetado el corpus, los textos fueron transformados en vectores utilizando el tokenizador de BERT. Este proceso incluyó los siguientes pasos:

- Tokenización[1]: Se utilizó el tokenizador correspondiente al modelo preentrenado seleccionado BERT. Este tokenizador aplica la técnica de WordPiece, que divide el texto en subpalabras optimizadas para el entrenamiento de transformers.
- Truncamiento y padding: Todos los textos fueron truncados o rellenados (padded) a una longitud máxima definida de 280 tokens, para garantizar uniformidad en el procesamiento por lotes (batches).
- Generación de máscaras de atención: Se generaron attention masks para que el modelo distinguiera entre tokens reales y tokens de relleno durante la inferencia.
- Codificación de etiquetas: Las etiquetas de polaridad (positiva, negativa y neutral) fueron transformadas a formato numérico mediante codificación one-hot o label encoding, según el tipo de salida requerido por el modelo.

Esta etapa fue clave para preparar adecuadamente el corpus para el entrenamiento de los modelos BERT, permitiendo que el sistema interpretara los textos no solo como secuencias de palabras, sino como representaciones contextuales profundas del lenguaje político emitido por los usuarios de Twitter.

3.4 Entrenamiento del modelo

Para la fase de entrenamiento, se seleccionaron dos modelos preentrenados basados en la arquitectura BERT: BERT base en español (BETO) y BERT base multilingüe con vocabulario en inglés. Ambos modelos fueron ajustados (fine-tuned) sobre el conjunto de datos etiquetado para la tarea específica de clasificación de polaridad (positivo, negativo, neutral) en tuits de contenido político.

Antes del entrenamiento, el corpus fue dividido en tres subconjuntos: 80% para entrenamiento, 10% para validación y 10% para prueba final, manteniendo el equilibrio entre clases en cada subconjunto. Esta partición permitió evaluar el rendimiento del modelo durante el proceso de ajuste y evitar el sobreentrenamiento (overfitting).

[1] La tokenización consiste en segmentar el texto en unidades más pequeñas (palabras), lo que permite un procesamiento más eficiente y facilita que los sistemas automáticos interpreten el lenguaje humano.

El entrenamiento se realizó utilizando la biblioteca Transformers de Hugging Face y el marco PyTorch, sobre una plataforma con aceleración por GPU (NVIDIA Tesla T4). Se empleó el optimizador AdamW, con una tasa de aprendizaje inicial de 2e-5, y se aplicó early stopping para detener el entrenamiento si no se observaba mejora en la validación tras cinco épocas consecutivas. El número total de épocas fue de 10, con un tamaño de lote (batch size) de 16. Durante cada iteración, se calcularon las métricas de desempeño: precisión (accuracy), precisión por clase (precision), recuperación (recall) y medida F1. Para manejar la posible desbalance de clases, se aplicó una ponderación de clases inversamente proporcional a su frecuencia.

3.5 Evaluación de resultados

Para medir la efectividad de los modelos entrenados, se aplicó una evaluación cuantitativa basada en métricas estándar de clasificación: precisión (accuracy), precisión por clase (precision), recuperación (recall), medida F1 y matriz de confusión. La evaluación se realizó sobre el conjunto de prueba (10% del total del corpus), que fue mantenido aislado durante todo el proceso de entrenamiento y validación para garantizar una estimación objetiva del rendimiento. Los resultados se presentan en la Sección 4.

4 Resultados

Los resultados del entrenamiento del modelo se presentan en la Tabla 1, donde se muestra el valor final de la exactitud (*accuracy*), métrica que refleja el porcentaje total de predicciones correctas realizadas por el modelo. Asimismo, se incluye la métrica de precisión (*precision*), que evalúa la capacidad del modelo para identificar correctamente instancias de una categoría específica, y la recuperación (*recall*), que indica la proporción de instancias relevantes que fueron efectivamente detectadas. En cuanto a la estrategia de registro de los datos durante el entrenamiento, se emplearon dos enfoques comunes: *steps* y *epochs*. La principal diferencia entre ambos radica en su frecuencia de actualización; mientras los *steps* actualizan los parámetros del modelo de manera continua a lo largo del proceso de entrenamiento, los *epochs* lo hacen una vez que se ha completado un ciclo completo sobre el conjunto total de datos.

Tabla 1. Tabla de resultados del entrenamiento del Modelo BERT base en español.

step	Loss	Accuracy	Precision	Recall
100	0.269	0.851	0.851	0.851
200	0.217	0.881	0.881	0.881
300	0.152	0.919	0.919	0.919
400	0.181	0.902	0.902	0.902
500	0.283	0.888	0.888	0.888
600	0.254	0.895	0.895	0.895
700	0.277	0.892	0.892	0.892

800	0.279	0.902	0.902	0.902
900	0.29	0.892	0.892	0.892
1000	0.209	0.905	0.905	0.905
1100	0.199	0.905	0.905	0.905

En la Tabla 2 se presentan los resultados del entrenamiento del modelo BERT base en inglés, a diferencia del experimento anterior, en esta ocasión se empleó una estrategia distinta para el registro del proceso de entrenamiento. Específicamente, se utilizó el enfoque basado en epochs. Esta metodología implica la actualización de los parámetros del modelo tras completar un ciclo completo sobre el conjunto de datos, lo cual permite evaluar su rendimiento de manera global en cada iteración.

Tabla 2. Tabla de resultados del entrenamiento del Modelo BERT base en inglés.

epochs	Loss	Accuracy	Precision	Recall
1	0.347	0.848	0.807	0.776
2	0.341	0.85	0.812	0.776
3	0.335	0.852	0.815	0.776
4	0.33	0.854	0.819	0.776
5	0.325	0.856	0.812	0.776

El modelo BETO (BERT base en español) mostró un rendimiento superior al modelo BERT base multilingüe en inglés, obteniendo una precisión global de 86.5%, mientras que el modelo en inglés alcanzó 81.4%. Este resultado confirma que el uso de un modelo preentrenado específicamente en español ofrece ventajas significativas en tareas lingüísticas sensibles al idioma y al contexto cultural. A nivel de clasificación por clase, ambos modelos presentaron un buen comportamiento en la detección de sentimientos positivos y negativos, aunque mostraron cierta dificultad para identificar correctamente la clase neutral, lo que suele ser habitual en tareas de análisis de sentimiento debido a su ambigüedad semántica.

En el caso de BETO, la medida F1 fue de 0.86 para la clase positiva, 0.90 para la clase negativa y 0.86 para la clase neutral, reflejando un buen equilibrio general. El modelo en inglés, en cambio, presentó una mayor caída en la clase neutral, con un F1 de 0.78. La matriz de confusión mostró que los errores de clasificación más frecuentes se daban entre clases adyacentes (por ejemplo, confundir tuits negativos con neutros), lo cual indica que en muchos casos los mensajes contenían ironía, ambigüedad o matices emocionales difíciles de capturar incluso para modelos avanzados.

De acuerdo con los resultados obtenidos es posible observar en el Diagrama las métricas de desempeño de los modelos BERT base en español e inglés. En esta comparación se aprecia que el modelo BERT base en español presenta un mejor rendimiento durante el entrenamiento en

comparación con su contraparte en inglés. Esta diferencia se debe a que cada modelo ha sido entrenado con vocabularios distintos, extraídos de corpus en diferentes idiomas. No obstante, esta variación no implica que alguno de los modelos sea menos eficaz, ya que ambos mantienen un alto nivel de desempeño dentro de sus respectivos contextos lingüísticos.

Estos resultados validan la efectividad del modelo BERT base en español (BETO) para tareas de clasificación de polaridad en español y evidencian la importancia de trabajar con modelos entrenados en el idioma específico del corpus. Además, refleja el potencial del análisis automatizado para extraer tendencias de opinión pública en contextos políticos digitales, lo que puede resultar útil para periodistas, analistas, sociólogos y estrategas de campañas electorales.

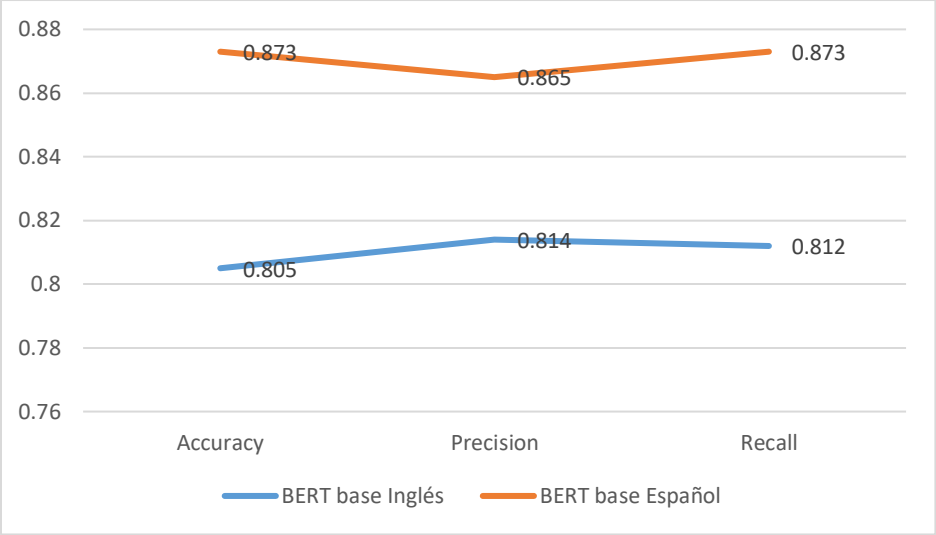


Figura 2. Comparativa de las métricas en modelos BERT base en español e inglés.

4.1 Discusión de resultados

Los resultados de esta investigación destacan el potencial de los modelos BERT para clasificar la polaridad en comentarios políticos en Twitter. El modelo entrenado en español (BETO) superó al modelo en inglés, con una precisión del 86.5% frente al 81.4%, evidenciando la importancia del contexto lingüístico en el rendimiento del modelo. BETO demostró mejor comprensión devocabulario y expresiones propias del discurso político en español.

No obstante, ambos modelos mostraron dificultades al clasificar tuits neutrales, debido a la ambigüedad y falta de marcadores claros, lo que sugiere la necesidad de incorporar enfoques más sensibles al tono y la ironía. También se identificaron limitaciones al interpretar tuits breves o con lenguaje informal, por lo que se recomienda enriquecer los modelos con elementos adicionales como emojis, hashtags o contexto conversacional. Pese a estas limitaciones, los modelos basados en BERT son herramientas valiosas para el análisis automatizado de opinión pública, con capacidad de escalar a grandes volúmenes de datos.

5 Conclusiones y Perspectivas

El desarrollo de modelos para la clasificación de la polaridad en comentarios políticos de Twitter utilizando BERT ha demostrado ser una estrategia efectiva para abordar el análisis de sentimientos en redes sociales. A través del preprocesamiento de datos y la aplicación de modelos preentrenados, fue posible construir clasificadores capaces de identificar con alta precisión la orientación emocional de los textos analizados. En particular, el modelo basado en BERT entrenado en español alcanzó una precisión del 86.5%, superando al modelo equivalente en inglés, que obtuvo un 81.4%. Esta diferencia reafirma la importancia de utilizar modelos adaptados al idioma y contexto específico del corpus. Los resultados evidencian el potencial del Procesamiento de Lenguaje Natural como herramienta para comprender el discurso político en medios digitales, lo cual representa un recurso valioso para líderes, analistas y tomadores de decisiones. El análisis de sentimientos puede contribuir significativamente a interpretar la opinión pública, mejorar la comunicación política y diseñar campañas más ajustadas a las percepciones sociales.

De acuerdo con los resultados obtenidos y las observaciones realizadas durante el desarrollo de este proyecto, se plantean diversas perspectivas para futuras investigaciones y aplicaciones prácticas. En primer lugar, es fundamental ampliar y diversificar el conjunto de datos, incorporando comentarios provenientes de distintos momentos, regiones y temáticas políticas, con el fin de fortalecer la capacidad del modelo para comprender un lenguaje más variado y mejorar su generalización. Asimismo, resulta recomendable explorar técnicas de optimización que permitan emplear variantes más ligeras y eficientes de BERT, como DistilBERT o ALBERT, favoreciendo su uso en entornos con recursos computacionales limitados.

Por otra parte, la inclusión de mecanismos que detecten matices lingüísticos complejos —tales como ironía, sarcasmo, ambigüedad y expresiones idiomáticas propias de las redes sociales— contribuiría a incrementar la precisión en la interpretación de comentarios. Además, se sugiere ampliar la aplicación de esta metodología a otros ámbitos como la educación, la salud, el comercio y la cultura, donde el análisis de opiniones digitales puede ser clave para la toma de decisiones estratégicas. Finalmente, este trabajo abre nuevas oportunidades para aplicar técnicas similares en otros ámbitos donde la interpretación de las emociones y opiniones expresadas en texto pueda tener un impacto estratégico, como la educación, la salud pública, el comercio o la gestión de crisis.

Referencias

[1] Arcila Calderón, C., Ortega, F., Amores, J. and Trullenque, S. (2017). Análisis supervisado de sentimientos políticos en español: clasificación en tiempo real de tweets basada en aprendizaje automático. El Profesional de la Información. 26. 973. 10.3145/epi.2017.sep.18.

[2] Cañete, J., Chaperon, G., Fuentes, R., and Pérez, J. (2020). Spanish Pre-Trained BERT Model and Evaluation Data. PML4DC at ICLR 2020. https://github.com/dccuchile/beto

[3] Devlin, J., Chang, M. W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 1(NAACL-HLT), 4171–4186. https://doi.org/10.48550/arXiv.1810.04805

[4] Giachanou, A., and Crestani, F. (2016). Like it or not: A survey of Twitter sentiment analysis methods. ACM Computing Surveys (CSUR), 49(2), 1–41. https://doi.org/10.1145/2938640

[5] He, L. (2024). Enhanced Twitter sentiment analysis with dual joint classifier integrating RoBERTa and BERT architectures. Frontiers in Physics, *12*, Article 1477714. https://www.frontiersin.org/articles/10.3389/fphy.2024.1477714

[6] Hutto, C., and Gilbert, E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. Proceedings of the International AAAI Conference on Web and Social Media, 8(1). https://doi.org/10.1609/icwsm.v8i1.14550

[7] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., and Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv preprint arXiv:1907.11692. https://doi.org/10.48550/arXiv.1907.11692

[8] Jiang, J., Ren, X., and Ferrara, E. (2023). Retweet-BERT: Political Leaning Detection Using Language Features and Information Diffusion on Social Networks. Proceedings of the International AAAI Conference on Web and Social Media, 17(1), 459-469. https://doi.org/10.1609/icwsm.v17i1.22160

[9] Marrapu, S., Senn, W., and Prybutok, V. (2024). Sentiment analysis of Twitter discourse on Omicron vaccination in the USA using VADER and BERT. Journal of Data Science and Intelligent Systems. https://doi.org/10.47852/bonviewJDSIS42022441

[10] Medhat, W., Hassan, A., and Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. Ain Shams Engineering Journal, 5(4), 1093–1113. https://doi.org/10.1016/j.asej.2014.04.011

[11] Mohammad, S. M., and Turney, P. D. (2013). Crowdsourcing a Word–Emotion Association Lexicon. Computational Intelligence, 29(3), 436–465. https://doi.org/10.1111/j.1467-8640.2012.00460.x

[12] Rahman, M. W. U., Shao, S., Satam, P., Hariri, S., Padilla, C., Taylor, Z., and Nevarez, C. (2022). A BERT-based Deep Learning Approach for Reputation Analysis in Social Media. 10.1109/AICCSA56895.2022.10017489.

[13] Renuka, O., and Radhakrishnan, N. (2024). BERT for Twitter sentiment analysis: Achieving high accuracy and balanced performance. Journal of Trends in Computer Science and Smart Technology, 6(1), 37–50. https://doi.org/10.36548/jtcsst.2024.1.003

[14] Schneider, N., Shouei, S., Ghantous, S., and Feldman, E. (2023). *Hate speech targets detection in Parler using BERT*. arXiv. https://arxiv.org/abs/2304.01179

[15] Song, C., Chen, S., Cai, X., and Chen, H. (2024). Sentiment analysis of Spanish political party tweets using pre-trained language models [Preprint]. arXiv. https://arxiv.org/abs/2411.04862

[16] Talaat, A.S. (2013). Sentiment analysis classification system using hybrid BERT models. J Big Data 10, 110. https://doi.org/10.1186/s40537-023-00781-w

[17] Tumasjan, A., Sprenger, T. O., Sandner, P. G., and Welpe, I. M. (2010). Predicting elections with Twitter: What 140 characters reveal about political sentiment. ICWSM, 10(1), 178–185. https://www.aaai.org/ocs/index.php/ICWSM/ICWSM10/paper/view/1441

[18] Yenduri, G., Rajakumar, B. R., Praghash, K., and Binu, D. (2024). Heuristic-assisted BERT for Twitter sentiment analysis. International Journal of Computational Intelligence and Applications. https://doi.org/10.1142/S1469026821500152

[19] Zhang, L., Wang, S., and Liu, B. (2018). Deep learning for sentiment analysis: A *survey* [Preprint]. arXiv. https://arxiv.org/abs/1801.07883

Caminemos Juntas

Let's walk together

Cecilia América Barranco Herrera, María del Carmen Santiago Díaz, Ana Claudia Zenteno Vázquez, Gustavo Trinidad Rubín Linares, Raúl Antonio Aguilar Vera, Abraham Sánchez López

Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla, 14 Sur y Av. San Claudio, C. U. Col. San Manuel. CP 72570. Puebla, México.

cecilia.barranco@alumno.buap.mx, {marycarmen.santiago, gustavo.rubin, ana.zenteno, a.sanchez}@correo.buap.mx

Abstract

Gender-based violence is a critical global issue, with alarming statistics in Mexico. Technology offers a viable channel for providing support to victims discreetly and safely. This article presents the development of “CAMI (Caminemos Juntas),” an intelligent management system designed to support women in situations of violence and help prevent it. The system integrates an interactive web portal with key prevention and detection resources, as well as a hybrid AI agent (chatbot). Its features include a risk assessment tool, an interactive map with emergency contacts, and access to legal and psychological counseling. The system was developed using the Scrum methodology, ensuring an iterative and user-centered approach. Usability test results show high satisfaction, validating the platform as an effective, accessible, and scalable tool for empowerment and prevention.

Resumen

La violencia de género es una problemática mundial crítica, con estadísticas alarmantes en México. La tecnología ofrece un canal viable para proporcionar apoyo a las víctimas de forma discreta y segura. Este artículo presenta el desarrollo de " CAMI (Caminemos Juntas)", un sistema gestor inteligente diseñado para apoyar a mujeres en situación de violencia y ayudar a prevenirla. El sistema integra un portal web interactivo con recursos clave de prevención y detección, así como un agente de IA (chatbot) híbrido. Entre sus funcionalidades destacan un instrumento de evaluación de riesgo, un mapa interactivo con contactos de emergencia y acceso a asesoría legal y psicológica. El sistema fue desarrollado bajo la metodología Scrum, asegurando un enfoque iterativo y centrado en la usuaria. Los resultados de las pruebas de usabilidad demuestran una alta satisfacción, validando la plataforma como una herramienta eficaz, accesible y escalable para el empoderamiento y la prevención.

Keywords and phrases: Violencia de Género, Chatbot, Inteligencia Artificial, Sistema de Apoyo para Mujeres. Empoderamiento Femenino.

1 Introducción

La violencia contra las mujeres y niñas es una de las violaciones de derechos humanos más extendidas a nivel global. Anualmente, miles de mujeres son asesinadas, y una de cada tres ha sufrido violencia física o sexual [1]. En México, el panorama es igualmente grave: se estima que 10 mujeres son asesinadas cada 24 horas y 7 de cada 10 han experimentado al menos un incidente de violencia a lo largo de su vida [2, 3].

A pesar de la magnitud del problema, menos del 40% de las mujeres que sufren violencia buscan ayuda en instituciones formales, a menudo por miedo, vergüenza o falta de acceso a recursos [4]. Es aquí donde la tecnología emerge como un medio facilitador para superar estas barreras, ofreciendo canales de comunicación discretos, accesibles y seguros.

Este trabajo presenta el diseño, desarrollo y validación de Caminemos Juntas, un sistema gestor inteligente que funciona como una plataforma de apoyo integral. El objetivo es ofrecer una herramienta tecnológica segura, centralizada y accesible que brinde recursos informativos, evaluación de riesgo y canalización a servicios profesionales para mujeres en situaciones de vulnerabilidad, con un enfoque especial en comunidades con acceso limitado a dichos servicios.

2 Marco Teórico y Estado del Arte

A continuación, se definen conceptos clave y se analiza el panorama de soluciones tecnológicas existentes.

2.1 Violencia de género

La violencia de género se define como todo acto que inflige daño físico, mental o sexual dirigido contra una mujer por el hecho de serlo [5]. Sus manifestaciones incluyen la violencia psicológica, física, económica, sexual y digital. Esta problemática sistémica requiere soluciones multifacéticas que no solo ofrezcan una reacción, sino también prevención y empoderamiento.

2.2 Tecnología como herramienta de apoyo

El uso de la tecnología para combatir la violencia de género ha ganado terreno a nivel mundial. Los Agentes de IA (chatbots), en particular, se han convertido en una herramienta valiosa por su capacidad de ofrecer una primera línea de ayuda anónima y disponible 24/7.

Se analizaron diversas soluciones existentes para identificar fortalezas y áreas de oportunidad:

- **Hello Cass (Australia):** Un chatbot basado en SMS que ofrece información y apoyo localizado de forma discreta, sin necesidad de una aplicación [6].
- **Robota (Latinoamérica):** Un chatbot de Acoso.online enfocado en asesorar a víctimas de pornografía no consentida, con contenido legal adaptado a varios países de la región [7].

- **Themis (México):** Implementado por la Red de Abogadas Violeta, este chatbot de Facebook Messenger ofrece asesoría jurídica gratuita y canaliza a las usuarias, permitiendo además la recolección de datos estadísticos para políticas públicas [8].
- **Violetta (México):** Un chatbot que, a través de preguntas interactivas, ayuda a identificar el nivel de violencia en una relación y redirige a psicólogos [9].

El análisis reveló que, si bien existen herramientas efectivas, muchas se centran en un único tipo de apoyo (legal, psicológico) o están limitadas a una plataforma específica. La oportunidad identificada fue crear un sistema integral que no solo combine estos servicios, sino que también incorpore herramientas de autoevaluación y recursos educativos en una plataforma web accesible y optimizada para conexiones de baja velocidad.

3 Arquitectura de la Aplicación

El proyecto se gestionó bajo la metodología ágil Scrum, permitiendo un desarrollo iterativo e incremental. Este enfoque facilitó la adaptación continua del sistema basándose en la retroalimentación de usuarias y expertas, asegurando que el producto final respondiera a necesidades reales.

3.1 Planificación y diseño

La fase inicial consistió en una investigación profunda para definir los requisitos del sistema, la cual incluyó:

- Análisis de estadísticas: Se revisaron datos de ONU Mujeres e INEGI para dimensionar la problemática.
- Entrevistas: Se dialogó con psicólogas, abogadas y mujeres en situación de riesgo para comprender sus necesidades.
- Definición de Requisitos:
    - Funcionales: Agente de IA para conversación, test de riesgo interactivo, mapa con recursos geolocalizados, acceso a asesoría profesional y recursos educativos.
    - No Funcionales: Disponibilidad 24/7, seguridad de datos (sin almacenamiento de información personal de las conversaciones), confidencialidad y accesibilidad en dispositivos móviles con bajo consumo de datos

3.2 Arquitectura de la aplicación

Se diseñó una arquitectura modular de tres capas para garantizar la escalabilidad y mantenibilidad del sistema.

El sistema proporciona una interfaz de usuario intuitiva, responsiva y dinámica. En el cual se priorizó un diseño limpio (UX/UI) para facilitar la navegación a usuarias con distintos niveles de habilidad tecnológica como se puede ver en la figura 1.

Figura 1. Home del sistema gestor inteligente CAMI.

Backend gestiona las solicitudes del cliente, la lógica y la comunicación con servicios externos. Es el intermediario entre el frontend y la inteligencia artificial, como se puede ver en la figura 2.

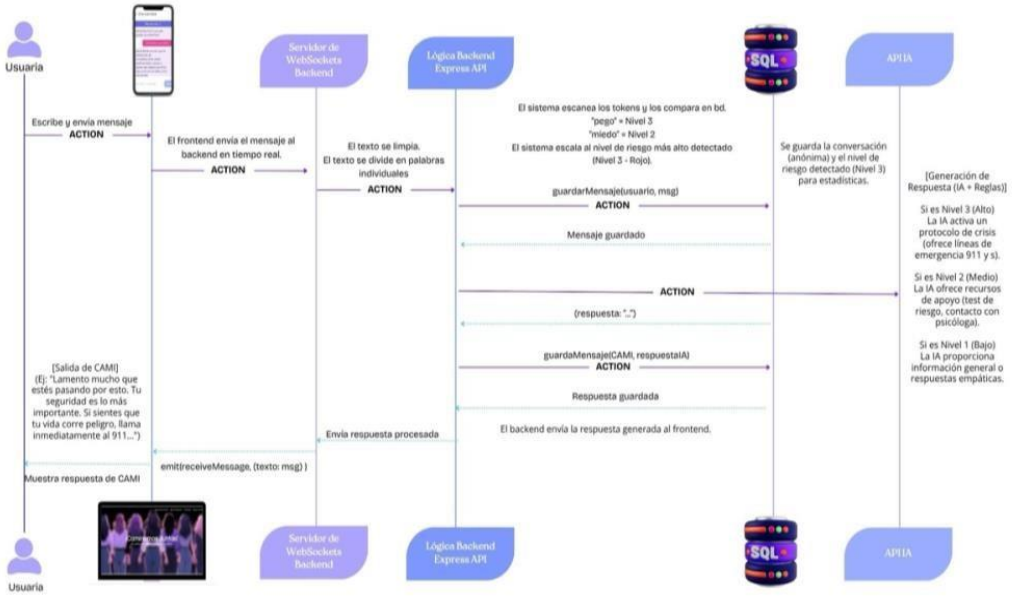


Figura 2. Arquitectura del sistema gestor inteligente CAMI.

Capa de Datos y Servicios Externos:

- Base de Datos: Se diseño una base de datos relacional para almacenar el contenido del portal como contactos de emergencia, centros de justicia por estado y el nivel de riesgo para garantizar la privacidad.
- Inteligencia Artificial: El núcleo del agente conversacional (Agente CAMI) el agente se desarrolló y entreno para asegurar que las respuestas sean empáticas, seguras y se mantengan dentro del contexto de apoyo. La comunicación con la IA, incluye una lógica de reintento exponencial para manejar errores de red.

3.3 Funcionalidades clave implementadas

Enseguida describimos los elementos de la implementación

- Agente de IA (CAMI): A través del Procesamiento de Lenguaje Natural (PLN), CAMI puede comprender las consultas de las usuarias, proporcionar información sobre tipos de violencia, derechos legales y recursos disponibles, y canalizar a servicios especializados de manera empática y segura, como se puede ver en la figura 3.

Figura 3. Agente inteligente “CAMI”.

- Test de Riesgo Interactivo, basado en la escala de Likert, este instrumento que permite a las usuarias autoevaluar su nivel de riesgo. Al finalizar, el sistema ofrece recomendaciones personalizadas según el resultado de acuerdo con los niveles obtenidos (Amarillo, Naranja o Rojo) como se muestra este árbol de decisión en la figura 4.

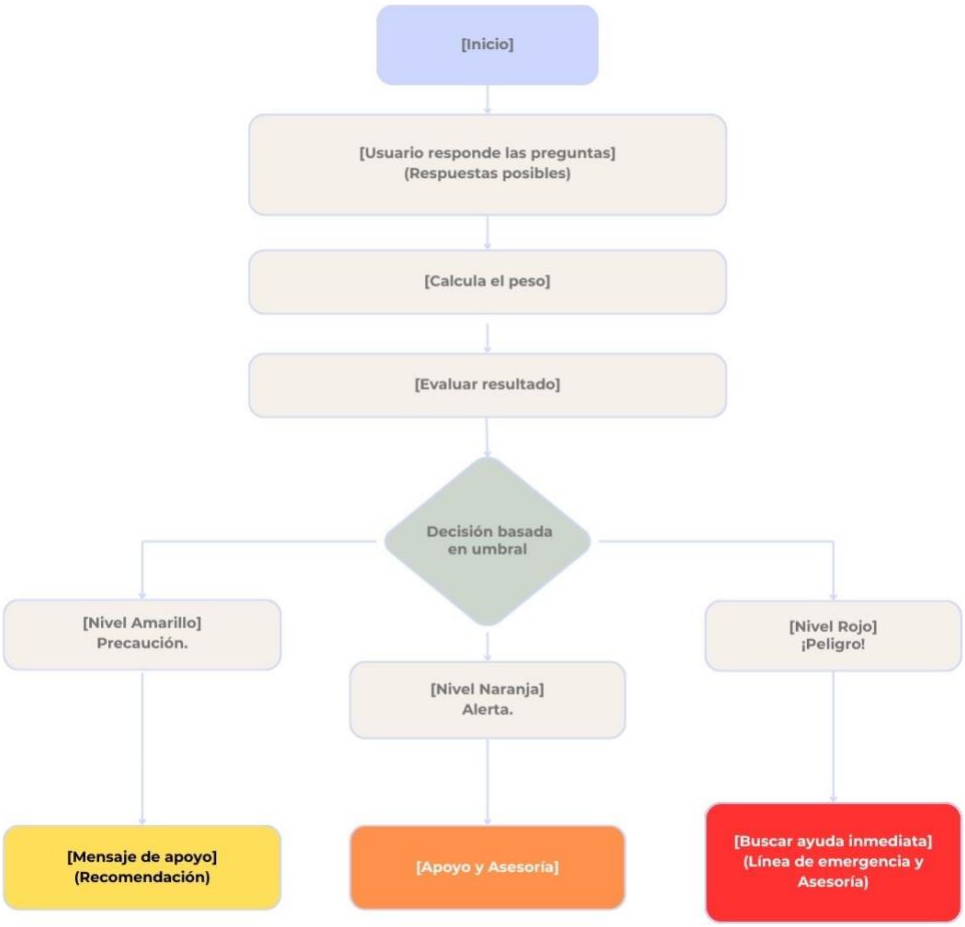


Figura 4. Árbol de decisión.

- Recursos de Empoderamiento: El portal web alberga una biblioteca de recursos multimedia (podcasts, artículos, noticias, videos y cursos gratuitos) diseñados para educar, prevenir la violencia y fomentar la autonomía económica y emocional.

4 Resultados

El sistema fue validado a través de pruebas de usabilidad con un grupo de 20 mujeres de diversos contextos (urbanos y rurales). Los resultados cuantitativos y cualitativos demostraron la efectividad y aceptación de la plataforma.

4.1 Comparativa con soluciones existentes

A diferencia de otras herramientas que se enfocan en un solo aspecto, CAMI ofrece una solución integral. En la Tabla 1 se resumen las principales mejoras.

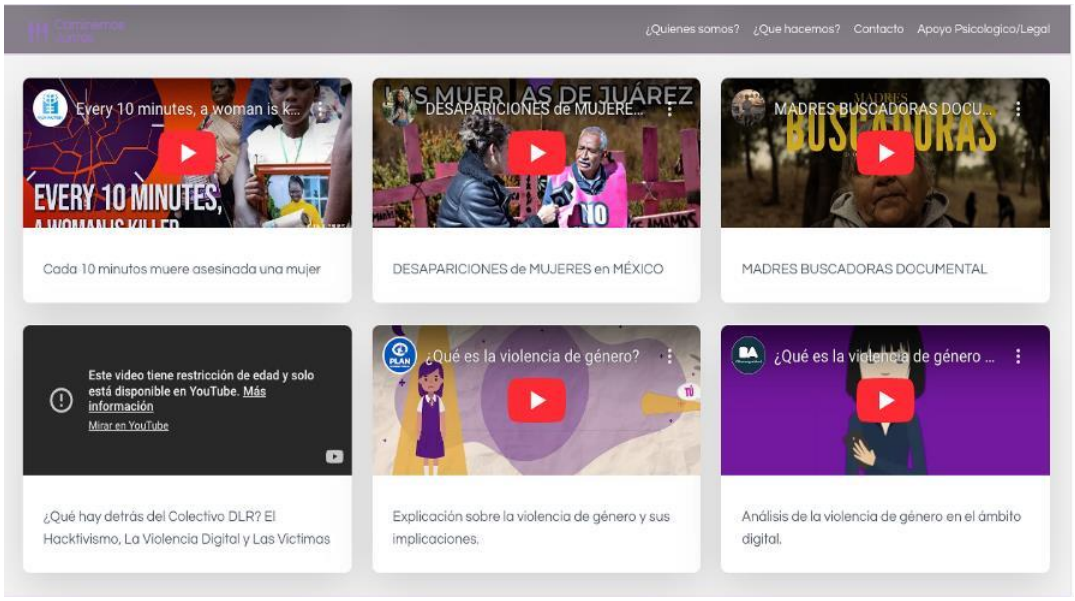


Figura 5. Recursos.

4.2 Pruebas de usabilidad y rendimiento

• Satisfacción del Usuario: El 95% de las participantes calificaron la interfaz como "intuitiva y fácil de navegar". El 85% consideró la interacción con el agente de IA CAMI como "muy útil", destacando la empatía y claridad de sus respuestas. El restante 15% lo calificó como “útil”.

• Rendimiento del Agente de IA: El tiempo de respuesta promedio del agente fue inferior a 5 segundos por consulta. El modelo demostró una alta capacidad para identificar la intención del usuario y proporcionar información relevante y segura.

• Impacto de las Herramientas:

○ El instrumento de riesgo fue completado por el 100% de las participantes, quienes afirmaron que les ayudó a reflexionar sobre su situación.

○ El mapa interactivo fue consultado por todas las participantes, y muchas descubrieron recursos de apoyo locales que desconocían.

4.3 Opiniones de las pruebas de usabilidad y rendimiento

Las opiniones cualitativas reforzaron los datos. Enseguida presentamos ejemplos de ello.

Usuaria X: *"CAMI me orientó de manera muy fácil y amable, era como hablar con una amiga. Me sentía atascada, pero me ayudó a ver opciones que no conocía"*.

Usuaria Y: *"El mapa interactivo me permitió encontrar un refugio cerca de mi casa. No sabía que existía"*.

Los testimonios validan que el sistema cumple su objetivo de ser una herramienta de apoyo cercana y efectiva.

Tabla 1. Comparativa funcional: CAMI vs. soluciones existentes

Característica	Soluciones Existentes	CAMI (Caminemos Juntas)
Enfoque	Generalmente unicanal (legal, psicológico o informativo).	Integral: combina apoyo emocional, legal, educativo y herramientas de autoevaluación.
Plataforma	Limitadas a apps de mensajería (FB Messenger, SMS, Telegram).	Plataforma web accesible desde cualquier navegador, optimizada para bajo consumo de datos.
IA	Basada en flujos predefinidos y palabras clave.	IA conversacional avanzada (Gemini) con comprensión de contexto y respuestas personalizadas.
Herramientas	Principalmente chat.	Herramientas Inteligentes: Agente IA, instrumento de riesgo, mapa interactivo, biblioteca de recursos.

5 Conclusiones y Trabajo a Futuro

El sistema gestor inteligente "Caminemos Juntas" ha demostrado ser una solución tecnológica viable, innovadora y efectiva para apoyar a mujeres en situación de violencia. La combinación de una arquitectura moderna, una metodología de desarrollo ágil y un enfoque centrado en las necesidades reales de las usuarias ha permitido crear una herramienta integral que destaca por su accesibilidad, seguridad y la calidad de sus recursos.

El proyecto no solo brinda una herramienta de respuesta inmediata, sino que también contribuye a la prevención y al empoderamiento a largo plazo. Los resultados positivos de las pruebas de

validación confirman que CAMI es más que un chatbot; es un espacio seguro y un primer paso para que muchas mujeres reconstruyan sus vidas.

Como trabajo a futuro, se plantean las siguientes líneas de acción:

1. **Expansión y Localización:** Integrar servicios más inclusivos, recursos locales de los estados de México, y adaptar el contenido a lenguas indígenas.
2. **Aprendizaje Automático:** Implementar un sistema de retroalimentación para que el modelo de IA aprenda y mejore continuamente a partir de interacciones anonimizadas.
3. **Aplicación Móvil:** Desarrollar una aplicación nativa (iOS/Android) que pueda incluir funcionalidades offline y un "botón de pánico".
4. **Análisis de Datos Agregados:** Recopilar datos anonimizados sobre los tipos de violencia más consultados o las regiones con mayor demanda para generar estadísticas que puedan informar políticas públicas, siempre garantizando la privacidad del usuario.

Agradecimientos
Los autores de este artículo agradecen por el apoyo otorgado en la realización de este trabajo a las mujeres usuarias y profesionales quienes nos asesoraron y apoyaron para que CAMI fuera una realidad.

Referencias

[1] UN Women. (2015). *Global database on Violence against Women*. UN Women Data Hub. https://data.unwomen.org/global-database-on-violence-against-women

[2] ONU Mujeres. (2021). *Un manual urgente para la cobertura de violencia contra las mujeres y feminicidios en México*. https://hchr.org.mx/wp/wp-content/uploads/2021/08/manual-periodistas-spotlight.pdf

[1] ONU Mujeres México. (2020). *COVID-19 y su impacto en la violencia contra las mujeres y niñas*. https://mexico.unwomen.org/sites/default/files/Field%20Office%20Mexico/Documentos/Publicaciones/2020/Abril%202020/COVID19_ViolenciaMujeresNinas_Generalabril2020.pdf

[2] unwomen. (2024, 27 de junio). *Datos y cifras: violencia contra las mujeres*. ONU Mujeres. https://www.unwomen.org/es/articulos/datos-y-cifras/datos-y-cifras-violencia-contra-las-mujeres

[3] ONU Mujeres. (2024, 27 de junio). *Preguntas frecuentes: Tipos de violencia contra las mujeres y las niñas*. ONU Mujeres. https://www.unwomen.org/es/articulos/preguntas-frecuentes/preguntas-frecuentes-tipos-de-violencia-contra-las-mujeres-y-las-ninas

[4] Hood, G. (s.f.). *Hello Cass*. https://hellocass.com.au/

[5] Acoso.Online. (s.f.). *Chat de ayuda*. https://acoso.online/cl/chat-de-ayuda/

[6] ChatBot, P. (2022, 16 de marzo). *Conoce a Themis, el chatbot que ofrece asesoría legal gratuita contra la violencia de género*. Chatbot Planeta. https://planetachatbot.com/themis-entrevista- chatbot-asesoria-legal-gratuita-contra-violencia-genero/

[7] Violetta. (s.f.). *Violetta*. Holasoyvioletta.com. Consultado el 25 de marzo de 2022.

Integración de Visión Artificial a un Brazo Robótico de 3 Grados

Computer Vision Integration with a 3-DOF Robotic Arm

Ángel Cruz Jiménez[1], Abel Alejandro Rubín Alvarado[2], Jonathan Axel Cruz Vazquez[2], Juan Humberto Sossa Azuela[2]

[1]Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Azcapotzalco. Av. De las Granjas 682, Santa Catarina, Azcapotzalco, 02550, Ciudad de México, CDMX.

[2]Centro de Investigación en Computación del Instituto Politécnico Nacional. Av. Juan de Dios Bátiz S/N, Nueva Industrial Vallejo, Gustavo A. Madero, 07700 Ciudad de México, CDMX.

acruzj2000@alumno.ipn.mx, arubina2022@cic.ipn.mx, hsossa@ipn.mx

Abstract

This work provides a rigorous technical and critical review of the integration of artificial vision systems with three-degree-of-freedom (3-DOF) robotic arms, focusing on applications in industrial automation. It explores in detail the hardware-software implementation strategies, real-time sensor fusion methodologies, and control algorithms, encompassing both classical frameworks and deep learning-enhanced approaches. Key computer vision components—including object detection, pose estimation, and 3D perception—are analyzed in the context of high-precision tasks such as pick-and-place and adaptive manipulation. Recent studies report accuracies exceeding 95% under industrial operating conditions, with sub-millimeter error margins. The review concludes with a discussion on current limitations and outlines emerging trends toward edge computing architectures, advanced multimodal fusion, and intelligent autonomous control.

Resumen

Este estudio presenta una revisión técnica y crítica sobre la integración de sistemas de visión artificial en brazos robóticos de tres grados de libertad (3-DOF), orientada a aplicaciones de automatización industrial. Se examinan en profundidad las estrategias de implementación hardware-software, los métodos de fusión sensorial en tiempo real y los algoritmos de control, destacando tanto los enfoques clásicos como los potenciados por aprendizaje profundo. Asimismo, se analizan los componentes clave de visión por computadora—detección de objetos, estimación de pose y percepción tridimensional—junto con su impacto en tareas de alta precisión como el pick-and-place y la manipulación adaptativa. Los estudios recientes reportan niveles de exactitud superiores al 95% en escenarios operativos industriales, con márgenes de error submilimétricos. Finalmente, se discuten las principales limitaciones y se delinean tendencias emergentes hacia arquitecturas de procesamiento en el borde, fusión multisensorial avanzada y control autónomo inteligente.

Keywords and phrases: Visión Artificial, Brazos Robóticos 3-DOF, Control Visual en Tiempo Real, Aprendizaje Profundo, Estimación de Pose, Pick and Place

1 Introducción

Este análisis investiga la integración de sistemas de visión artificial con brazos robóticos de tres grados de libertad (3-DOF) con fines de automatización industrial, abarcando métodos de implementación técnica, componentes de visión por computadora y aplicaciones en el sector industrial. Investigaciones recientes revelan avances sustanciales en metodologías de fusión sensorial, detección de objetos mediante aprendizaje profundo y algoritmos de control en tiempo real, con aplicaciones predominantes centradas en tareas de pick-and-place que alcanzan una precisión superior al 95% y exactitud dentro de rangos submilimétricos.

La convergencia de sistemas de visión artificial con brazos robóticos de tres grados de libertad (3-DOF) representa un avance crucial en el campo de la automatización industrial, facilitando la capacidad de los robots para percibir, interpretar e interactuar de forma autónoma con su entorno. Este análisis exhaustivo evalúa metodologías de vanguardia en tres dominios principales: estrategias de implementación técnica, componentes de visión por computadora y aplicaciones relevantes para la automatización industrial.

2 Marco Teórico y Estado del Arte

Las implementaciones contemporáneas de brazos robóticos 3-DOF emplean predominantemente microcontroladores Arduino para regular servomotores, donde cada articulación se acciona mediante comandos angulares precisos [1]. Esta arquitectura permite el procesamiento en tiempo real de datos visuales y su traducción en comandos motores, estableciendo una conexión fluida entre la percepción sensorial y las acciones resultantes.

Los sistemas modernos utilizan una variedad de modalidades sensoriales para mejorar las capacidades perceptuales. Investigaciones ilustran la combinación de cámaras, guantes de datos, sensores de fuerza y sensores inerciales (IMU), culminando en ecosistemas sensoriales integrales [2]. Configuraciones avanzadas integran sensores Kinect con IMU, utilizando técnicas de filtrado de Kalman para la fusión de datos, mejorando la precisión y estabilidad del seguimiento de movimiento [3].

Las metodologías más recientes de fusión sensorial combinan múltiples modalidades para reforzar la robustez del sistema. Estos sistemas integran cámaras RGB con sensores de profundidad (Intel RealSense D415, D435, Microsoft Kinect V2), proporcionando capacidades perceptivas tridimensionales completas [4, 5]. La integración de datos visuales y de profundidad facilita una estimación precisa de la pose de objetos y una comprensión espacial detallada.

Implementaciones avanzadas utilizan técnicas de filtrado de Kalman para fusionar datos sensoriales, mostrando eficacia particular al combinar datos de sensores Kinect e IMU para mejorar la precisión del seguimiento de movimiento [3]. Implementaciones basadas en FPGA alcanzan tiempos de

ejecución del orden de los 240 microsegundos, habilitando fusión sensorial en tiempo real con latencia mínima [6].

Los sistemas modernos de fusión sensorial emplean arquitecturas de procesamiento distribuidas, con la detección de objetos y la extracción de características ejecutadas en CPUs, mientras que el filtro de Kalman opera sobre hardware FPGA dedicado. Esta estrategia equilibra de forma óptima la complejidad computacional y las exigencias operativas en tiempo real [6].

Los mecanismos de control avanzados incorporan Model Predictive Control (MPC) para optimizar los torques de entrada en brazos robóticos 3-DOF. Las metodologías MPC permiten posicionamiento preciso y agarre de objetos minimizando funciones de costo y considerando restricciones dinámicas [7, 8]. Estos sistemas exhiben la capacidad de alcanzar posiciones objetivo con precisión mediante cálculos de cinemática inversa y transformaciones de coordenadas espaciales.

Los sistemas actuales utilizan metodologías de servo control visual basadas tanto en imagen como en posición. La investigación indica precisiones de seguimiento de 1.1 mm en movimientos uniformes y 8 mm en escenarios de velocidad variable, con una precisión de agarre superior al 96.67% en velocidades operativas inferiores a 20 m/s [9].

2.1 Control potenciado por aprendizaje profundo

Desarrollos recientes integran paradigmas de aprendizaje profundo con algoritmos de control convencionales. El modelo MPC-WGAN-Faster R-CNN combina conceptos de control predictivo con redes generativas adversariales de Wasserstein y tecnologías de reconocimiento visual Faster R-CNN para mejorar la precisión y adaptabilidad en entornos operativos complejos [10].

Las estrategias convencionales de detección de objetos emplean técnicas de segmentación basadas en color, particularmente el análisis en el espacio de color HSV (matiz, saturación, valor) para identificar y localizar objetos [11]. Bibliotecas como OpenCV y NumPy sirven como herramientas fundamentales para el procesamiento de imágenes y la identificación de colores en aplicaciones robóticas.

Las implementaciones contemporáneas aprovechan marcos avanzados de aprendizaje profundo, incluyendo YOLOv7 (You Only Look Once V7) en conjunto con redes generativas adversariales (GANs) para un reconocimiento rápido y preciso de objetos [8]. Estos sistemas alcanzan precisiones de reconocimiento del 95.2% a 29 fotogramas por segundo, demostrando capacidades de rendimiento en tiempo real [10].

Los sistemas de última generación implementan pipelines de detección de objetos capaces de gestionar múltiples objetos simultáneamente. La investigación indica que estos sistemas pueden detectar objetos bien texturizados mientras garantizan una estimación precisa de pose, incluso ante niveles tolerables de rotación fuera del plano [8].

3 Arquitectura del Sistema

El proyecto se diseñó bajo una arquitectura modular para la clasificación de objetos utilizando visión artificial e integración con un brazo robótico de 3 grados de libertad. La clasificación se basa en la identificación de 5 figuras geométricas diferentes: triángulo, rectángulo, trapecio, rombo y hexágono, con 6 colores diferentes. Cada combinación forma una categoría de clasificación diferente. El sistema fue desarrollado con un enfoque modular que permitió segmentar cada subsistema para facilitar la escalabilidad y depuración.

4 Sistema de Visión Artificial

El subsistema de visión artificial se encarga de la detección, segmentación y clasificación de figuras según su forma y color en tiempo real. Fue desarrollado utilizando Python 3 sobre una Raspberry Pi 3B+, la cual ofrece las posibilidades de procesamiento de datos como para el control de señales. La Raspberry Pi 3B+ incorpora un procesador ARM Cortex-A53 de 4 núcleos, adecuado para tareas de procesamiento de imágenes. Además, cuenta con pines GPIO que permiten generar señales de salida para actuadores, así conectar interfaces integradas para video, red y periféricos. Por lo anterior, la convierte en un elemento útil para el proyecto.

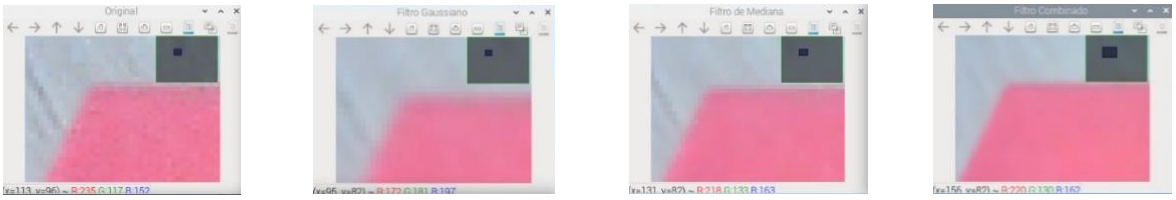
Durante la fase inicial, las imágenes capturadas por la webcam presentaban ruido. Por lo tanto, se aplicó un preprocesamiento que consistía en un filtro gaussiano de tamaño 3x3 con desviación estándar σ = 1.2 descrito por:

$$H(x,y) = \frac{1}{2\pi\sigma^2} * e^{-\frac{x^2+y^2}{2e^2}} \quad (1)$$

Sumado a un de un filtro de media de tamaño 3x3 descrito por:

$$H(i,j) = \frac{1}{(2k+1)^2} \quad (2)$$

Esta combinación buscaba suavizar la imagen, eliminar las irregularidades de iluminación y facilitar la detección de colores y contornos en etapas posteriores.



(a) (b) (c) (d)

Figura 1. Preprocesamiento de la imagen. (a) original, (b) Imagen con Filtro Gaussiano, (c) Imagen con Filtro Media, (d) Imagen con Filtro Combinado

Posteriormente, la imagen fue convertida del espacio RGB a HSV (Hue, Saturation, Value). Esta transformación permite una segmentación robusta frente a las variaciones del entorno como la

iluminación. En el espacio HSV la componente H representa el tono de color, S representa la saturación y V representa la intensidad.

Con la finalidad de obtener rangos para cada color, se desarrolló un script de calibración que permite visualizar y ajustar mediante deslizadores los limites superiores e inferiores de H,S y V. Esta herramienta fue empleada para determinar los rangos para cada color, garantizando que el sistema reconozca únicamente los objetos deseados, aun en presencia de fondos diversos.

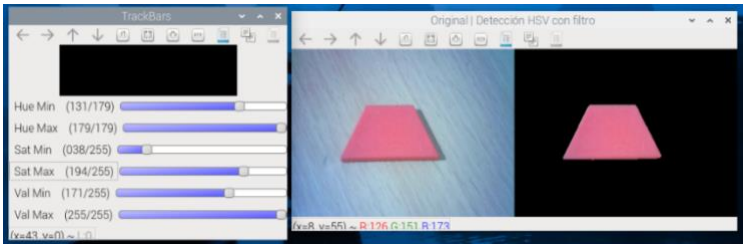


Figura 2. Interfaz gráfica del script de calibración HSV mostrando la delimitación de color.

Con los rangos obtenidos, se realizó la segmentación de la imagen mediante la generación de máscaras binarias, en las cuales los objetos con los colores objetivo fueron aislados del fondo. A partir de esta imagen binaria, se procedió a la extracción de contornos que consiste en identificar los limites de las regiones conectadas por los pixeles blancos de la imagen. Después, se aplicó una aproximación poligonal a cada contorno mediante la reducción de puntos, con la finalidad de simplificar la forma, pero conservando su geometría esencial y causando la extracción de características geométricas como el número de lados y ángulos.

Figura 3. Ejemplos de los Objetos a Clasificar

En el algoritmo de clasificación geométrica se aplicaron criterios de evaluación. El primer criterio se basa en el número de vértices de la figura. Para figuras de 4 vértices, se aplicaba un criterio basado en el análisis de los ángulos internos y la pendiente de sus lados. Por ejemplo, se distingue un cuadrado por sus 4 vértices y ángulos interiores aproximados a 90°, mientras que un trapecio es relacionado por sus lados paralelos, calculados mediante la comparación de pendientes. Asimismo, se distingue los 2 tipos de rombos según sus ángulos: uno con valores de 60° y 120° aproximadamente, y otro con 30° y 150° aproximadamente

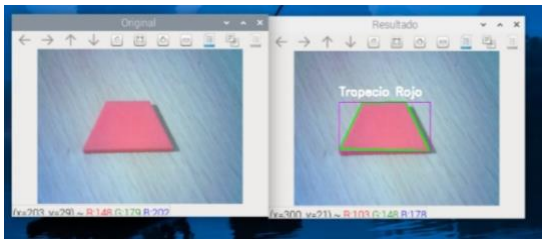


Figura 4. Ejecución del programa para la detección de formas y color.

Con el propósito de impedir errores de clasificación de las figuras, se agregó un mecanismo de validación temporal, Este mecanismo requiere que la detección de la figura se mantenga durante 4 segundos. Este umbral fue definido tras múltiples pruebas.

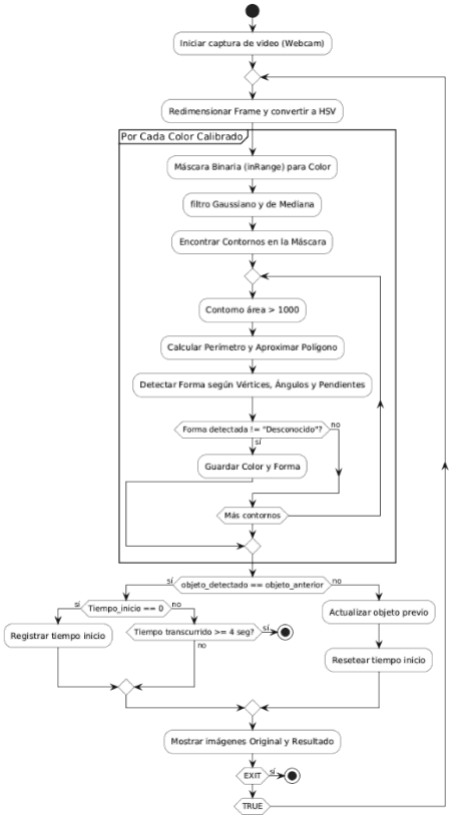


Figura 5. Diagrama de Flujo Ejemplo de la Ejecución del Sistema de Visión Artificial.

5 Sistema de Control

Al contar con la clasificación del objeto, el sistema ejecutara una secuencia de acción utilizando un brazo robótico HiWonder MaxArm de 3 grados de libertad. Este manipulador esta equipado con 3 servomotores tipo Bus HTS-35H, que se comunican por protocolo UART directamente con la Raspberry Pi, a través del pin GPIO 14 configurado como transición (Tx) y una bomba de sujeción con válvula.

Figura 6. Imagen del Brazo Robótico.

Para facilitar el control del brazo robótico y ser más preciso, se desarrolló una biblioteca personalizada denominada BusServo.py, que engloba funciones para el posicionamiento, ajuste de velocidad, asignación de identificadores y lecturas de estado. Con base en lo anterior, se implementaron funciones básicas de posicionamiento del brazo como fueron un posicionamiento inicial (HOME); desplazamiento de posicionamiento (READY), desplazamiento de recolección (PICK) y trayectorias especificas dependiendo la clasificación del objeto.

La estructura de programación se baso en los robots industriales. Principalmente en el modo JOIN de los robots industriales, donde se tiene total libertad de cada grado de libertad de manera independiente. Se creo las función move_servo con una estructura de 3 variables que definen: el identificador del motor, la posición y el tiempo en el que lo debe de hacer. Lo anterior con la finalidad de tener una perspectiva industrial y tener la posibilidad de escalar el proyecto aun nado a la modularidad del sistema.

a)

```
from Suction import nozzle_init, pick, place, pu

def home():
    move_servo(1, 723, 1500); time.sleep(2)
    move_servo(3, 556, 1500); time.sleep(2)
    move_servo(2, 540, 1200); time.sleep(2)

def pick_up():
    move_servo(1, 864, 1500); time.sleep(2)
    move_servo(3, 600, 1500); time.sleep(2)
    move_servo(2, 400, 1200); time.sleep(2)
    move_servo(3, 761, 1500); time.sleep(2)
    move_servo(2, 286, 1200); time.sleep(2)
    pick()
    move_servo(2, 400, 1200); time.sleep(2)
    move_servo(3, 600, 1500); time.sleep(2)

def ready():
    move_servo(1, 864, 1500); time.sleep(2)
    move_servo(3, 590, 1500); time.sleep(2)
```

b)

Figura 7. Imagen de la Programación del Brazo Robótico HiWonder (a) y un robot Fanuc (b)

En paralelo, se diseñó el subsistema de succión de la bomba de vacío y su válvula. Su funcionamiento fue englobado en la biblioteca desarrollada Suction.py, que permite activar y desactivar la bomba o abrir o cerrar la válvula con funciones como: suction_on(), suction_stop(), valve_on() y valve_off().

Figura 8: Imagen del la Bomba de Succión del Brazo Robótico.

Concluyendo en funciones compuestas para manipular el objeto sujetándolo y soltándolo (pick() y place()).

Con la detección, validación y clasificación del objeto, se ejecuta de forma automática la secuencia completa: el brazo se desplaza a la posición de recogida, activa la bomba para succionar la pieza, se dirige a la posición intermedia y finalmente se mueve al contenedor correspondiente de acuerdo con el color y forma. Luego de liberar el objeto, el brazo regresa a una posición inicial.

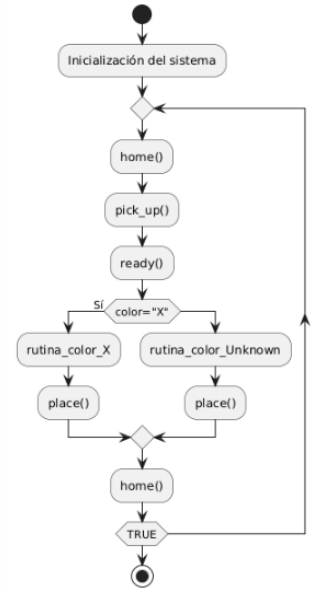


Figura 9. Diagrama de Flujo Ejemplo de la Ejecución del Sistema de Control.

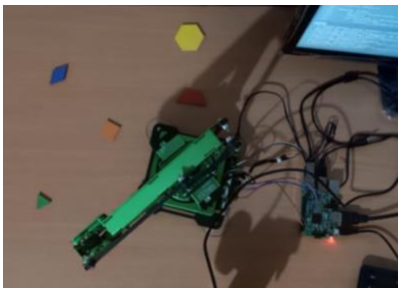


Figura 10. Posiciones del brazo robótico para la clasificación.

6 Implementación

El sistema se ejecutará dentro de un entorno Raspbian en la Raspberry Pi. La ejecución se realizará a través de la terminal, ya sea de forma local o remota.

Cada funcionalidad se encuentra en bibliotecas independientes. El módulo de visión.py contiene la visión artificial, la captura de imagen, el preprocesamiento, conversión a HSV, segmentación y clasificación de los objetos por parámetros de color y forma. La biblioteca robot.py contiene las funciones de control de posición del brazo, Suction.py controla la bomba y válvula, concluyendo en un código principal que integra todos los componentes bajo un ciclo principal de ejecución.

7 Pruebas

Para evaluar el desempeño del sistema, se diseñó una métrica de pruebas en un entorno controlado de laboratorio Las pruebas se realizaron sobre una mesa de trabajo con superficie de colores y textura irregular, con la finalidad de incrementar la dificultad en la segmentación de la imagen y evaluar la robustez del sistema ante fondos no uniformes.

Las condiciones de iluminación del entorno serán por dos lámparas, una de ellas con luz cálida y la otra con luz blanca, se harán dos pruebas una con cada tipo de Luz. La variación de luz causa variaciones espectrales diferente. Este escenario se estableció con el propósito de someter el sistema a condiciones parecida a entornos industriales donde la iluminación puede varia de acuerdo con el proceso o producto que se esté manejando.

El sistema de visión se implementó utilizando una cámara tipo USB. La cámara se fijo en posición paralela a la superficie y una distancia de 150mm al punto de trabajo, con el fin de garantizar uniformidad en la perspectiva y escala de los objetos durante todas las pruebas.

Las piezas para clasificar ya fueron especificadas anteriormente. El sistema fue configurado con un umbral de validación de 4 segundos de detección continua, de modo que únicamente se consideraban válidas las clasificaciones que permanecen estables durante ese periodo.

Se tomaron 40 piezas al azar para clasificar, evaluando los siguientes puntos:

- Precisión en la clasificación por color y forma (%): Parámetro que evalúa el número de aciertos respecto al número de intentos.
- Tiempo promedio de clasificación: tiempo transcurrido desde la puesta de la pieza hasta la confirmación estable de la clasificación, no considerando los 4 segundos.
- Tasa de éxito de pick-and-place (%): porcentaje de piezas que fueron correctamente sujetadas, trasladadas y depositadas en el contenedor, respecto al total piezas identificadas correctamente.
- Falsos positivos: número de objetos clasificados incorrectamente en una categoría no perteneciente.

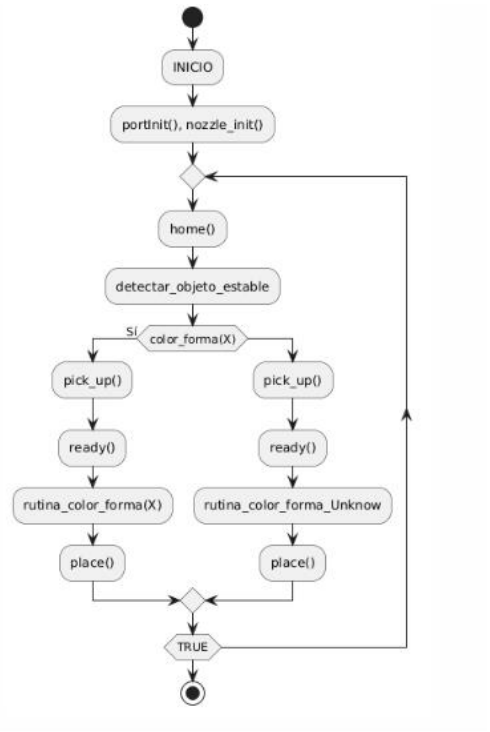


Figura 11. Diagrama de Flujo Ejemplo de la Ejecución del Sistema de Integrado.

Figura 12. Imagen del clasificado de piezas.

Este protocolo experimental busca no solo medir la eficacia del sistema bajo condiciones ideales, sino también evaluar su tolerancia a escenarios adversos que simulan aplicaciones industriales.

8 Resultados y Discusión

8.1 Resultados de desempeño

El desempeño del sistema se evaluó en un entorno controlado con condiciones de iluminación blanca y cálida, utilizando un conjunto de 40 piezas de prueba. La precisión de clasificación, el

tiempo promedio de detección y la eficiencia del subsistema de manipulación se registraron los siguientes datos.

Tabla 1. Resultados obtenido bajo condiciones de luz blanca.

Pieza	**Acierto de detección (%)**	**Tiempo promedio de clasificación (s)**	**Tasa de éxito de Pick and Place (%)**
TRAPECIO ROJO	100	1.22	100
ROMBO AZUL	100	1.13	100
ROMBO CREMA	88	2.47	86
CUADRADO NARANJA	100	2.22	100
HEXÁGONO AMARILLO	86	2.36	83
TRIANGULO VERDE	100	1.32	86
PROMEDIO	**96**	**1.79**	**92**

8.2 Análisis de desempeño y discusión

Los resultados obtenidos bajo condiciones de iluminación blanca, el sistema alcanzo un 96% de precisión promedio en la clasificación de objetos, con tiempos promedios de detección entre 1.13 y 2.47 segundos. Los casos con mayor dificultad se observaron en el hexágono amarillo y el rombo crema. Este comportamiento se relaciona principalmente a la reducida diferencia espectral entre ambos colores, además, el fondo de prueba era de tonalidad similar lo que causaba una perdida de contraste en el espacio de color HSV.

Bajo condiciones de iluminación cálida, el desempeño presento una disminución generalizada. La caída más significativa correspondió al rombo crema, teniendo un 75% porcentaje de acierto de detección, que además registro el mayor tiempo de clasificación siendo 3.51 segundos. El incremento en los tiempos promedio de clasificación respecto a los obtenidos con luz blanca confirma la sensibilidad del sistema a las variaciones espectrales causadas por las fuentes de iluminación. Este fenómeno se traduce en un retraso en la confirmación estable de detección.

Tabla 2. Resultados obtenidos bajo condiciones de luz cálida.

Pieza	Acierto de detección (%)	Tiempo promedio de clasificación (s)	Tasa de éxito de Pick and Place (%)
TRAPECIO ROJO	100	2.16	100
ROMBO AZUL	100	2.24	100
ROMBO CREMA	75	3.51	83
CUADRADO NARANJA	86	3.16	100
HEXÁGONO AMARILLO	86	3.20	100
TRIANGULO VERDE	100	2.14	86
PROMEDIO	**91**	**2.74**	**92**

En ambos casos, la textura irregular y heterogénea del fondo de prueba generó contornos que complicaron la aproximación poligonal de las figuras, provocando un incremento en los falsos positivos y afectando tanto la precisión como la eficiencia del sistema. Este efecto evidencia que, además de la variación cromática, la complejidad del fondo es un factor crítico que influye directamente en la robustez de la segmentación y la clasificación de objetos.

El subsistema de manipulación mostro una tasa de éxito en las operaciones superior al 90%, en ambos escenarios de prueba. Esto confirma que la arquitectura de control basa en funciones encapsuladas garantiza una ejecución confiable siempre que la clasificación inicial sea correcta. Sin embargo, en ambos casos se tuvieron problemas con la manipulación de rombo amarillo y rombo crema. El rombo amarillo cuenta con dimensiones mucho mayores a las demás piezas, asimismo, un mayor peso. Por otro lado, el rombo crema es la pieza de menores dimensiones de las piezas, lo que provoca que se requiera un mayor control de movimiento para su manipulación.

9 Conclusiones y Trabajo a Futuro

El proyecto antes descrito presento un sistema funcional que integra visión artificial integrado a un brazo robótico de 3 grados de libertad para la clasificación de objetos por color y forma en tiempo real. Lo anterior a través del uso de Raspberry Pi 3B+. y un brazo robótico, se logró implementar una arquitectura modular capaz de realizar detección estable y ejecutar movimientos automáticos para clasificar piezas de acuerdo con sus características.

En comparación a otros proyectos similares, este proyecto unifica la parte de visión artificial y de control en un solo elemento de procesamiento, es decir, no requiere un intermediario de microcontrolador para el desempeño de control del sistema, traducido en sistemas más complejos, con un mayor consumo energético y de costos.

La implementación de una arquitectura modular permitió segmentar el sistema en componentes independientes: visión y control del brazo robótico. Esta organización facilita la depuración del

código, al permitir aislar errores de manera específica en módulos específicos sin afectar el resto del sistema, de la misma manera, su modificación. Otra ventaja de este sistema es su posible escalabilidad donde se pueden reemplazar partes del sistema como una posible aplicación futura diferente de clasificación de objetos, sin necesidad de reescribir por completo todo el código. Este enfoque segmentado promueve el mantenimiento eficiente y adaptar el sistema a futuras nuevas funcionalidades o requerimientos.

A nivel de control, la lógica de programación para el proyecto sigue a una estructura similar a la utilizada en la rama industrial esto en posicione definidas como HOME, secuencias de movimiento determinadas por motor, posición y velocidad basadas en un modo JOIN, división del código por funcionalidad de la herramienta utilizada como pick() y place() reflejando una programación orientada a tareas con herramientas tal como suceden en las celdas de manufactura industriales donde se cuentan con periféricos y/o herramientas.

Sin embargo, el sistema actual depende de rutinas fijas, lo que limita demasiado su precisión a modificación en el entorno. Como propuestas de mejora se propone implementar cinemática inversa y un mayor análisis en la morfología del brazo robótico, permitiendo que el brazo calcule dinámicamente la posición necesaria para llegar a un punto establecido dentro del espacio de trabajo, dotando al sistema de mayor flexibilidad y adaptabilidad.

Además, se proyecta la integración de redes neuronales para reemplazar el enfoque clásico de detección por contornos, logrando una clasificación más robusta ante objetos deformados, cambios de iluminación o ruido en la imagen. Aunado a lo anterior, se plantea migrar el sistema a una FPGA como puede ser la Jetson Nano, y desarrollar interfaces graficas de control.

Las propuestas anteriores permitirán evolucionar el sistema actual hacia una solución más dinámica, escalable y autónoma, con posibles aplicaciones industriales o educativas.

Agradecimientos

Los autores de este articulo agradecen a la Secretaría de Ciencia, Humanidades, Tecnología e Innovación (Secihti) y al Centro de investigación en Computo (CIC), en especial al Laboratorio de Robótica y Mecatrónica por el apoyo durante el desarrollo de este proyecto.

Referencias

[1] Aburas, O., Hussein, M., Elnageh, A., Ebshish, M., & Algadi, M. (2024). Designing and implementing a 3DOF robotic arm for color sorting and object identification using computer vision technology. *International Journal of Engineering and Information Technology, 12*(1).

[2] Wang, S., Qi, W., & Su, H. (2023). Sensor fusion-based teleoperation control of anthropomorphic robotic arm. *Biomimetics, 8*(2), 169.

[3] Chen, F., Wang, F., Dong, Y., Yong, Q., Yang, X., Zheng, L., Gao, Y., & Su, H. (2023). Sensor fusion-based anthropomorphic control of a robotic arm. *Bioengineering, 10*(11), 1243.

[4] Niu, H., Ji, Z., Zhu, Z., Yin, H., & Carrasco, J. (2021). 3D vision-guided pick-and-place using Kuka LBR iiwa robot. *arXiv preprint*, arXiv:2102.10710.

[5] Niu, H., Ji, Z., Zhu, Z., Yin, H., & Carrasco, J. (2021). 3D vision-guided pick-and-place using Kuka LBR iiwa robot. In *Proceedings of the IEEE/SICE International Symposium on System Integration* (pp. 1–6).

[6] Yan, Z., Chang, Y., Yuan, L., Wei, F., Wang, L., Dong, X., & Han, H. (2024). Deep learning-driven robot arm control fusing convolutional visual perception and predictive modeling for motion planning. *Journal of Organizational and End User Computing, 36*(1).

[7] Maloum, A. M. H., Muchuka, N. M., & Kiruki, C. R. M. (2024). FPGA-based object tracking: Integrating deep learning and sensor fusion with Kalman filter. *Indonesian Journal of Electrical Engineering and Computer Science, 34*(2), 888–899.

[8] Paul, S. K., Chowdhury, M. T., Nicolescu, M., & Feil-Seifer, D. (2021). Object detection and pose estimation from RGB and depth data for real-time, adaptive robotic grasping. In *Robotics Research* (pp. 1–16). Springer.

[9] Nguyen, V. T., Nguyen, P. T., Su, S. F., Tan, P. X., & Bui, T. L. (2025). Vision-based pick and place control system for industrial robots using an eye-in-hand camera. *IEEE Access, 13*, 3536496.

[10] Lan, X., & Yang, S. (2024). Integrated application and optimized design strategies of automatic and servo control systems for industrial robots. *Applied Mathematics and Nonlinear Sciences, 9*(1).

[11] Wang, Y., Zhou, Y., Lai, W. C., & Li, R. (2023). Design of a four-axis robot arm system based on machine vision. *Applied Sciences, 13*(15), 8836.

Calibración de Crosstalk para Sensores ToF Multi-zona con Cubiertas Protectoras

Crosstalk Calibration for Multi-zone ToF Sensors with Protective Covers

Javier Martínez Fernández, Nicolas Quiroz Hernández, Ana María Rodríguez Domínguez, Roberto Olmos Pimentel

Benemérita Universidad Autónoma de Puebla, Av. San Claudio s/n, Cd. Universitaria, C.P. 72592, Puebla, México.

mf223470449@alm.buap.mx, nicolas.quirozh@correo.buap.mx, anamaria.rodriguez@correo.buap.mx, roberto.olmos@correo.buap.mx

Abstract

Multi-zone Time-of-Flight (ToF) sensors are an interesting technology for robotics and automation applications. However, their accuracy is affected when used with protective covers due to the phenomenon of optical crosstalk. This work presents a characterization and calibration method for the VL53L8CH sensor using a cover at different separation distances (0.25 mm, 0.5 mm, and 1 mm). The results show that, without proper calibration, measurement errors increase at separations of 0.5 mm or more. To solve this problem, a software calibration procedure was implemented that corrected the error caused by crosstalk, allowing the sensor's accuracy to reach levels close to its performance without a cover.

Resumen

Los sensores de Tiempo de Vuelo (ToF) multi-zona son una tecnología interesante para aplicaciones de robótica y automatización. Sin embargo, su precisión se ve afectada cuando se utilizan con cubiertas protectoras debido al fenómeno de diafonía óptica. Este trabajo presenta un método de caracterización y calibración para el sensor VL53L8CH utilizando una cubierta a diferentes distancias de separación (0.25 mm, 0.5 mm y 1 mm). Los resultados demuestran que, sin una calibración adecuada, los errores de medición aumentan en separaciones de 0.5 mm o más. Para resolver este problema, se implementó un procedimiento de calibración por software que logró corregir el error provocado por la diafonía, permitiendo que la precisión del sensor llegue a niveles cercanos a su rendimiento sin cubierta.

Keywords and phrases: ToF, VL53L8CH, Calibración de Sensores, Diafonía Óptica, Sensores Multi-zona.

1 Introducción

Los sensores de distancia basados en el principio de Tiempo de Vuelo (ToF) se han convertido en componentes fundamentales en una amplia gama de campos tecnológicos, incluyendo la robótica móvil, la automatización industrial y los sistemas de interacción humano-máquina. Su capacidad para proporcionar mapas de profundidad en tiempo real en un formato compacto y de bajo consumo de energía los hace ideales para tareas de navegación, evasión de obstáculos y reconocimiento de gestos. A pesar de sus ventajas, la integración de estos sensores en productos finales a menudo requiere el uso de una cubierta protectora para resguardarlos de condiciones ambientales adversas como polvo, humedad o impactos. La presencia de esta cubierta, aunque necesaria, introduce un problema físico significativo: un porcentaje de la luz emitida por el sensor es reflejada por la propia cubierta y regresa al receptor sin haber alcanzado el objetivo. Este fenómeno, conocido como diafonía óptica (*crosstalk*), introduce un error sistemático en la medición que puede afectar considerablemente la precisión del sensor.

El presente trabajo busca analizar este fenómeno en el sensor VL53L8CH y proponer una solución a este problema, centrándose específicamente en la caracterización y calibración cuando se utiliza con una cubierta protectora. Se analiza el impacto de la cubierta en las mediciones de distancia y se presenta un método de calibración para corregir el error inducido por la diafonía óptica.

2 Marco Teórico y Estado del Arte

2.1. Principio de funcionamiento: Tiempo de vuelo directo

La tecnología de sensores de Tiempo de Vuelo (*Time-of-Flight*, *ToF*) permite la medición de distancia a objetos en una escena. Dentro de las técnicas existentes, el método de Tiempo de Vuelo directo (dToF) es uno de los más utilizados en la industria. El principio de funcionamiento se basa en un sistema de iluminación activo. Una fuente de luz, comúnmente un láser que emite en el espectro infrarrojo (IR), proyecta un pulso hacia el objetivo. La luz se refleja en la superficie del objeto y una fracción de esta es captada de vuelta por un fotodetector en el sensor [Figura 1].

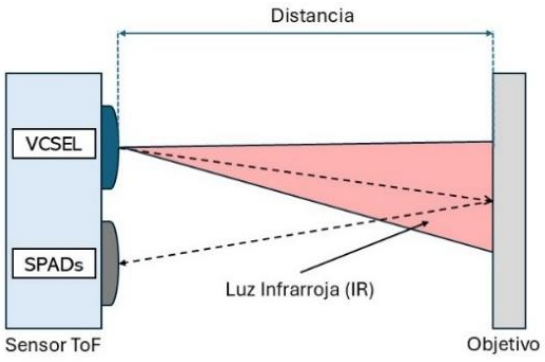


Figura 1. Principio de funcionamiento de los sensores de Tiempo de Vuelo Directo.

La distancia se determina midiendo el lapso entre la emisión del pulso de luz y la recepción de los fotones reflejados. Este intervalo, conocido como tiempo de vuelo (t), se relaciona directamente con la distancia (d) utilizando la siguiente ecuación 1:

$$d = \frac{c \cdot t}{2} \qquad (1)$$

donde:

d es la distancia al objeto.
c es la velocidad de la luz en el medio (aproximadamente $3 \times 10^8\ m/s$ en el aire).
t es el tiempo de vuelo medido.

El factor de $1/2$ es necesario porque el tiempo medido (t) corresponde al recorrido completo de la señal: desde el sensor hasta el objeto y de regreso [1], [2].

2.2. Sensor VL53L8CH: descripción y características

El VL53L8CH es un sensor de Tiempo de Vuelo multi-zona que integra todos los componentes necesarios en un único módulo compacto. Sus elementos principales incluyen un emisor láser VCSEL (*Vertical-Cavity Surface-Emitting Laser*) que opera a una longitud de onda de 940 nm, un arreglo de fotodetectores SPAD (*Single Photon Avalanche Diode*) como receptor, y un microcontrolador de bajo consumo de energía integrado para el procesamiento de las mediciones. A diferencia de los sensores de un solo punto, este sensor permite realizar un mapeo de profundidad al dividir su campo de visión (*Field-of-View, FoV*) en múltiples zonas independientes. Además, tiene una resolución programable que puede alcanzar un máximo de 64 zonas (en una configuración de 8x8), lo que permite la creación de un mini mapa de profundidad. Para la comunicación con un sistema anfitrión (*host*), el módulo soporta interfaces I²C (hasta 1 MHz) y SPI (hasta 3 MHz) [3].

2.3. Diafonía óptica en sensores de tiempo de vuelo

En muchas aplicaciones, es necesario proteger el sensor ToF de factores externos como el polvo, la humedad o los impactos físicos. Esto se logra mediante la instalación de una cubierta de protección, generalmente de cristal o acrílico, que debe tener una alta transmitancia para la luz infrarroja. Si bien la cubierta cumple una función de protección física y filtrado óptico, su presencia ocasiona un fenómeno conocido como: diafonía óptica (*crosstalk)* [4].

Este fenómeno se define como la porción de la luz emitida por el emisor que llega al receptor a través de trayectorias no deseadas, es decir, sin haber sido reflejada por el objetivo de interés en la escena. Esta señal de ruido no contiene información útil sobre la distancia y, si es lo suficientemente intensa, puede introducir errores significativos en las mediciones, como se muestra en la figura 2.

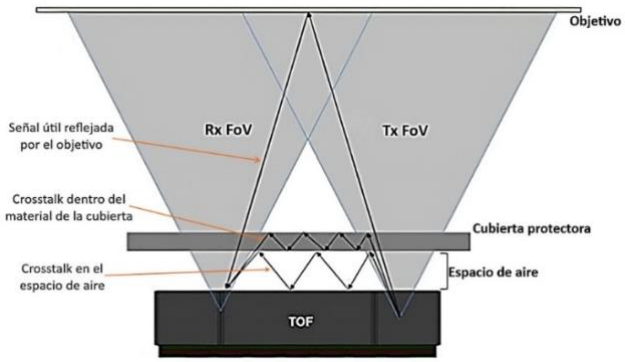


Figura 2. Efecto de diafonía óptica (*crosstalk*) provocado por una cubierta de protección.

Las principales trayectorias ópticas que generan diafonía óptica son:

- Reflexiones internas dentro del material de la propia cubierta protectora.
- Reflexiones en el espacio de aire que existe entre el sensor y la cubierta.

La magnitud de la diafonía está directamente influenciada por varios factores de diseño, como el grosor de la cubierta, el tamaño del espacio de aire y la calidad del material. Un nivel elevado de este efecto puede afectar el rendimiento del sistema, provocando una pérdida de la señal útil, falta de linealidad en las mediciones de distancia y un aumento en la desviación estándar de los resultados. Por esta razón, el diseño y la integración de la cubierta protectora deben realizarse cuidadosamente.

2.4. Método de primer contenedor significativo (FSB)

El método de Primer Contenedor Significativo (*First Significant Bin*, *FSB*) es una técnica utilizada en sensores ToF para procesar los datos crudos y estimar la distancia a un objeto. Sin embargo, el término no es exclusivo de los sensores de distancia, sino que es un concepto más amplio utilizado en el procesamiento de señales y el análisis estadístico en diversas áreas de la ciencia.

En el campo de la neurociencia, por ejemplo, este método se aplica a histogramas de tiempo peri-estímulo (PSTH) para determinar con precisión la latencia de la respuesta neuronal, identificando el primer intervalo temporal en que la tasa de disparo supera un umbral estadístico significativo [5], [6]. De manera análoga, en la física de alta energía, los sistemas ToF emplean un concepto equivalente para medir el tiempo de viaje de partículas subatómicas, discriminando la señal de llegada del ruido de fondo del detector [7]. En todos estos casos, el método resulta fundamental para identificar el inicio de un evento a partir de datos temporales discretos.

2.5. Trabajos relacionados

La tecnología de sensores de Tiempo de Vuelo, especialmente los módulos miniaturizados y de bajo costo como los de la serie VL53L_X, ha sido objeto de diversos estudios de caracterización y

aplicación. Estos trabajos se han centrado en evaluar su rendimiento en función de distintas variables para validar su uso en robótica, sistemas vestibles (*wearables*) y en otras aplicaciones.

Por ejemplo, Jans et al. [8] realizaron una caracterización del sensor VL53L1X, analizando su precisión, ruido y la influencia de la iluminación ambiental. De manera similar, Cheng y Mamishev [9] evaluaron el mismo sensor bajo diferentes condiciones de luz, temperatura y tamaño del objetivo para una aplicación de distanciamiento social. Otros autores, como Laković et al. [10] y Li et al. [11], han investigado la precisión de sensores ToF al interactuar con superficies de distintos materiales y colores, principalmente para la detección de entornos en robots móviles.

Si bien estos trabajos proporcionan un importante análisis para entender el rendimiento de los sensores ToF, hay una característica que no toman en cuenta, todos realizan las pruebas con el sensor expuesto directamente al entorno. En otras palabras, el problema específico de diafonía óptica y los procedimientos de calibración necesarios para corregirlo, no han sido revisados en la literatura encontrada.

3 Metodología Experimental

La calibración y caracterización del sensor se realizó en un entorno controlado, con una temperatura ambiente de ∼21 °C y una iluminación proporcionada por lámparas led en el techo del laboratorio (200 lm/m²), esto para garantizar la repetibilidad de las mediciones y minimizar las variables ambientales.

3.1. Montaje experimental y sistema de medición

El sistema de medición consiste en una placa de expansión X-NUCLEO-53L8A1, que integra el sensor multi-zona VL53L8CH, montada sobre una placa de desarrollo NUCLEO-F401RE que actúa como sistema anfitrión. La comunicación entre ambas placas se estableció mediante comunicación I²C. Para evaluar el efecto de diafonía óptica, se utilizó una cubierta de cristal y tres espaciadores de 0.25 mm, 0.5 mm y 1 mm.

El sistema de medición se fijó a un trípode [Figura 3-a], el cual a su vez se colocó sobre una plataforma de elevación [Figura 3-b]. Esta configuración permitió ajustar con mayor precisión la altura del sensor para centrar su campo de visión en el objetivo de medición (una pared plana y homogénea). Para garantizar la perpendicularidad entre el sensor y el objetivo se diseñó una estructura mediante impresión 3D equipada con tres punteros láser. La alineación se verificaba para asegurar que la separación de los puntos láser proyectados en la pared fuera consistente [Figura 3-c]. Una vez confirmada la alineación, los láseres se apagaban durante la recolección de datos para evitar cualquier tipo de interferencia con las mediciones del sensor ToF.

(a) **(b)** **(c)**

Figura 3. Estructura experimental. (a) Sistema de medición. (b) Elevador. (c) Punteros laser.

3.2. Extracción de datos para calibración

El proceso de recolección de datos para calibración de diafonía sólo se realiza una vez para cada nueva separación de la cubierta. Para esto, se deben considerar los siguientes criterios:

- El objetivo de referencia debe ser plano, con una reflectividad baja y conocida.
- El objetivo debe colocarse a una distancia conocida y fija del sistema de medición. La distancia mínima es de 600 mm y debe cubrir todo el campo de visión del sensor.
- El procedimiento debe realizarse en las condiciones de operación finales, es decir, con la cubierta protectora y el espaciador ya instalados.

Los parámetros de operación del VL53L8CH se establecieron de la siguiente manera:

- Modo de operación: Autónomo
- Resolución: 8x8 (64 zonas)
- Frecuencia de medición: 15 Hz

- Tiempo de integración: 20 ms
- Parámetros de histograma (CNH): 18 bins con un factor de agrupamiento (*binning*) de 1

Se recolectaron 10 muestras consecutivas para promediar y analizar la estabilidad de la lectura. Posteriormente, se repitió el mismo procedimiento de medición para cada una de las tres configuraciones de espacio de aire, utilizando los espaciadores de 0.25 mm, 0.5 mm y 1 mm. La recolección de datos fue automatizada mediante un *script*, siguiendo la lógica que se muestra en la figura 4. La magnitud del error introducido se guarda dentro del mismo dispositivo para su posterior calibración.

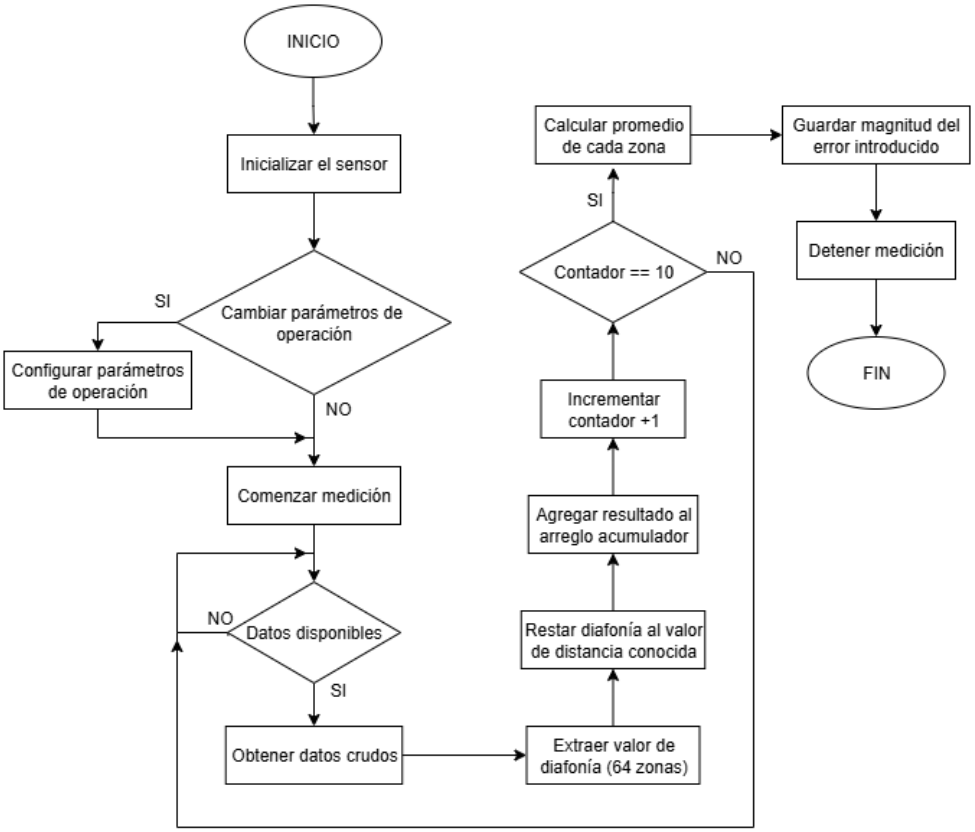


Figura 4. Diagrama para obtener y guardar datos de calibración.

3.3. Calibración de diafonía óptica

El firmware del sensor tiene una inmunidad inherente a la diafonía. Sin embargo, en la mayoría de los casos no es suficiente para obtener una buena medición. Para lo cual, es necesario aplicar una calibración por software.

Una vez que se conoce la magnitud de la señal de ruido que la cubierta protectora refleja hacia el sensor se puede restar matemáticamente de las mediciones futuras, permitiendo que solo la señal reflejada por el objetivo real sea considerada para el cálculo de la distancia. Adicional a esto, la luz

ambiente se considera como un valor umbral para hacer aún más efectiva la detección de distancia en los histogramas, el diagrama de flujo para la calibración se muestra en la figura 5.

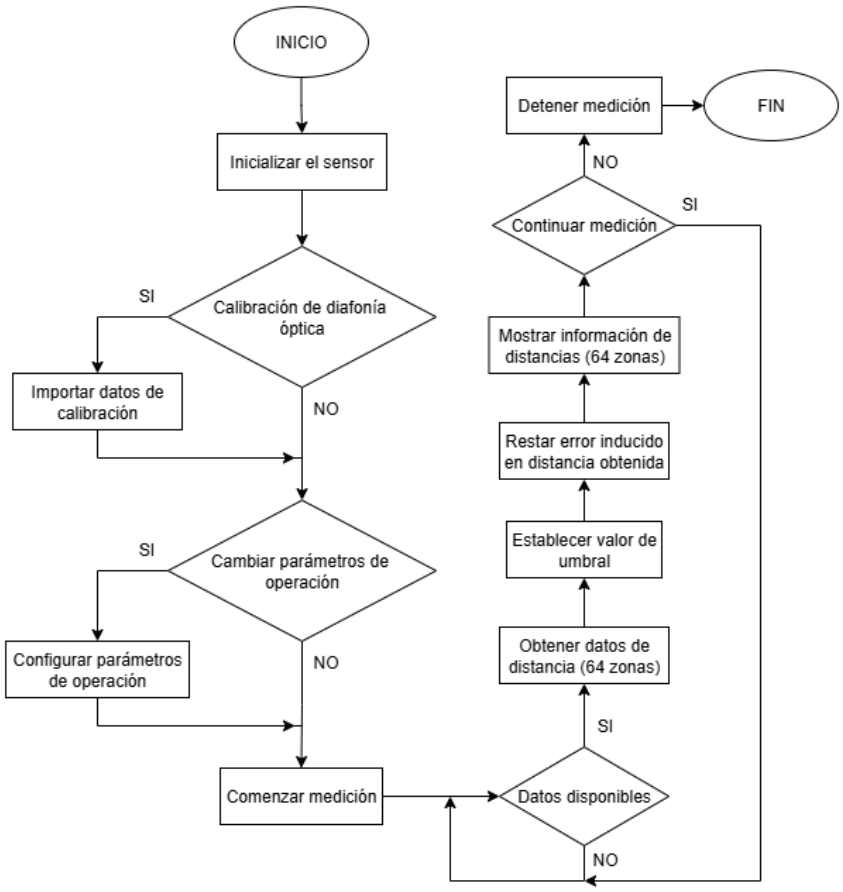


Figura 5. Diagrama para calibración de diafonía óptica.

Para analizar el impacto de la cubierta protectora y evaluar la eficacia de la calibración, se realizó una medición de línea base sin la cubierta y mediciones posteriores con la cubierta instalada a diferentes distancias del sensor. En todas las mediciones se mantuvo la misma configuración del sensor con el fin de asegurar la consistencia de los resultados.

Se realizó un barrido de mediciones de distancia sin la cubierta protectora. Se midió con un rango desde 200 mm hasta 4000 mm en incrementos de 200 mm. Al igual que en la recolección de datos de calibración, en cada punto de distancia se obtuvieron 10 muestras consecutivas. Este procedimiento se repitió para cada uno de los escenarios experimentales; espaciadores de 0.25 mm, 0.5 mm y 1 mm, con calibración y sin calibración.

4 Resultados y Discusión

En esta sección se presentan y analizan los resultados obtenidos de los seis escenarios experimentales. El análisis se centra en cuantificar el impacto de la cubierta protectora en la precisión de las mediciones y en evaluar la efectividad del procedimiento de calibración de diafonía.

4.1 Efecto de diafonía óptica en el sistema sin calibrar

Para establecer una referencia del rendimiento óptimo, el sensor fue evaluado inicialmente sin la cubierta protectora. La línea de datos para el caso sin cubierta (línea base) se encuentra muy cercana a la línea de medición ideal, lo que indica una buena linealidad y precisión en todo el rango de medición de 200 mm a 4000 mm.

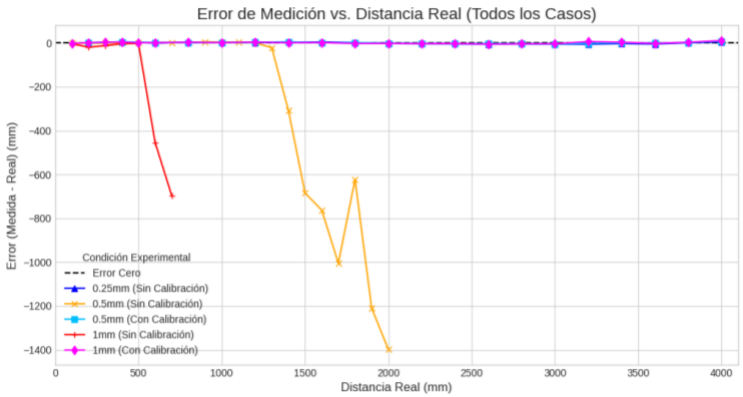


Figura 6. Gráfica del error de medición vs distancia real del sensor VL53L8CH.

Al añadir la cubierta con el espaciador más pequeño de 0.25 mm, el rendimiento se mantuvo prácticamente idéntico al de la línea base. Esto sugiere que, para espacios de aire mínimos, la compensación automática del firmware es suficiente para corregir la interferencia. Sin embargo, el impacto de este se vuelve relevante al aumentar el espacio de aire. Para los casos no calibrados de 0.5 mm y 1 mm, se observa una desviación importante a partir de aproximadamente 700 mm y 1400 mm, respectivamente. En la figura 6, se muestra este efecto, señalando cómo el error crece exponencialmente hasta reportar datos arbitrarios, haciendo que el sensor no funcione correctamente para mediciones de rango medio y largo.

4.2 Eficacia de la calibración de diafonía óptica

Después de aplicar la calibración de diafonía, como se observa en la figura 7, las mediciones para los casos que habían presentado mayores problemas (0.5 mm y 1 mm) han reportado mejoras significativas en los datos, presentando un comportamiento muy similar a la prueba base (sin cubierta y sin calibración) y la prueba 0.25 mm sin calibración.

Para analizar con mayor detalle la precisión obtenida, en la figura 8 se presenta una gráfica de barras de error para los casos con mejor rendimiento. Se puede observar que, después de la calibración, los casos de 0.5 mm y 1 mm, presentan un error promedio alrededor de 1.2 mm. En comparación a los otros dos casos, la longitud de las barras de error indica una variabilidad en las mediciones, lo que sugiere que, aunque la calibración corrige el error sistemático, la precisión del sensor aún presenta problemas al aumentar la separación de la cubierta protectora.

Estos resultados demuestran que, si bien la adición de una cubierta protectora puede afectar notablemente el rendimiento de los sensores ToF, un procedimiento de calibración por software es una solución efectiva para restaurar la precisión y fiabilidad del sistema de medición.

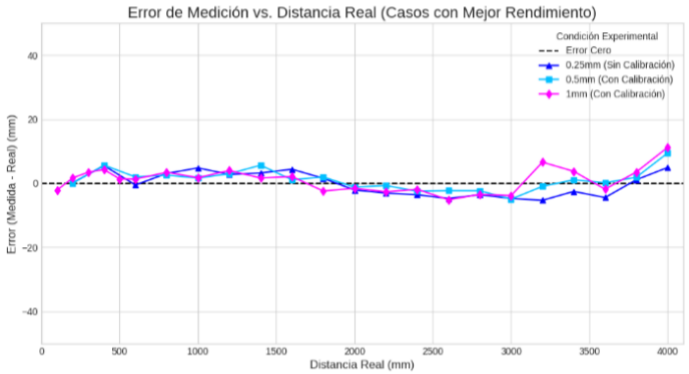


Figura 7. Gráfica del error de medición vs distancia real del sensor VL53L8CH.

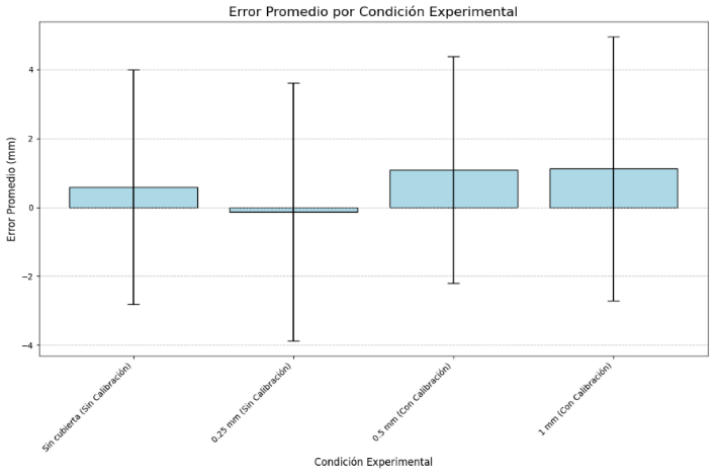


Figura 8. Gráfica del error de medición con los casos con mejor rendimiento.

5 Conclusiones

La calibración manual por software demostró ser una solución práctica y computacionalmente eficiente, restaurando la precisión del sensor a niveles casi idénticos a su funcionamiento sin cubierta. A diferencia de métodos que proponen la supresión de la diafonía a nivel de hardware [12] o que utilizan complejos modelos de forma de onda completa con un alto costo computacional, nuestro enfoque de sustracción directa es ideal para sistemas embebidos con recursos limitados. Los resultados validan que, para corregir el error sistemático introducido por una cubierta protectora, no es indispensable recurrir a algoritmos complejos, logrando una efectividad notable con una implementación sencilla.

Se concluye también que una calibración estática, como la implementada, corrige eficazmente la componente principal del error por diafonía. Sin embargo, la mayor variabilidad observada en las mediciones con espaciados de 0.5 mm y 1 mm sugiere la existencia de efectos secundarios menores. Esto es consistente con investigaciones que proponen correcciones dinámicas que se ajustan a la distancia del objetivo [13]. Si bien nuestro método es suficiente para la mayoría de las aplicaciones,

esto abre una línea de trabajo futuro para explorar calibraciones adaptativas que podrían refinar aún más la precisión del sensor en configuraciones con mayores espacios de aire.

Agradecimientos

Los autores de este artículo agradecen a la Secretaría de Ciencia, Humanidades, Tecnología e Innovación (SECIHTI) y a la Benemérita Universidad Autónoma de Puebla por el apoyo otorgado en la realización de este trabajo.

Referencias

[1] G. Di Giulio, et al., "Modeling and Analysis of a Direct Time-of-Flight Sensor Architecture for LiDAR Applications," Sensors, vol. 19, no. 24, p. 5464, 2019.

[2] F. Heide, et al., "Full-Waveform Modeling for Time-of-Flight Measurements based on Arrival Time of Photons," arXiv preprint arXiv:2208.03426, 2022.

[3] STMicroelectronics, "VL53L8CH Datasheet (DS14310)," Nov. 2024.

[4] STMicroelectronics, "AN5939 - Guidelines for the cover glass of the VL53L8 Time-of-Flight (ToF) multizone sensor family," Application note, Rev 2, Oct. 2023.

[5] S. Ciocchi *et al*., “Encoding of conditioned fear in central amygdala inhibitory circuits,” *Nature*, vol. 468, no. 7321, pp. 277–282, Nov. 2010, doi: 10.1038/nature09559.

[6] F. E. Theunissen, “An Auditory Memory System for Individual Vocal Recognition in the Zebra Finch,” Ph.D. dissertation, Univ. of California, Berkeley, CA, USA, 1994. [Online]. Available: https://escholarship.org/uc/item/4db738vh

[7] A. Sánchez-Hernández, “Determinación de Δmd y calibración absoluta del etiquetado de sabor para la medida de Δmd, en desintegraciones reconstruidas completamente en el experimento CDF,” Ph.D. dissertation, Centro de investigación y de Estudios Avanzados del I. P. N., México, D. F., 2005. Report No. FERMILAB-THESIS-2005-27.

[8] R. M. Jans, A. S. Green, and L. J. Koerner, "Characterization of a Miniaturized IR Depth Sensor with a Programmable Region-of-Interest That Enables Hazard Mapping Applications," IEEE Sensors Journal, vol. 20, no. 10, pp. 5213-5220, May 2020.

[9] X. Cheng and M. Mamishev, "Characterization of a Wearable LiDAR Sensor for Social Distancing," in 2022 IEEE IAS Global Conference on Emerging Technologies (GlobConET), 2022, pp. 115-120.

[10] N. Laković, et al., "Application of low-cost VL53L0X ToF sensor for robot environment detection," in 18th International Symposium INFOTEH-JAHORINA, 2019, pp. 1-4.

[11] Z. Li, J. H. Marsh, and L. Hou, "High Precision Laser Ranging Based on STM32 Microcontroller," 2020 IEEE International Conference on Mechatronics and Automation (ICMA), 2020.

[12] P. Pokhil et al., "A 40nm 8-Pixel 4-Tap 100-MS/s 1.7-GS/s-Equivalent-Rate Time-of-Flight Range Sensor with Crosstalk-Suppressed Histogramming ADC for Automotive LiDAR," in 2021 IEEE International Solid-State Circuits Conference (ISSCC), San Francisco, CA, USA, 2021, pp. 1-3.

[13] K. Kil, D. Kwon, J. Lee, J. Hur, and E. S. Jung, "A Study on the Compensation of ToF Sensor Crosstalk Noise for Various Distances," in 2020 International Conference on Consumer Electronics - Asia (ICCE-Asia), Seoul, Korea (South), 2020, pp. 1-3.

Desarrollo de una Red Inteligente de Monitoreo de Calidad del Aire para la Ciudad de Puebla

Development of a Smart Air Quality Monitoring Network for the City of Puebla

Ángel José González Pineda[1], Mariano Larios Gómez[1], Marco A. Mora Ramírez [2, *]

[1]Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla.

[2]Facultad de Ciencias Químicas, Benemérita Universidad Autónoma de Puebla

*Correspondence to: marco.morar@correo.buap.mx

Abstract

This work presents the design, development, and implementation of a functional prototype for a low-cost monitoring network that measures particulate matter (PM_1, $PM_{2.5}$, PM_4, PM_{10}), temperature, and humidity in the city of Puebla. The solution is based on Low-Cost Devices (LCDs) rooted in the Internet of Things (IoT) concept, and using the ESP32 development board and a particulate matter sensor. Data is transmitted to a cloud architecture on Amazon Web Services (AWS), which manages the ingestion, storage, and processing of the information. Functional device prototypes were obtained, ready for a pilot deployment in the city of Puebla, and the base cloud services have already been established. This project aims to contribute to the democratization of environmental monitoring technology by providing a data port for open science and generating information that complements health studies, supporting the creation of public policies in the city of Puebla.

Resumen

Este trabajo presenta el diseño, desarrollo e implementación de un prototipo funcional para crear una red de monitoreo de bajo costo para medir las concentraciones de material particulado ($PM_{1.0}$, $PM_{2.5}$, PM_4, PM_{10}), temperatura y humedad en la ciudad de Puebla. La solución se basa en dispositivos de bajo costo (DBC) basados en el concepto del Internet de las Cosas (IoT) que utilizan una placa de desarrollo ESP32 y un sensor de partículas. Los datos son transmitidos a una arquitectura en la nube sobre Amazon Web Services (AWS), que gestiona la ingesta, almacenamiento y procesamiento de la información. Se obtuvieron prototipos funcionales de los dispositivos, listos para un despliegue piloto en la ciudad de Puebla y los servicios base ya establecidos en la nube. Este proyecto pretende contribuir a la democratización de la tecnología de monitoreo ambiental y de calidad del aire proveyendo un puerto de datos para la ciencia abierta y generando información que puede complementar estudios de salud y apoyar la creación de políticas públicas en la ciudad de Puebla.

Keywords and phrases: Arquitectura de Software, Low-cost Sensors, IoT, Particulate Matter, Cloud Computing.

1 Introducción

La contaminación del aire representa una de las prioridades sanitarias más importantes a nivel mundial. De acuerdo con datos de la Organización Mundial de la Salud (OMS) la gran mayoría de la población que habita zonas urbanas está expuesta a niveles de contaminación que exceden las recomendaciones [1]. Uno de los contaminantes más nocivos en la atmosfera es el material particulado (PM), especialmente las partículas finas ($PM_{2.5}$) y ultrafinas ($PM_{1.0}$), que pueden penetrar profundamente en el sistema respiratorio y circulatorio, causando graves problemas de salud, incluyendo enfermedades cardiovasculares y respiratorias [2,3]. El monitoreo de la calidad del aire se realiza tradicionalmente mediante redes de estaciones equipadas con instrumentos de nivel regulatorio [4]. Sin embargo, el alto costo de adquisición, instalación y mantenimiento de estos equipos limita severamente su número, especialmente en países en desarrollo. En consecuencia, de la escasa cobertura espacial se crean brechas significativas en el análisis de la contaminación del aire, asignando valores promedio a zonas donde la exposición real de la población puede variar drásticamente debido a fuentes de emisión locales como el tráfico o la industria. Para abordar estas limitaciones, las tecnologías emergentes basadas en sensores de bajo costo (SBC) han permitido generar dispositivos que han ganado una relevancia significativa en los últimos 10 años. De acuerdo con la agencia de protección ambiental (EPA) de los Estados Unidos, los SBC se caracterizan por costar menos de 1000 dólares y estar equipados con una electrónica miniaturizada [5]. La implementación de los SBC permite aumentar de manera considerable el número de sitios de monitoreo que conforman las redes de monitoreo, pasando de un enfoque macro a micro a una fracción del costo [6]. En este contexto, se presenta un proyecto enfocado en el diseño, desarrollo e implementación de un prototipo funcional de una red de monitoreo de bajo costo para la medición de material particulado, temperatura y humedad en puntos estratégicos de la ciudad de Puebla. Este artículo se estructura de la siguiente manera: la sección 2 presenta el estado del arte en redes de monitoreo similares; la sección 3 detalla la arquitectura del sistema, desde el nodo sensor hasta la plataforma en la nube; la sección 4 describe la metodología y los resultados preliminares obtenidos; y finalmente, la sección 5 expone las conclusiones y el trabajo a futuro.

2 Marco Teórico y Estado del Arte

2.1 Redes de monitoreo de calidad del aire con sensores de bajo costo

Los SBC para medir material particulado tienen el potencial demostrado de complementar las mediciones que se efectúan con las redes de monitoreo de calidad del aire de nivel regulatorio (tradicionales). Sin embargo, la precisión de las mediciones depende de las condiciones ambientales y del tamaño de las partículas que se midan. De acuerdo con mediciones de campo se estima un error absoluto medio de entre 4 y 7 $\mu g\ m^{-3}$ [7, 8], pero este puede cambiar dependiendo de las condiciones de humedad y la temperatura. Si bien algunos sensores funcionan bien a concentraciones más altas (>40 $\mu g\ m^{-3}$), su confiabilidad a concentraciones más bajas puede ser limitada [9]. En este contexto se están realizando mejoras en los dispositivos, pero también se está trabajando con algoritmos de calibración para reducir las limitaciones y mejorar la precisión. A pesar de estas limitaciones, los SBC ofrecen la posibilidad de aumentar la resolución espacial del monitoreo de la calidad del aire, sobre todo en países emergentes donde no se cuenta con muchos sitios de monitoreo. Por esta razón, el uso de dispositivos basados en SBC para complementar las redes de monitoreo tradicionales es una tendencia global y diversas iniciativas han demostrado la

viabilidad y el valor de estos sistemas para la ciencia ciudadana, la investigación y la gestión urbana. Proyectos como Purple Air, AIRO, Breathe London y CanAirIO [10, 11, 12, 13] son ejemplos de cómo las redes basadas en SBC pueden generar datos de alta resolución para entender mejor la dinámica de la contaminación del aire en entornos urbanos [8]. Estos proyectos combinan hardware accesible con plataformas de datos para ofrecer información valiosa a la comunidad. Además de realizar mediciones estáticas, los SBC tienen una portabilidad adecuada que les permite realizar mediciones móviles, donde a partir de un teléfono inteligente que provee wifi los SBC envían datos a un repositorio [14]. Así mismo existen diversos casos donde se emplean SBC con y IoT como alternativa al monitoreo convencional. En particular, investigaciones recientes demuestran que los SBC formando una red comunicada a través de IoT son capaces de reproducir bien los patrones temporales de las concentraciones de $PM_{2.5}$ [15]. Así mismo, después de la calibración, los SBC son capaces de lograr una buena correlación con los instrumentos de referencia [16]. En México, trabajos recientes, en las ciudades de Guanajuato y Querétaro, muestran la implementación de SBC y la IoT para determinar las concentraciones de partículas atmosféricas [17, 18].

3 Arquitectura del Sistema

La arquitectura se divide en tres capas principales: el nodo sensor físico, la infraestructura de comunicación y nube y la plataforma de visualización. La figura 1 muestra de manera esquemática las capas de la arquitectura.

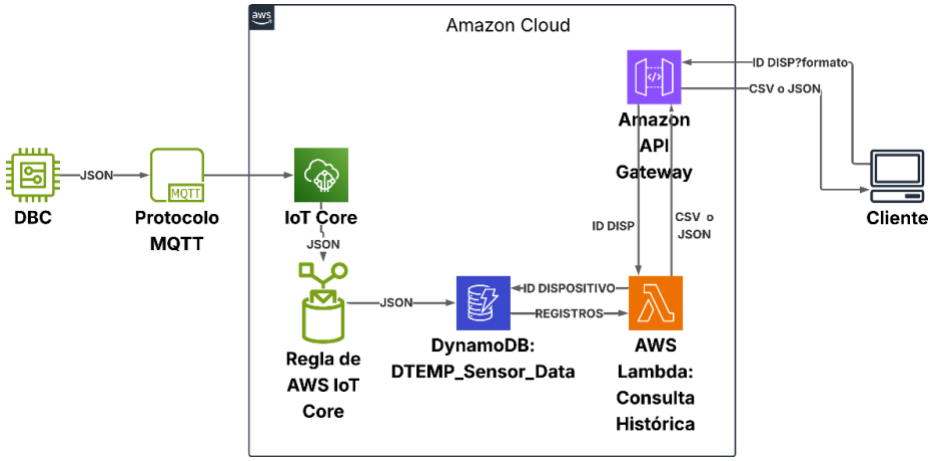


Figura 1. Arquitectura de nube propuesta implementando la arquitectura de software orientada a servicios.

3.1 Nodo sensor (DBC)

La parte central de la red son los nodos, donde se ubica cada uno de los dispositivos basados en SBC, diseñados para ser autónomos y de bajo costo. Cada nodo consta de:

- Un *microcontrolador* que en este caso corresponde a una placa de desarrollo ESP32, las conexiones con el sensor (SEN55) se realizan de acuerdo con las especificaciones del fabricante y las librerías [18]

- *Un sensor de bajo costo* (SBC) que en este caso es el módulo Sensirion SEN55 que es capaz de medir las concentraciones de material particulado ($PM_{1.0}$, $PM_{2.5}$, PM_4 y PM_{10}), temperatura y humedad relativa y
- *Firmware:* El software del dispositivo se desarrolla sobre el marco de trabajo de Arduino para facilitar la programación y el manejo de los componentes. Se emplean librerías del sensor SEN55 [18]

3.2 Arquitectura de nube en AWS

Los datos recolectados por los dispositivos son transmitidos a la nube de Amazon Web Services (AWS) de acuerdo con el siguiente flujo:

1. *Ingesta y Comunicación*. Los dispositivos se conectan de forma segura a AWS IoT Core, que actúa como un broker MQTT. Cada dispositivo tiene un certificado único para su autenticación.
2. *Procesamiento y Almacenamiento*. Una regla de IoT core envía los datos a una base de datos de DynamoDB.
3. *Acceso a Datos***.** Se utilizan AWS API Gateway y AWS Lambda para crear un conjunto de APIs RESTful que permiten a aplicaciones externas consultar los datos, ya sea en tiempo real o en rangos históricos.

3.3 Plataforma de visualización

Se está desarrollando una aplicación web prototipo que consume los datos a través de las APIs. Esta plataforma permitirá la visualización georreferenciada de los sensores en un mapa, así como la generación de gráficos de series temporales para analizar las tendencias de los contaminantes medidos. Mientras tanto, se ha desarrollado una demo funcional alojada en un AWS S3 Bucket.

4 Metodología y Resultados Preliminares

4.1 Metodología

El proyecto sigue una metodología de desarrollo iterativo y se encuentra en una etapa preliminar de prototipado. El prototipo debe ser capaz de enviar datos vía wifi a la nube; para ello, en la base de la pirámide de desarrollo se realizaron conexiones, la configuración del sensor de partículas, la programación de los dispositivos y de la plataforma en la nube. Este enfoque permite la validación temprana de cada componente del sistema (hardware, comunicación, software) y la incorporación de mejoras continuas a partir de los aprendizajes obtenidos en cada fase, reduciendo riesgos y asegurando la robustez del sistema final. En cuanto a las mejoras continuas, comenzamos con placas como Arduino UNO y módulos externos wifi. Sin embargo, el prototipo emplea una placa de desarrollo ESP32, que reduce el tamaño de los dispositivos y tiene el módulo wifi integrado. Así mismo, la configuración de los dispositivos para realizar la conexión con la red wifi se realizaba inicialmente mediante un cable de comunicación entre una laptop y el dispositivo. Sin embargo, en este proceso iterativo, se logró generar una interfaz que puede usarse desde el teléfono y permite enlazar vía Bluetooth a los dispositivos con la red de internet. Estas mejoras tienen el objetivo de aumentar la certeza de contar con datos, pero al mismo tiempo hacen más práctica la instalación de los dispositivos en un ambiente de mediciones real. Ya que, en el futuro cercano, no reportado aquí, se continuará con la instalación de los dispositivos en diferentes puntos de la ciudad de Puebla, en particular en las azoteas de las casas que cuenten con internet.

4.2 Resultados preliminares

Actualmente, se ha completado la fase de prototipado inicial. Se ha construido y validado un prototipo funcional de los dispositivos y se ha observado la operación correcta del SBC (SEN55) y la comunicación con la plataforma de AWS. La cadena de datos completa, desde la captura en el sensor hasta su almacenamiento y visualización básica, ha sido probada exitosamente en un entorno controlado. Se han iniciado las pruebas en un ambiente controlado para ver el rendimiento y funcionamiento de múltiples sensores conectados. El seguimiento de la cadena de datos y otras pruebas se ha observado mediante el cliente de pruebas de mensajería MQTT de AWS, ver Figura 2. Así mismo, se pueden visualizar las concentraciones de PM en tiempo real a través de gráficos de series de tiempo, como se muestra en la Figura 3.

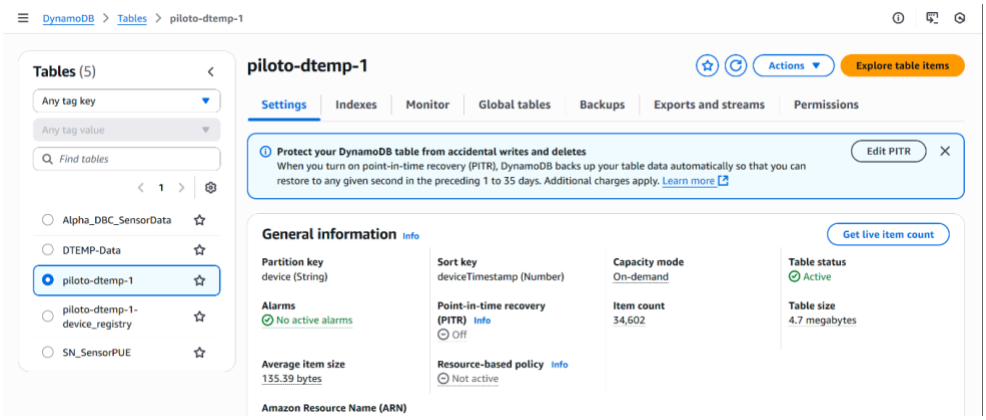


Figura 2. Captura de pantalla del cliente de pruebas de mensajería MQTT de AWS.

5 Conclusiones y Prospectivas

Se ha diseñado e implementado con éxito la arquitectura de un sistema de monitoreo de calidad del aire de bajo costo, que ahora es capaz de desplegar en tiempo real las concentraciones de material particulado y algunas variables meteorológicas, demostrando la viabilidad de integrar dispositivos IoT con una plataforma de nube robusta para la recolección y análisis de datos ambientales. El prototipo funcional valida la cadena de datos de extremo a extremo y sienta las bases para un despliegue a mayor escala. En este sentido y como parte del proyecto IPNVIEP2024-001 sirve como base para desplegar la red de monitoreo de calidad del aire con los primeros 20 dispositivos en el corredor Puebla-Tlaxcala, que permitirá al grupo de investigación del Laboratorio Universitario de Calidad del Aire (LUCA) de la Facultad de Ciencias Químicas desarrollar análisis sobre patrones espaciales y temporales de la contaminación atmosférica y al mismo tiempo proveer los datos a través de un portal de ciencia abierta para fomentar la investigación y la colaboración, incluyendo el estudio del impacto del PM en pacientes con asma en colaboración con el Hospital Universitario de Puebla. El trabajo sobre los dispositivos es iterativo, y como tal plantea continuar con posibles mejoras, en este sentido se está trabajando para que los dispositivos envíen una alerta al administrador sobre fallas en las mediciones, se aumente un módulo de suministro eléctrico vía solar que garantice autonomía en sitios de interés donde no se tiene acceso a la energía eléctrica. Por último, la calibración de los dispositivos, basados en sensores de bajo costo, se realizará en un futuro cercano empleando datos de un equipo de referencia de nivel regulatorio.

Figura 3. Despliegue de información en tiempo real.

Agradecimientos

Agradecemos al Programa de Fomento a la Investigación (VIEP) de la Benemérita Universidad Autónoma de Puebla y al Instituto Politécnico Nacional (IPN) por el financiamiento otorgado a través del proyecto VIEP-IPN 2024 (001).

Referencias

[1] Dominski, F. H., Lorenzetti Branco, J. H., Buonanno, G., Stabile, L., Gameiro da Silva, M., & Andrade, A. (2021). Effects of air pollution on health: A mapping review of systematic reviews and meta-analyses. Environmental Research, 201. https://doi.org/10.1016/j.envres.2021.111487

[2] Ohlwein, S., Kappeler, R., Kutlar Joss, M., Künzli, N., & Hoffmann, B. (2019). Health effects of ultrafine particles: a systematic literature review update of epidemiological evidence. *International Journal of Public Health*, *64*(4), 547–559. https://doi.org/10.1007/S00038-019-01202-7,

[3] Feng, S., Gao, D., Liao, F., Zhou, F., & Wang, X. (2016). The health effects of ambient PM2.5 and potential mechanisms. In Ecotoxicology and Environmental Safety (Vol. 128, pp. 67–74). https://doi.org/10.1016/j.ecoenv.2016.01.030

[4] Watson, J. G., Chow, J. C., Dubois, D., Green, M., Frank, N., & Pitchford, M. (1997). *Guidance for Network Design and Optimum Site Exposure for PM and PM2.5*.

[5] Hall, E. S. (2014). Integrating Sensor Monitoring Technology into the Current Air Pollution Regulatory Support Paradigm: Practical Considerations. *American Journal of Environmental Engineering*, *6*(4), 147–154. https://doi.org/10.5923/J.AJEE.20140406.02

[6] Morawska, L., Thai, P. K., Liu, X., Asumadu-Sakyi, A., Ayoko, G., Bartonova, A., Bedini, A., Chai, F., Christensen, B., Dunbabin, M., Gao, J., Hagler, G. S. W., Jayaratne, R., Kumar, P., Lau, A. K. H., Louie, P. K. K., Mazaheri, M., Ning, Z., Motta, N., … Williams, R. (2018). Applications of low-cost sensing technologies for air quality monitoring and exposure assessment: How far have they gone? Environment International, 116, 286. https://doi.org/10.1016/J.ENVINT.2018.04.018

[7] Feenstra, B., Papapostolou, V., Hasheminassab, S., Zhang, H., Boghossian, B. der, Cocker, D., & Polidori, A. (2019). Performance evaluation of twelve low-cost PM2.5 sensors at an ambient air

monitoring site. *Atmospheric Environment*, *216*, 116946. https://doi.org/10.1016/J.ATMOSENV.2019.116946

[8] Badura, M., Batog, P., Drzeniecka-Osiadacz, A., & Modzel, P. (2018). Evaluation of Low-Cost Sensors for Ambient PM2.5 Monitoring. *Journal of Sensors*, *2018*(1), 5096540. https://doi.org/10.1155/2018/5096540

[9] Johnson, K. K., Bergin, M. H., Russell, A. G., & Hagler, G. S. W. (2018). Field Test of Several Low-Cost Particulate Matter Sensors in High and Low Concentration Urban Environments. *Aerosol and Air Quality Research*, *18*(3), 565–578. https://doi.org/10.4209/AAQR.2017.10.0418

[10] Zimmerman, N. (2022). Tutorial: Guidelines for implementing low-cost sensor networks for aerosol monitoring. *Journal of Aerosol Science*, *159*, 105872. https://doi.org/10.1016/J.JAEROSCI.2021.105872

[11] Kumar, T., & Doss, A. (2022). AIRO: Development of an Intelligent IoT-based Air Quality Monitoring Solution for Urban Areas. Procedia Computer Science, 218, 262–273. https://doi.org/10.1016/J.PROCS.2023.01.008

[12] Environmental Research Group (ERG) at Imperial College London. (2025). *Breathe London Communities*. https://www.breathelondon-communities.org/

[13] CanAirIO. (2025). https://canair.io/es/

[14] Munir, M. M., Adrian, M., Saputra, C., & Lestari, P. (2022). Utilizing Low-cost Mobile Monitoring to Estimate the PM2.5 Inhaled Dose in Urban Environment. *Aerosol and Air Quality Research*, *22*(6), 220079. https://doi.org/10.4209/AAQR.220079

[15] Cowell, N., Baldo, C., Chapman, L., Bloss, W., & Zhong, J. (2024). What can we learn from nested IoT low-cost sensor networks for air quality? A case study of PM2.5 in Birmingham, UK. *Meteorological Applications*, *31*(4), e2220. https://doi.org/10.1002/MET.2220

[16] Cowell, N., Baldo, C., Chapman, L., Bloss, W., & Zhong, J. (2024). What can we learn from nested IoT low-cost sensor networks for air quality? A case study of PM2.5 in Birmingham, UK. *Meteorological Applications*, *31*(4), e2220. https://doi.org/10.1002/MET.2220

[17] Vazquez-Vera, J. L., Espinosa-Calderon, A., Diaz-Carmona, J., Lopez-Farias, R., & Hernandez-Paniagua, I. Y. (2020). Design of a Low-cost Air Quality Remote Monitoring System based on IOT and Sensor Sensitivity Validation. *2020 IEEE International Autumn Meeting on Power, Electronics and Computing, ROPEC 2020*. https://doi.org/10.1109/ROPEC50909.2020.9258685

[18] Rodríguez-Trejo, A., Böhnel, H. N., Ibarra-Ortega, H. E., Salcedo, D., González-Guzmán, R., Castañeda-Miranda, A. G., Sánchez-Ramos, L. E., Chaparro, M. A. E., & Chaparro, M. A. E. (2024). Air Quality Monitoring with Low-Cost Sensors: A Record of the Increase of PM2.5 during Christmas and New Year's Eve Celebrations in the City of Queretaro, Mexico. *Atmosphere 2024, Vol. 15, Page 879*, *15*(8), 879. https://doi.org/10.3390/ATMOS15080879

[19] SEN55, Retrieved from: https://github.com/Sensirion/arduino-i2c-sen5x

Percepción Espacial en Tiempo Real para la Navegación Autónoma de un Robot Diferencial bajo una Ley iLQR

Real-time Spatial Perception for Autonomous Navigation of a Differential Robot under an iLQR Law

Ignacio Gómez Juárez[1], Emilio Zdenko Abarca Cruz[2], Abel Alejandro Rubín Alvarado[3,] Juan Humberto Sossa Azuela[3]

[1]Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Azcapotzalco, Av. De las Granjas 682, Santa Catarina, Azcapotzalco, 02550 Ciudad de México, CDMX.

[2]Facultad de Ingeniería de la Universidad Autónoma del Estado de México, Cerro de Coatepec S/N, Ciudad Universitaria, Universitaria, 50110 Toluca de Lerdo, Méx.

[3]Centro de Investigación en Computación, Av. Juan de Dios Béatiz, esq. Miguel Othón de Mendizábal, Col. Nueva Industrial Vallejo, Alcaldía Gustavo A. Madero, C.P. 07738, CDMX.

Igomezj2000@alumno.ipn.mx, eabarcac004@alumno.uaemex.mx, arubina2200@alumno.ipn.mx, hsossa@ipn.mx

Abstract

This work focuses on implementing autonomous navigation for a differential mobile robot, integrating a LiDAR sensor, odometry data, and an optimal controller based on the Iterative Linear Quadratic Regulator (iLQR) algorithm. The NVIDIA Jetson Nano serves as the embedded computing platform, operating under the Robot Operating System (ROS) architecture. The main objective is to experimentally validate the robot's ability to follow defined trajectories, perform environmental perception through point detection and visualization, and execute real-time collision avoidance maneuvers. Emphasis is placed on the synergy between iLQR's iterative control, the Jetson Nano's processing capabilities for complex tasks such as LiDAR data acquisition and odometry estimation, and the modularity offered by ROS for integrating these essential components into a robust and precise autonomous system.

Resumen

El presente trabajo aborda la implementación de la navegación autónoma en un robot móvil diferencial, integrando un sensor LiDAR, datos de odometría y un controlador óptimo basado en el algoritmo Iterative Linear Quadratic Regulator (iLQR). La plataforma de cómputo embebido utilizada es la NVIDIA Jetson Nano, operando bajo la arquitectura de Robot Operating System (ROS). El objetivo principal es validar experimentalmente la capacidad del robot para seguir trayectorias definidas, realizar la percepción del entorno mediante la detección y visualización de puntos, y ejecutar maniobras de evitación de colisiones en tiempo real. Se destaca la sinergia entre el control iterativo de iLQR, la capacidad de procesamiento de la Jetson Nano para tareas complejas como la

lectura de LiDAR y la estimación odométrica, y la modularidad que ofrece ROS para la integración de estos componentes esenciales en un sistema autónomo robusto y preciso.

Keywords and phrases: Autonomous Navigation, Differential Mobile Robot, Optimal Control, iLQR, LiDAR, ROS, Odometry, Jetson Nano, SLAM, Embedded Systems.

1 Introducción

La navegación autónoma en robots diferenciales exige una estrecha coordinación entre la percepción del entorno, el diseño de un control óptimo y el procesamiento en plataformas de cómputo embebido. El algoritmo Iterative Linear Quadratic Regulator (iLQR) se ha consolidado como una solución eficaz para el seguimiento de trayectorias en sistemas no lineales con restricciones de movilidad [1, 2]. Estudios comparativos han mostrado que iLQR supera al LQR clásico al refinar iterativamente la política de control sobre la dinámica completa del robot, mejorando tanto la precisión de la trayectoria como la eficiencia en el uso de la energía.

La incorporación de sensores LiDAR y encoders para obtener datos de odometría, junto con el uso de ROS para el procesamiento y mapeo, permite generar mapas fiables y mantener una navegación precisa en espacios interiores [3] Por otro lado, la Jetson Nano ha demostrado ofrecer un balance óptimo entre potencia computacional y eficiencia energética, posibilitando la ejecución en tiempo real de algoritmos de SLAM, fusión sensorial y generación de comandos PWM para el accionamiento de motores [4].

En el presente trabajo se explora la integración de iLQR, percepción basada en LiDAR y odometría, y la plataforma Jetson Nano, con el fin de validar experimentalmente la capacidad del robot para seguir trayectorias definidas, visualizar los puntos detectados en su entorno y evitar colisiones de manera autónoma y en tiempo real.

2 Marco Teórico y Estado del Arte

La navegación autónoma de robots móviles ha sido ampliamente abordada en la robótica moderna, destacando los robots diferenciales por su simplicidad mecánica y flexibilidad en entornos estructurados y no estructurados. Estos sistemas suelen modelarse mediante ecuaciones cinemáticas y dinámicas que capturan sus restricciones no holonómicas y comportamientos inerciales. Para el seguimiento de trayectorias en sistemas no lineales como los robots diferenciales, se ha comprobado que el algoritmo Iterative Linear Quadratic Regulator (iLQR) permite minimizar el error de seguimiento y el esfuerzo de control mediante una política iterativa sobre un modelo linealizado de la dinámica [1, 2].

iLQR es una extensión del LQR clásico que permite aplicar control óptimo en sistemas no lineales a través de un proceso iterativo de retropropagación de ganancias, usando una dinámica linealizada a lo largo de una trayectoria nominal. Este método ha sido implementado exitosamente en diversos sistemas móviles, mostrando mejoras en precisión y eficiencia energética en comparación con otros enfoques lineales [1].

La percepción del entorno es un componente esencial para garantizar que un robot navegue con seguridad, evitando colisiones y reaccionando a obstáculos dinámicos. Entre los sensores más utilizados para esta tarea destacan los LiDAR, que permiten generar nubes de puntos del entorno en tiempo real, y los encoders, que miden el desplazamiento de las ruedas para estimar la posición del robot mediante odometría [3]. La odometría, aunque susceptible a errores acumulativos, es una fuente confiable de datos cuando se fusiona adecuadamente con sensores externos mediante algoritmos como el SLAM (Simultaneous Localization and Mapping) en el marco de ROS.

ROS (Robot Operating System) proporciona una arquitectura modular para el desarrollo de sistemas robóticos, facilitando la integración entre sensores, algoritmos de control y mapeo. Diversos proyectos han demostrado la viabilidad de utilizar ROS junto con Jetson Nano para tareas de navegación autónoma, ya que esta unidad de cómputo embebida es capaz de ejecutar en tiempo real procesos como la lectura de LiDAR, odometría, mapeo y control [4].

La Jetson Nano se presenta como una solución efectiva debido a su bajo consumo, capacidad de procesamiento con GPU integrada y compatibilidad con bibliotecas como CUDA y OpenCV. Esto la convierte en una plataforma idónea para desplegar controladores como iLQR en robots de tamaño reducido, especialmente cuando se requiere ejecutar algoritmos de visión o mapeo en paralelo al control.

En conjunto, la integración de control óptimo mediante iLQR, percepción mediante LiDAR y odometría, y procesamiento embebido en Jetson Nano bajo la arquitectura ROS, permite desarrollar robots móviles capaces de navegar de forma autónoma, seguir trayectorias y evitar colisiones con precisión y robustez, lo cual es crucial en aplicaciones reales como logística, exploración y asistencia autónoma.

3 Arquitectura de la Aplicación

La arquitectura del sistema se basa en una Jetson Nano como unidad central de procesamiento, encargada de ejecutar el controlador iLQR, gestionar la adquisición de datos de sensores y generar las señales de control para los actuadores. La elección de la Jetson Nano responde a su capacidad de cómputo paralelo mediante GPU integrada, lo que permite implementar algoritmos de control y visión artificial en tiempo real con un consumo energético reducido, cualidades que la hacen adecuada para aplicaciones de robótica móvil. La figura 1 muestra un diagrama general de la arquitectura propuesta.

El algoritmo iLQR se ejecuta en tiempo real sobre la Jetson Nano, recibiendo como entrada el estado actual del robot, estimado a través de odometría basada en encoders, y generando como salida los torques óptimos para los motores. Estos torques son transformados en señales PWM a través de un módulo de mapeo calibrado, y enviados al driver L298N, el cual alimenta los motores DC del robot diferencial.

En paralelo, un sensor LiDAR montado sobre el robot adquiere datos del entorno y transmite un conjunto de distancias angulares que permiten construir una nube de puntos en tiempo real. Esta nube de puntos puede visualizarse o procesarse para futuras tareas de navegación autónoma o SLAM.

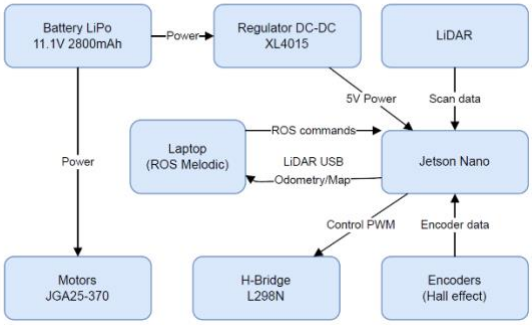


Figura 1. Diagrama de bloques del funcionamiento del robot diferencial.

Nota. En el diagrama de funcionamiento del robot diferencial se pueden apreciar como interactúa el hardware del robot. Creación propia.

El sistema de tracción del robot se basa en dos motores de corriente continua con reducción JGA25-370, seleccionados por su balance entre alta velocidad sin carga y torque adecuado para aplicaciones de movilidad. Estos motores operan a 12 V y cuentan con un reductor de relación 4.4:1, ofreciendo una velocidad nominal de 1370 rpm y un torque de 13.73 mNm bajo carga. Además, incluyen codificadores incrementales de efecto Hall de doble fase, los cuales permiten realizar estimaciones precisas de desplazamiento mediante conteo de pulsos.

Los pulsos generados por los encoders son procesados por la unidad de control (Jetson Nano), la cual segmenta las rotaciones de las ruedas y establece una relación entre pulsos por revolución y distancia recorrida. Esta información es fundamental para la estimación odométrica de la posición del robot y para la implementación del controlador óptimo. La tabla 1 resume las principales características técnicas de los motores utilizados.

Tabla 1. Especificaciones del motor JGA25-370

Parámetro	**Valor**
Tensión nominal	12 V
Relación de reducción	4.4:1
Velocidad sin carga	1931 rpm
Corriente sin carga	100 mA
Velocidad con carga nominal	1370 rpm
Corriente con carga nominal	330 mA
Torque con carga nominal	13.73 mNm
Torque en bloqueo	49.02 mNm
Corriente en arranque (máx.)	1.5 A
Capacidad de carga máxima	20 kg

Nota. Adaptado de "JGA25-370 DC Geared Motor Datasheet. UNIT electronics, 2025.

Para la entrega de potencia a los motores y la inversión de su sentido de giro se utiliza un módulo puente H L298N, capaz de manejar hasta dos motores DC de forma independiente. Este módulo

soporta una corriente continua de hasta 2 A y picos de hasta 3 A, lo cual lo hace adecuado para manejar los picos de arranque observados en la operación del sistema sin comprometer los niveles de corriente de la Jetson Nano. La Jetson Nano a su vez esta alimentada por una batería Lipo de 11.1 v conectada a un regulador de voltaje XL4015 que protege a la tarjeta Jetson Nano de sobrecargas manteniendo el voltaje a un máximo de 5v. La Jetson Nano, es seleccionado por su capacidad de procesamiento paralelo mediante GPU y su compatibilidad con librerías de inteligencia artificial, visión artificial y control en tiempo real. Esta unidad es responsable de ejecutar el controlador iLQR, procesar señales de los sensores (encoders, LiDAR, etc.) y generar señales de control en tiempo real. Las especificaciones principales del kit Jetson Nano se listan en la tabla 2.

Tabla 2. Especificaciones del Kit de desarrollo Jetson Nano

Característica	**Valor**
Voltaje de operación	5 VDC (MicroUSB o Jack 5.5x2.5 mm)
Corriente requerida	2 A (mínimo) / 4 A (máximo)
GPU	NVIDIA Maxwell™, 128 núcleos
Frecuencia máxima de GPU	921 MHz
CPU	ARM Cortex-A57, cuatro núcleos, 1.43 GHz
USB	1x USB 3.0, 3x USB 2.0
Entradas/Salidas (E/S)	3x UART, 2x SPI, 2x I2S, 4x I2C, 14x GPIO
Consumo de energía	5 W – 10 W

Nota. Adaptado de "Nvidia jetson nano developer kit”. Nvidia, 2020.

La elección de esta unidad se basa en su bajo consumo, facilidad de programación, compatibilidad con Python y ROS, y su capacidad para ejecutar múltiples procesos concurrentes, lo cual resulta esencial en sistemas de robótica móvil que integran control, percepción y procesamiento de sensores en tiempo real.

El sensor empleado en este proyecto es el ydlidar x4, un escáner láser bidimensional de tipo rotativo, seleccionado por su bajo costo, facilidad de integración con plataformas ROS y rendimiento adecuado para tareas básicas de percepción espacial. Este sensor opera mediante el principio de telemetría láser (LiDAR: *Light Detection and Ranging*), utilizando luz láser infrarroja para medir distancias con alta frecuencia y precisión angular. El ydlidar x4 realiza barridos horizontales en un campo de visión de 360 grados, generando aproximadamente 1000 puntos por escaneo con una frecuencia ajustable de entre 5 y 12 Hz. Cada medición representa la distancia a un obstáculo en una dirección específica, lo que permite reconstruir un mapa plano del entorno cercano del robot. Una de sus ventajas es su tamaño compacto y bajo consumo energético, lo que lo hace ideal para plataformas móviles de tamaño reducido como la Jetson Nano. La Tabla 3 resume las principales características técnicas del sensor ydlidar x4.

Para montar todos los componentes en una sola estructura, se desarrolló un diseño en SolidWorks, considerando las fuerzas, cargas y esfuerzos generados por los elementos del sistema. Se llevó a cabo un análisis estático y topológico de la estructura con el objetivo de obtener un chasis resistente y, al mismo tiempo, lo más ligero posible.

Tabla 3. Especificaciones del sensor ydlidar x4.

Parámetro	Valor
Rango de medición	0.12 m – 10 m
Precisión	±1.5 cm (típico)
Frecuencia de escaneo	5 – 12 Hz
Resolución angular	~0.48° (a 5 Hz)
Campo de visión	360°
Longitud de onda del láser	785 nm (infrarrojo)
Tipo de láser	Clase I (seguro para la vista)
Interfaz de comunicación	USB 2.0 (UART a través de CP2102)
Velocidad de comunicación	128000 baudios
Tensión de operación	5 V DC
Consumo eléctrico	~0.3 A
Dimensiones	72.0 mm × 72.0 mm × 42.0 mm
Peso	170 g

Nota. Adaptado de "Ydlidar x4 pro". Ydlidar, 2025.

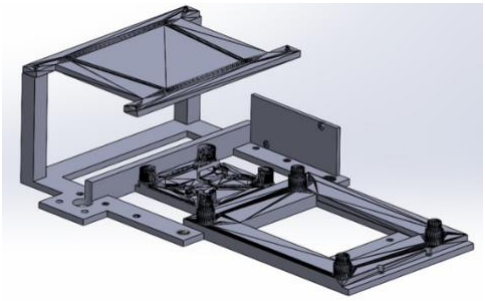


Figura 2. Diseño del chasis del robot diferencial.

Nota. Chasis del robot diferencial modelado en SolidWorks para su posterior impresión 3D en PLA. Creación propia.

Además, se optó por un diseño modular, que permite desmontar y volver a montar cada componente con facilidad, utilizando tornillos. Esta característica facilita futuras modificaciones o mantenimientos del sistema, brindando flexibilidad en el desarrollo del prototipo. Una vez finalizado el diseño, la estructura fue impresa en PLA mediante tecnología de impresión 3D. Posteriormente, se procedió al ensamblaje del hardware correspondiente al robot móvil diferencial.

En el siguiente diagrama de flujo (figura 4) se puede apreciar la representación del flujo de datos en el algoritmo de control iLQR implementado en el robot diferencial.

A continuación, se presenta el diagrama de flujo del algoritmo de reconocimiento del entorno implementado a través de ROS melodic, el LiDAR y la Jetson Nano. (Vea figura 5)

Figura 3. Robot diferencial armado.

Nota. Robot diferencial con diseño modular completamente ensamblado. Creación propia.

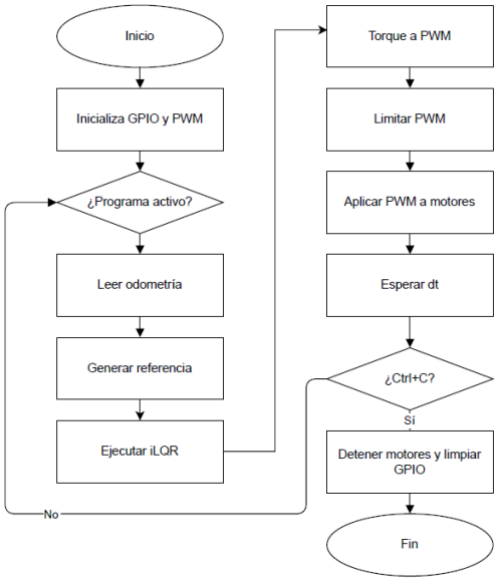


Figura 4. Diagrama de flujo algoritmo iLQR.

Nota. Creación propia.

4 Desarrollo

a. Desarrollo del controlador.

El robot utilizado es una plataforma diferencial de dos ruedas impulsadas de manera independiente, y un punto de apoyo pasivo. El modelo cinemático de este sistema asume que el movimiento ocurre en un plano bidimensional y que no hay deslizamiento lateral. Las ecuaciones que describen la evolución del estado cinemático (x, y, θ) en el tiempo son las ecuaciones uno, dos y tres que se muestran a continuación.

$$\frac{dx}{dt} = v * cos(\theta) \quad (1)$$

$$\frac{\mathrm{dy}}{\mathrm{dt}} = \mathrm{v} * \sin(\theta) \quad (2)$$

$$\frac{d\theta}{dt} = \omega \quad (3)$$

donde v es la velocidad lineal y ω es la velocidad angular del robot [5].

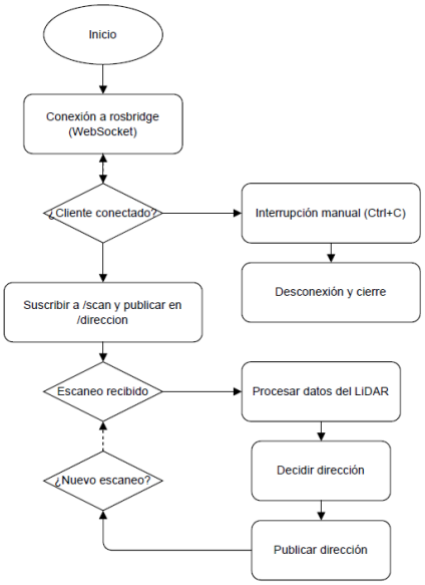


Figura 5. Diagrama de flujo algoritmo de detección del LiDAR.

Nota. Creación propia.

Para generar un comportamiento más realista en el control, se desarrolló un modelo dinámico que considera las fuerzas aplicadas, la fricción y las características mecánicas de la plataforma. El modelo dinámico se describe por:

$$\frac{dv}{dt} = \left(\frac{r}{m}\right) * (\tau_d + \tau_i - f_v) \quad (4)$$

$$\frac{d\omega}{dt} = \left(\frac{r}{J*L}\right) * (\tau_d - \tau_i - f_\omega) \quad (5)$$

Donde r es el radio de la rueda, m es la masa del robot, J es el momento de inercia, L es la distancia entre ruedas, f_v representan fricción lineal, f_ω representan fricción y rotacional[5].

Los torques se escalan por la eficiencia mecánica (η) y la relación de engranes (G) [6].

$$\tau_{real} = \eta * G * \tau_{motor} \quad (6)$$

Para estimar el estado del robot en tiempo real se implementó un sistema de odometría diferencial utilizando dos encoders cuadratura de 360 pulsos por revolución, uno en cada rueda. Cada pulso se transforma en una distancia recorrida usando la formula siguiente [7].

$$d = \frac{2\pi r}{N} \quad (7)$$

donde N es el número de pulsos por revolución. A partir de la distancia recorrida por cada rueda (d_d, d_i), se estima el desplazamiento del centro del robot mediante las ecuaciones ocho y nueve [7].

$$\Delta s = \frac{d_d + d_i}{2} \tag{8}$$

$$\Delta\theta = \frac{d_d - d_i}{L} \tag{9}$$

Y la nueva posición global se calcula por las fórmulas diez y once [8].

$$x_{\{k+1\}} = x_k + \Delta s * cos\left(\theta_k + \frac{\Delta\theta}{2}\right) \tag{10}$$

$$y_{\{k+1\}} = y_k + \Delta s * sin\left(\theta_k + \frac{\Delta\theta}{2}\right) \tag{11}$$

$$\theta_{\{k+1\}} = \theta_k + \Delta\theta \tag{12}$$

Este sistema de odometría se actualiza en tiempo real mediante interrupciones gestionadas por GPIO, lo que permite alimentar constantemente el controlador con una estimación confiable del estado.

Para poder enviar comandos de control físico a los motores, fue necesario caracterizar experimentalmente la relación entre el valor PWM aplicado y el torque resultante. A partir de la medición de los torques medidos en el motor respondiendo a diferentes señales pwm se construyó un polinomio cuadrático de interpolación:

$$PWM(\tau) = a\tau^2 + b\tau + c \tag{13}$$

donde a, b y c fueron determinados mediante regresión polinomial. Esto permite convertir de manera eficiente los torques calculados por el controlador a señales PWM para accionar los motores [9].

El controlador implementado fue un Iterative Linear Quadratic Regulator (iLQR), una técnica de control óptimo basada en la minimización de un costo cuadrático, considerando la dinámica del sistema y la evolución temporal. El objetivo es encontrar la secuencia de acciones u_0, u_1, ..., u_{N-1} que minimice la siguiente función de costo.

$$J = \Sigma \left(x_k - x_{ref_k}\right)^T Q \left(x_k - x_{ref_k}\right) + u_k^T R\, u_k + \left(x_N - x_{ref_N}\right)^T Qf \left(x_N - x_{ref_N}\right) \tag{14}$$

La solución se obtiene iterando un pase hacia adelante (simulación de la trayectoria) y un pase hacia atrás (actualización de la política de control). La política de control se representa en la ecuación número quince [10].

$$u_k = K_k\left(x_k - x_{ref_k}\right) + d_k \tag{15}$$

Finalmente, se aplica solo el primer control de la secuencia generada, mientras se recalcula en cada iteración en función del estado estimado por odometría, implementando así un control en lazo cerrado.

La trayectoria usada para validación consistió en una circunferencia parametrizada en espacio y tiempo. En cada instante, se calcula la posición deseada $x_{ref}(t)$ y se alimenta al controlador iLQR. Este genera el torque óptimo, que luego se convierte a PWM y se aplica a los motores. Se añadieron mecanismos para suavizar transiciones entre comandos, establecer límites mínimos de activación y garantizar la seguridad durante la operación

b. Implementación del sensor LiDAR.

La incorporación del sensor YDLIDAR X4 en la plataforma NVIDIA Jetson Nano se realizó mediante Robot Operating System (ROS) Melodic, lo que permitió gestionar la comunicación con el hardware, procesar datos y visualizarlos en tiempo real. La conexión física se estableció vía USB, mientras que la integración en ROS se apoyó en el SDK oficial de YDLidar y un espacio de trabajo Catkin donde se configuró el archivo lidar.launch con los parámetros de operación (puerto, velocidad de comunicación y modelo de sensor).

Una vez en funcionamiento, el sensor publica lecturas en el tópico /scan, con información sobre ángulos, rangos de detección e intensidades reflejadas. Cada escaneo genera alrededor de 1000 puntos a 5 Hz, suficientes para aplicaciones de mapeo en tiempo real y navegación en entornos reducidos.

Para aprovechar estos datos, se implementó un script en Python que filtra lecturas inválidas y analiza la zona frontal del campo de visión, dividiéndola en izquierda, centro y derecha. En caso de detectar obstáculos a distancias críticas, se ejecuta una estrategia de evasión, publicando decisiones en el tópico /direccion, el cual es interpretado por los nodos encargados del control de motores.

Finalmente, se desarrolló una visualización gráfica en coordenadas cartesianas (X, Y) mediante matplotlib, lo que permite monitorear el entorno percibido por el robot en tiempo real sin afectar la ejecución de la lógica de navegación.

c. Resultados y pruebas experimentales.

Antes de realizar la implementación física del controlador iLQR, se llevaron a cabo simulaciones considerando tanto el modelo dinámico como el cinemático del robot móvil diferencial. Estas simulaciones permitieron validar la efectividad del enfoque propuesto y ajustar los parámetros necesarios para el seguimiento de trayectorias.

Para una trayectoria de referencia rectilínea, se obtuvo los torque y posicionamientos que se pueden apreciar en la figura 6.

Nota: En la figura 6 del lado izquierdo se puede apreciar el movimiento del carrito diferencial en la simulación milímetros, y a la derecha, el torque aplicado en cada rueda en Nm-1. Creación propia.

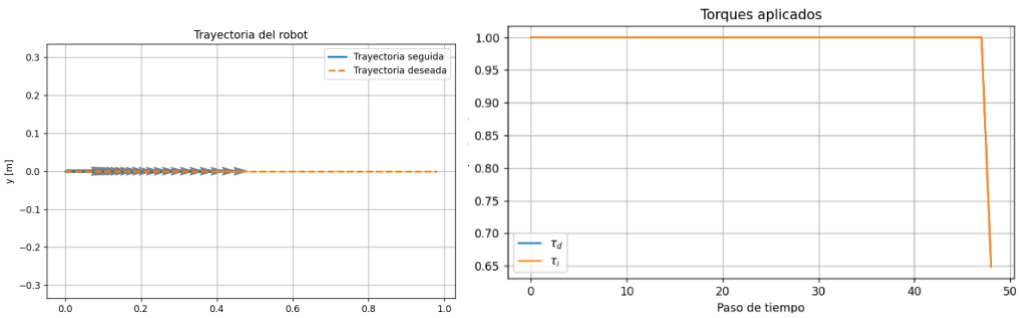


Figura 6. Gráficas de trayectoria recta deseada contra la trayectoria simulada seguida y el control de torques generados por el robot diferencial.

Posteriormente, se evaluó una trayectoria de referencia circular, donde se obtuvieron los torque y posicionamientos que se pueden apreciar en la figura 7.

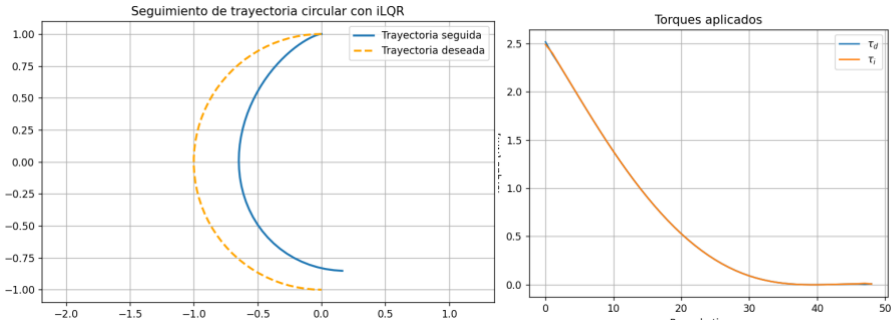


Figura 7. Grafica de trayectoria circular deseada contra la trayectoria simulada seguida y sus torques aplicados.

Nota: En la figura 7 del lado izquierdo se puede apreciar el movimiento del carrito diferencial en la simulación en milímetros, y a la derecha, el torque aplicado en cada rueda en Nm^{-1}. Creación propia.

Es importante destacar que, en el caso de la trayectoria circular, fue necesario definir una referencia que incluyera no solo posiciones, sino también velocidades y orientaciones derivadas mediante gradientes suaves. De no ser así, el algoritmo iLQR tiende a no converger adecuadamente, quedando atrapado en mínimos o máximos locales de la función de costo. A continuación, en la figura 9 se muestra la gráfica de los valores de la función de costo, con una referencia no suficientemente especifica, la cual nunca logra converger en un valor optimo, esto produce trayectorias no deseadas como la que se observa en la siguiente figura 9.

Nota. La grafica de valores en la función del algoritmo iLQR nunca alcanzo un valor optimo sin importar cuantas iteraciones se hicieran. Creación propia.

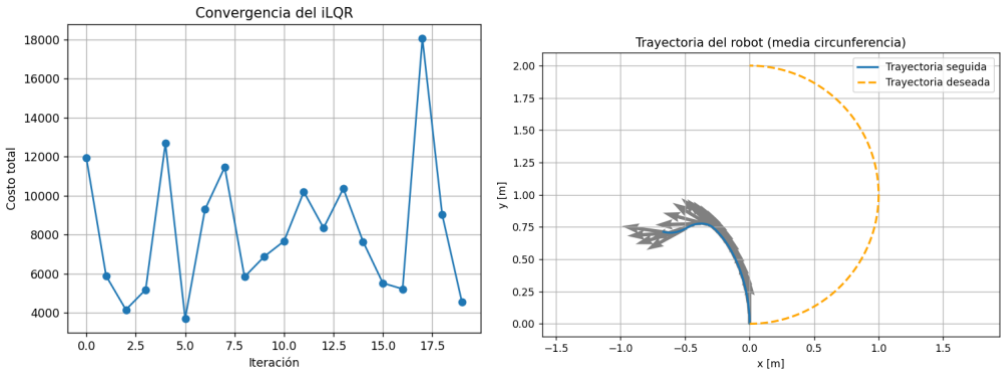


Figura 8. Grafica de convergencia en la función de costo y trayectoria de seguimiento fallida.

En la implementación física, considerando las simulaciones previas y los ajustes correspondientes, el controlador basado en el modelo dinámico y cinemático del robot diferencial fue capaz de seguir exitosamente una trayectoria rectilínea y una trayectoria circular. El seguimiento se realizó mediante el algoritmo de control óptimo iLQR, el cual reguló los torques aplicados a lo largo del trayecto, logrando converger hacia un valor óptimo de la función de costo.

A partir de las mediciones obtenidas mediante enconders instalados en los motores del robot y utilizando odometría diferencial, se reconstruyó el recorrido real del robot, el cual se compara con una trayectoria rectilínea inclinada de referencia, las cuales se presentan en la Figura 9.

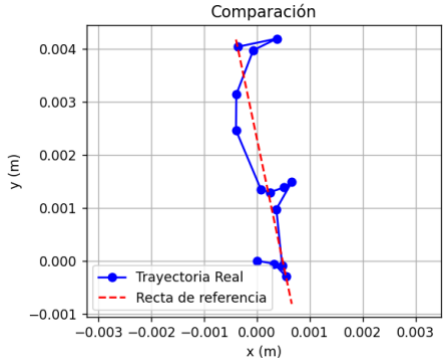


Figura 9. Gráfica superpuesta de la trayectoria recta deseada y seguida

Nota. Creación propia.

Posteriormente, se evaluó una trayectoria de referencia circular, con la misma metodología, la cual se puede apreciar en la figura 10.

Para evaluar el desempeño del sistema de control, se comparó el error de posición en cada uno de los puntos que conforman las trayectorias recta y circular. En total se consideraron 14 puntos de referencia, y cada trayectoria fue repetida 15 veces con el fin de obtener

resultados estadísticamente representativos. A partir de estas repeticiones, se calcularon los valores promedio del error en cada punto.

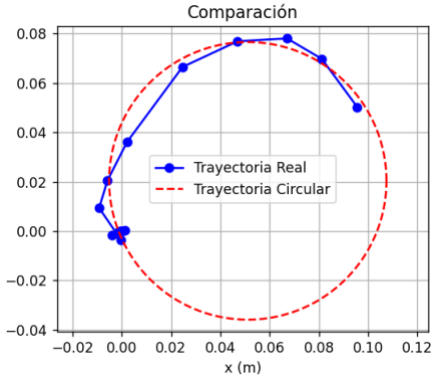


Figura 10. Gráfica superpuesta de la trayectoria circular deseada y seguida

Nota. Creación propia.

En la Tabla 4 se presentan los márgenes de error obtenidos al comparar las trayectorias deseadas con las trayectorias ejecutadas físicamente por el robot.

En la trayectoria recta, se obtuvo MAE = 0.00026 m, RMSE = 0.00034 m y desviación estándar = 0.00023 m, con intervalo de confianza al 95% de [0.00012, 0.00040] m. En la trayectoria circular, MAE = 0.00200 m y RMSE = 0.00277 m, reflejando mayor complejidad en movimientos curvos. El coeficiente de variación fue de 88% y 79% en trayectorias recta y circular, respectivamente, indicando alta consistencia relativa frente a errores absolutos pequeños.

Comparativamente, estos errores son menores que el error lateral reportado por Huang et al. (2025) de 0.05 m en trayectorias rectas y 0.06 m en trayectorias curvas y muestran mejor precisión en curvas que controladores lineales como LQR [12]. Estudios con MPC presentan mayor carga computacional [13], mientras que trabajos con enfoques cinemáticos o PID son principalmente cualitativos o simulados, limitando la comparación cuantitativa.

La integración del iLQR con LiDAR permitió detección de obstáculos en tiempo real y desplazamiento autónomo seguro (Figura 11).

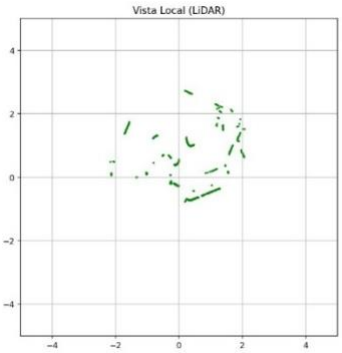


Figura 11. Recreación de puntos representados en el entorno.

Nota. Creación propia.

Tabla 4. Errores de posición promedio en las trayectorias.

Punto	Error Línea (m)	Error Circular (m)
1	0.000474	0.001191
2	0.000171	0.000581
3	0.000000	0.000998
4	0.000027	0.002481
5	0.000084	0.000111
6	0.000476	0.002945
7	0.000314	0.000469
8	0.000034	0.005042
9	0.000122	0.000703
10	0.000354	0.004812
11	0.000208	0.003046
12	0.000277	0.000463
13	0.000764	0.003626
14	0.000000	0.001586

Nota. Creación propia.

Se desarrolló un programa que integra el iLQR con datos de LiDAR, permitiendo al robot móvil desplazarse autónomamente y evitar obstáculos en tiempo real. Se incluye un enlace a un video que muestra esta implementación. Pruebas realizadas

5 Conclusiones y Trabajo a Futuro

Los resultados muestran que el controlador iLQR (Iterative Linear Quadratic Regulator) puede seguir trayectorias de referencia con alta precisión, aunque no de manera exacta, gracias a la optimización de su función de costo regulada por las matrices Q, R y Qf.

Como algoritmo iterativo, el iLQR puede converger a soluciones subóptimas si el número de iteraciones es insuficiente, o acumular errores numéricos si este es excesivo. Por ello, es fundamental ajustar cuidadosamente las iteraciones y los parámetros de costo para equilibrar precisión y eficiencia computacional.

Una ventaja notable del iLQR es su capacidad de integrarse con otros algoritmos de planificación, lo que mejora la autonomía y robustez del robot diferencial. Como trabajo futuro, se propone reemplazar la odometría basada en encoders por estimaciones obtenidas mediante LiDAR, con el fin de reducir errores acumulativos y mejorar la retroalimentación del sistema de control.

En conclusión, el iLQR se presenta como una herramienta prometedora para el control de trayectorias en sistemas móviles, con amplio potencial de mejora mediante sensores avanzados y estrategias de planificación más sofisticadas.

Agradecimientos

Los autores de este artículo agradecen al Consejo Nacional de Ciencia y Tecnología (CONACYT) y al Centro de Investigación en Computación por el apoyo otorgado en la realización de este trabajo.

Referencias

[1] Zhang, Y., Wang, H., & Li, J. (2012). Application of iterative LQR on a mobile robot with simple dynamics. *Journal of Zhejiang University-SCIENCE A*, 807–814. doi:10.1631/jzus.C1100379

[2] Aaqaoui, L., Pereira, J., & Santos, E. (2024). Application of iterative LQR on a mobile robot with simple dynamics. Obtido de arXiv: https://arxiv.org/abs/2404.18312

[3] Zhao, X., Li, P., & Kumar, R. (2023). ROS-based autonomous navigation robot platform (Owlbot). Sensors. doi:10.3390/s25144306

[4] Rahul, A., Singh, K., & Patel, R. (2022). Autonomous vehicle navigation with LiDAR using path planning. doi:arXiv:2212.07155

[5] Figueroa, J. (2023). Cinemática y dinamica de robots moviles con ruedas. Ciladi. doi:ISBN: 978-9942-7078-3-3

[6] Craig, J. J. (2020). Robot modeling and control. Wiley. doi:9781119523996

[7] Dudek, G., & Jenkin, M. (2010). Computational principles of mobile robotics. Cambridge University Press. doi:9780521692120

[8] Siegwart, R. (2011). Introduction to autonomous mobile robots. MIT Press. doi:9780262015356

[9] Bishop, R. C. (2016). Modern control systems. Pearson. doi:9780134407623

[10] Li, W., & Todorov, E. (2004). Iterative linear quadratic regulator design for nonlinear biological movement systems. INSTICC Press, 222-229.

[11] Huang, e. a. (2025). Enhanced path tracking control for differential drive robots under low-adhesion conditions. Nature Robotics, 45-58.

[12] Williams, G., Drews, P., Goldfain, B., Rehg, J., & Theodorou, E. (2016). Aggressive driving with model predictive path integral control. 2016 IEEE International Conference on Robotics and Automation (ICRA), (pp. 1433-1440). doi:10.1109/ICRA.2016.7487278

[13] Kong, J., Pfeiffer, M., Schildbach, G., & Borrelli, F. (2015). Kinematic and dynamic vehicle models for autonomous driving control design. 2015 IEEE Intelligent Vehicles Symposium (IV), (pp. 1094-1099). doi:10.1109/IVS.2015.7225830

Análisis y Visualización del Impacto del Estado Emocional en el Rendimiento Académico mediante Herramientas en la Nube y Algoritmos Predictivos

Analysis and Visualization of the Impact of Emotional State on Academic Performance using Cloud Tools and Predictive Algorithms

Roberto Carlos Silva Lozano, Judith Pérez Marcial, Beatriz Alejandra Flores Rojas, María del Carmen Santiago Díaz, Ana Claudia Zenteno Vázquez, Gustavo Trinidad Rubín Linares.

Benemérita Universidad Autónoma de Puebla, Complejo Regional Centro San José Chiapa. Calle 2 Sur, Ciudad Modelo, San José Chiapa, Puebla

joselyne.florese@alumno.buap.mx , {judith.perez, beatriza.flore, marycarmen.santiago, ana.zenteno, gustavo.rubin}@correo.buap.mx

Abstract

The emotional well-being of university students is a determining factor in their academic performance, particularly in engineering programs with high cognitive load. This paper analyzes the relationship between emotional state and academic performance in university students using structured surveys and data analysis techniques. Emotional, academic, and demographic data were collected from 344 students enrolled in engineering programs from the 2020 to 2024 cohorts. The information was processed using a cloud-based architecture, integrating services for data storage, cleaning, analysis, predictive modeling, and visualization. The results reveal relevant patterns among variables such as stress, anxiety, motivation, and GPA, showing higher performance among students with elevated levels of anxiety. The main contribution of this study is the integration of emotional analytics, machine learning, and cloud computing in a real-world academic setting, providing a scalable solution that can support institutional decision-making focused on student well-being and improved academic performance.

Resumen

El bienestar emocional de los estudiantes universitarios es un factor determinante en su rendimiento académico, particularmente en programas de ingenierías con alta carga cognitiva. En este trabajo se analiza la relación entre el estado emocional y el desempeño académico de estudiantes universitarios mediante el uso de encuestas estructuradas y técnicas de análisis de datos. Se recolectaron datos emocionales, académicos y demográficos de 344 estudiantes pertenecientes a programas de ingeniería correspondientes a las generaciones 2020 a 2024. La información fue procesada mediante una arquitectura basada en cómputo en la nube, integrando servicios para almacenamiento, limpieza, análisis, modelado predictivo y visualización de datos. Los resultados evidencian patrones relevantes entre variables como estrés, ansiedad, motivación y promedio académico, mostrando una mayor en el desempeño de estudiantes con niveles elevados de ansiedad. La principal contribución de este estudio es la integración de analítica emocional, aprendizaje automático y computo en la nube en un entorno académico real, proporcionando una

solución escalable que puede apoyar a la toma de decisiones institucionales orientadas al bienestar estudiantil y la mejora del rendimiento académico.

Keywords and phrases: **:** Estado Emocional, Rendimiento Académico, Cómputo en la Nube. Machine Learning. Analítica Educativa

1 Introducción

El bienestar emocional de los estudiantes universitarios se ha consolidado en los últimos años como un factor determinante para el logro académico, la permanencia escolar y el desarrollo integral. Diversas investigaciones han demostrado que variables como el estrés, la ansiedad, la motivación y el apoyo socioemocional influyen directamente en procesos cognitivos esenciales impactando de manera significativa el rendimiento académico, particularmente en programas de educación superior con alta carga cognitiva y elevados niveles de exigencia.

En los entornos universitarios actuales, los estudiantes enfrentan múltiples desafíos académicos, personales y emocionales que pueden afectar tanto su desempeño como su bienestar general [1][2]. A pesar de la creciente evidencia sobre la relevancia del componente emocional en el proceso de aprendizaje, muchas instituciones educativas aún carecen de mecanismos sistemáticos, escalables y basados en datos que permitan analizar de forma integral la relación entre el estado emocional y el rendimiento académico, así como apoyar la toma de decisiones académicas e institucionales.

En este contexto, el uso de herramientas de cómputo en la nube y técnicas de aprendizaje automático (machine learning) representa una alternativa viable para la recolección, el procesamiento y el análisis de grandes volúmenes de datos de manera eficiente, segura y automatizada. Las plataformas de servicios en la nube facilitan la integración de procesos de almacenamiento, limpieza, análisis y visualización de información, lo que permite identificar patrones complejos y relaciones relevantes entre variables emocionales y académicas [3][4].

El objetivo de este trabajo es analizar el impacto del estado emocional en el rendimiento académico de estudiantes universitarios mediante el uso de encuestas estructuradas, herramientas de cómputo en la nube y modelos predictivos. La principal contribución del estudio radica en la integración de analítica emocional, aprendizaje automático y cómputo en la nube en un entorno académico real, bajo una arquitectura escalable, respetando principios éticos, de consentimiento informado y protección de datos personales, con el fin de generar información que apoye la toma de decisiones orientadas al bienestar estudiantil y la mejora del desempeño académico [5][6].

2 Estado del Arte

El estudio del impacto del estado emocional en el rendimiento académico ha cobrado relevancia en los últimos años debido al reconocimiento de que los procesos de aprendizaje no dependen únicamente de factores cognitivos, sino también de variables emocionales y contextuales. Diversas investigaciones han demostrado que emociones como la ansiedad, el estrés y la desmotivación influyen de manera directa en la atención, la memoria, la autorregulación y la toma de decisiones, lo que repercute significativamente en el desempeño académico de los estudiantes.

En este sentido, Acosta y Clavero [7] analizan la influencia de las emociones sobre el rendimiento académico desde una perspectiva psicológica, destacando que los estados emocionales negativos, como el estrés académico y la ansiedad, generan bloqueos cognitivos que afectan la concentración y la capacidad de retención de información. Los autores concluyen que el rendimiento académico no puede comprenderse de forma aislada de la dimensión emocional, ya que existe una relación bidireccional entre el desempeño escolar y el estado emocional del estudiante. Este trabajo sienta las bases teóricas para considerar las emociones como un componente clave dentro de los modelos educativos contemporáneos.

De manera complementaria, Moreano Núñez et al. [8] presentan una aproximación empírica al análisis del estado emocional y su relación con el rendimiento académico, enfatizando que factores como la presión académica, la carga de trabajo y el entorno personal influyen en la aparición de emociones negativas que impactan el desempeño. Los autores destacan que estudiantes con mayores niveles de ansiedad y estrés tienden a presentar un menor rendimiento académico, mientras que aquellos con mayor estabilidad emocional muestran mejores resultados. Estos hallazgos refuerzan la necesidad de implementar mecanismos de monitoreo emocional que permitan identificar oportunamente situaciones de riesgo académico.

Si bien estos estudios aportan evidencia sobre la importancia del estado emocional en el aprendizaje, su alcance suele limitarse a análisis tradicionales basados en métodos estadísticos convencionales y conjuntos de datos de tamaño reducido. En este contexto, el uso de tecnologías emergentes como el cómputo en la nube y el análisis de grandes volúmenes de datos representa una oportunidad para ampliar el alcance y la profundidad de este tipo de investigaciones.

Silva, Riera y Ríos [9] destacan el potencial del cómputo en la nube para el procesamiento y análisis de grandes cantidades de información, particularmente en entornos donde se requiere escalabilidad, flexibilidad y eficiencia computacional. Aunque su estudio se enfoca en aplicaciones biomédicas, los principios que describen son directamente aplicables al ámbito educativo, especialmente en el análisis de datos emocionales y académicos que requieren procesamiento automatizado y seguro. Los autores resaltan que la nube permite integrar distintas fuentes de datos y ejecutar modelos analíticos complejos sin necesidad de infraestructuras locales costosas.

Por su parte, Carlos [10] introduce el concepto de Big Data como un nuevo paradigma para el análisis de información, señalando que la capacidad de recolectar y procesar grandes volúmenes de datos heterogéneos permite identificar patrones y tendencias que no serían detectables mediante enfoques tradicionales. En el ámbito educativo, este paradigma resulta particularmente relevante para el análisis del comportamiento estudiantil, ya que posibilita el estudio conjunto de variables académicas, emocionales y contextuales. La integración de Big Data con técnicas de aprendizaje automático abre la puerta a modelos predictivos capaces de anticipar escenarios de bajo rendimiento o abandono escolar.

La flexibilidad del cómputo en la nube para el desarrollo de soluciones personalizadas es analizada por Ramos Pérez [11], quien destaca que las plataformas en la nube permiten adaptar la infraestructura tecnológica a las necesidades específicas de cada proyecto. Este enfoque resulta especialmente pertinente para sistemas de monitoreo académico y emocional, donde los volúmenes de datos pueden crecer de forma progresiva y requerir ajustes dinámicos en almacenamiento y capacidad de procesamiento. Además, el autor subraya que la nube facilita la automatización de procesos y la integración de servicios analíticos avanzados.

Desde una perspectiva organizacional, la Universidad Privada Dr. Rafael Belloso Chacín [12] analiza el impacto de la computación en la nube en los procesos empresariales, resaltando beneficios como la optimización de recursos, la reducción de costos y la mejora en la toma de decisiones basada en datos. Aunque el estudio se centra en el ámbito empresarial, sus conclusiones son extrapolables al

sector educativo, donde la gestión eficiente de información académica y emocional puede contribuir a la mejora de los servicios de apoyo estudiantil y al diseño de estrategias institucionales basadas en evidencia.

En relación con la gestión de datos, Vera et al. [13] realizan un estudio secundario sobre las características de las bases de datos para el cómputo en la nube, destacando aspectos clave como la disponibilidad, la escalabilidad y la seguridad de la información. Los autores enfatizan que la correcta selección de tecnologías de almacenamiento es fundamental para garantizar la integridad y confidencialidad de los datos, especialmente cuando se trata de información sensible, como los datos emocionales y personales de los estudiantes.

La dimensión ética y la protección de datos personales en entornos de cómputo en la nube es abordada por Cruz, Piccirilli y Florencia [14], quienes proponen un modelo de análisis para la gestión de datos personales en proyectos basados en la nube. Este modelo clasifica los datos en personales, sensibles y tratados por terceros, y resalta la importancia de aplicar estándares internacionales como la norma ISO/IEC 27018 para garantizar el cumplimiento normativo y la confianza en los proveedores de servicios en la nube. Este enfoque resulta especialmente relevante para investigaciones que involucran información emocional, dado el carácter sensible de este tipo de datos.

En conjunto, los trabajos revisados evidencian, por un lado, la influencia significativa del estado emocional en el rendimiento académico y, por otro, el potencial del cómputo en la nube y el análisis de datos para gestionar información compleja de manera eficiente y segura. Sin embargo, se identifica una brecha en la integración de ambos enfoques, ya que son escasos los estudios que combinan el análisis emocional, el uso de plataformas en la nube y modelos de aprendizaje automático en un entorno académico real. Esta investigación busca contribuir a cerrar dicha brecha mediante una propuesta metodológica que integra analítica emocional, cómputo en la nube y machine learning, considerando además principios éticos y de protección de datos personales.

3 Metodología

La metodología empleada en este estudio se diseñó para analizar la relación entre el estado emocional y el rendimiento académico de estudiantes universitarios, integrando técnicas de recolección de datos, cómputo en la nube y aprendizaje automático. El proceso metodológico se estructuró en siete fases, las cuales permitieron garantizar un flujo sistemático, escalable y reproducible para el análisis de los datos como se muestra en la Figura 1.

En la primera fase de definición y preparación de la información se definió el objetivo general del estudio, orientado a identificar el impacto del estado emocional en el rendimiento académico mediante el análisis de datos emocionales y académicos. Se establecieron las variables de interés, incluyendo factores emocionales (estrés, ansiedad, motivación y percepción de apoyo emocional), hábitos personales y académicos, así como el promedio académico. Asimismo, se seleccionó un enfoque cuantitativo basado en encuestas estructuradas y se consideraron principios éticos, como el consentimiento informado y la protección de los datos personales de los participantes.

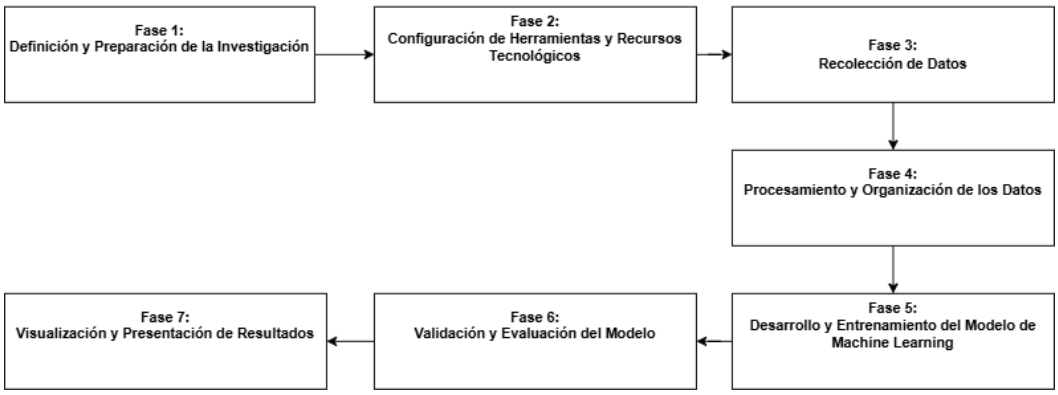


Figura 1. Fases de la Metodología.

En la segunda fase de Configuración de las herramientas y recursos tecnológicos, se diseñó una arquitectura tecnológica basada en cómputo en la nube para soportar el almacenamiento, procesamiento y análisis de la información. Se configuraron servicios para el resguardo de los datos recolectados, la automatización de tareas de limpieza y validación, el almacenamiento estructurado en bases de datos relacionales, el entrenamiento de modelos de aprendizaje automático y la generación de visualizaciones interactivas. Esta configuración permitió garantizar escalabilidad, seguridad y eficiencia en el manejo de los datos. La selección de herramientas de AWS como S3, Lambda, RDS, SageMaker y QuickSight. Se configuró el entorno para almacenar datos, automatizar procesos de limpieza, organizar información en bases de datos relacionales, entrenar modelos de machine learning y crear visualizaciones interactivas. Como se muestra en la Figura 2.

Figura 2. Herramientas tecnologías.

En la tercera fase de la recolección de datos se realizó mediante una encuesta digital aplicada a estudiantes universitarios de distintas generaciones. El instrumento incluyó preguntas demográficas, académicas y emocionales, así como escalas estandarizadas para la medición del estrés y la ansiedad. Los datos recolectados fueron almacenados automáticamente en el sistema de almacenamiento en la nube, asegurando su integridad y disponibilidad para las fases posteriores del estudio. En total, se obtuvieron 344 registros válidos.

En la fase 4 procesamiento y organización de los datos, los datos fueron sometidos a un proceso automatizado de limpieza y organización mediante AWS Lambda. Este proceso incluyó la

eliminación de inconsistencias, la estandarización de formatos, el manejo de valores faltantes y la validación de variables clave, particularmente el promedio académico. Posteriormente, la información procesada fue estructurada y almacenada en una base de datos en Amazon RDS, lo que facilitó su consulta SQL, análisis y conexión con otras herramientas analíticas. En la fase 5 de desarrollo y entrenamiento del modelo de machine learning, se desarrollaron modelos de aprendizaje automático supervisado con el objetivo de identificar patrones y relaciones entre el estado emocional y el rendimiento académico en Amazon SageMaker. Se exploraron diferentes configuraciones y algoritmos, ajustando sus parámetros para optimizar el desempeño del modelo. Esta fase permitió generar modelos capaces de reconocer tendencias y perfiles asociados a distintos niveles de desempeño académico. En la fase 6 de validación y evaluación del modelo el entreno fue validado utilizando datos no empleados durante la fase de entrenamiento, con el fin de evaluar su capacidad de generalización. Se analizaron métricas de desempeño y se realizaron ajustes iterativos para mejorar la precisión del modelo y reducir el sobreajuste. Esta validación permitió confirmar la utilidad del modelo para el análisis predictivo del rendimiento académico en función de variables emocionales. Finalmente, en la fase siete se desarrollaron visualizaciones y paneles interactivos que permitieron presentar de manera clara los resultados obtenidos, creando dashboards y gráficas con Amazon QuickSight. Estas representaciones gráficas facilitaron el análisis de la relación entre variables emocionales y académicas, así como la identificación de patrones relevantes y tendencias generales. La visualización de resultados constituyó una herramienta clave para la interpretación de los hallazgos y su comunicación a distintos públicos interesados.

4 Resultados

Se realizó un análisis exploratorio de los 344 registros obtenidos mediante encuestas aplicadas a estudiantes universitarios. Este análisis permitió identificar patrones generales, tendencias y relaciones entre variables emocionales, académicas y demográficas.

Distribución del Promedio Académico:

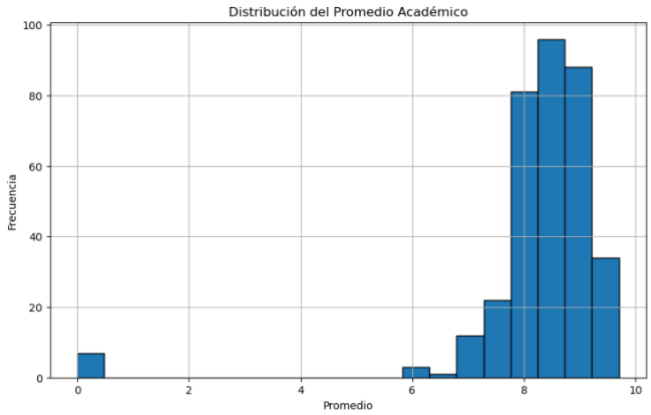


Figura 3. Distribución del 'Promedio'.

Como se muestra en la Figura 3. La mayoría de los estudiantes presentan promedios entre 7 y 10, con una alta concentración en valores de 8 y 9. Se observaron pocos casos con promedios menores a 7, lo cual sugiere un buen desempeño general, aunque también se detectaron algunos valores atípicos cercanos a 0.

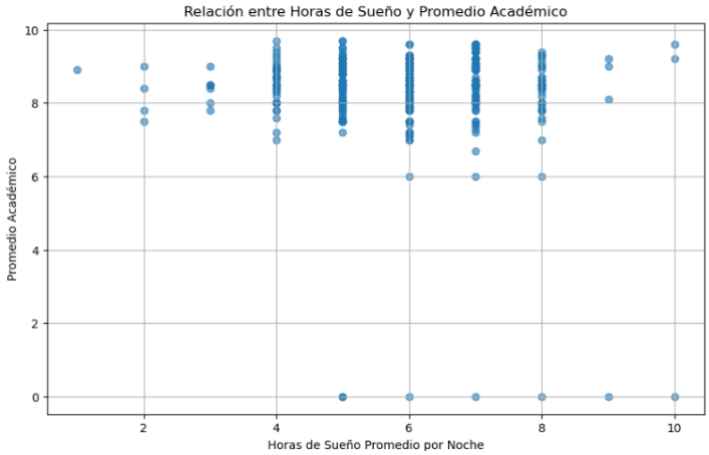


Figura 4. Horas de sueño y rendimiento.

Como se muestra en la Figura 4. La mayoría duerme entre 6 y 7 horas por noche. Un gráfico de dispersión entre horas de sueño y promedio académico mostró que no existe una correlación lineal clara, aunque los promedios más altos tienden a concentrarse entre 5 y 8 horas de sueño, lo que sugiere una relación indirecta con el rendimiento.

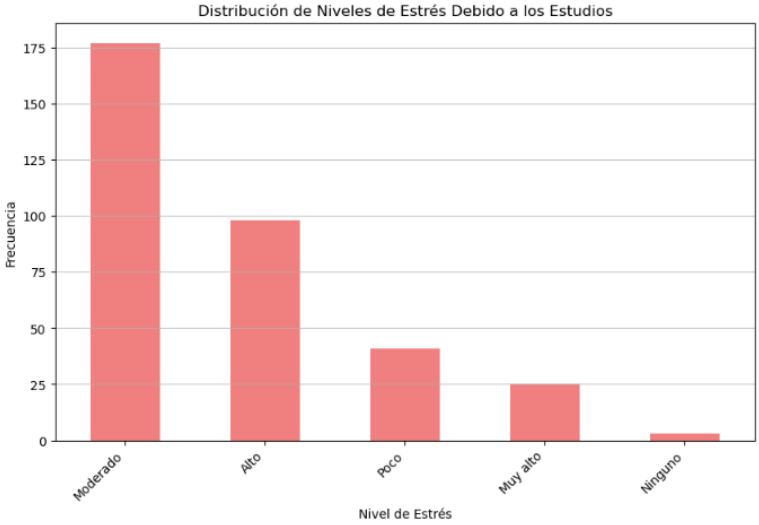


Figura 5. Niveles de estrés.

Como se muestra en la Figura 5.El nivel de estrés más reportado fue “Moderado”, seguido de “Alto”. Esto confirma que una gran parte de los estudiantes enfrentan presiones significativas en su vida académica.

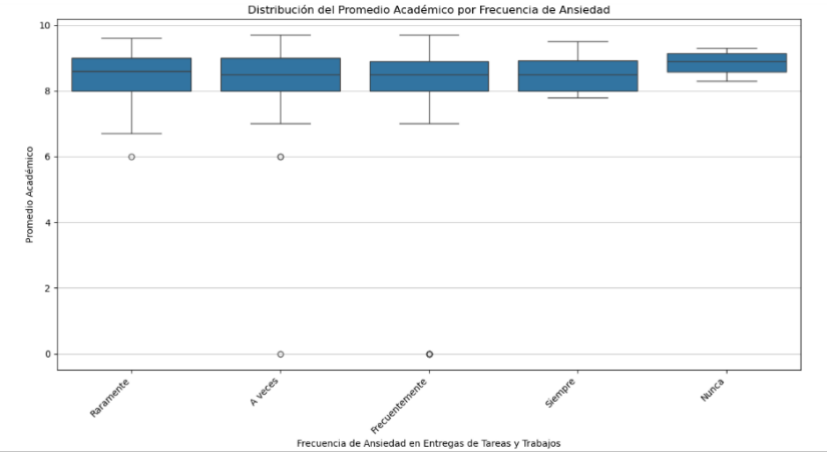


Figura 6. Frecuencia de ansiedad con el rendimiento.

Como se muestra en la Figura 6. Estudiantes con ansiedad frecuente presentaron un promedio académico ligeramente menor (media: 7.94) y mayor variabilidad. Aquellos que reportaron no experimentar ansiedad obtuvieron promedios más altos (media: 8.85), aunque con menor representación en la muestra.

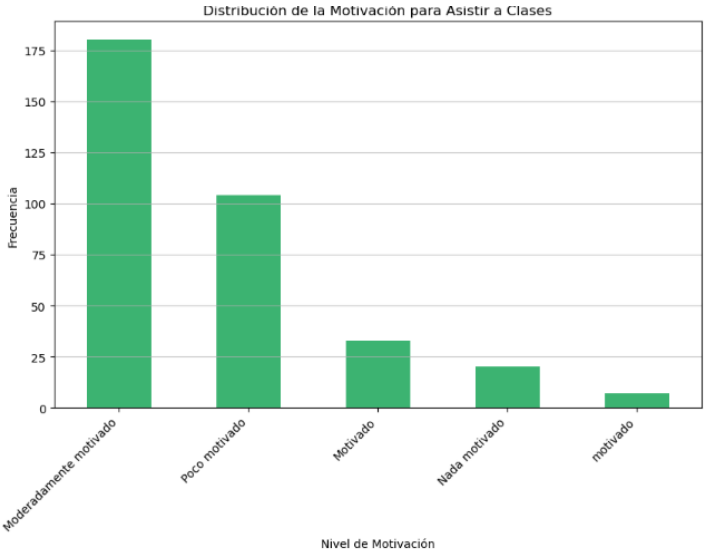


Figura 7. Motivación para ir a clases.

Como se muestra en la Figura 7. La mayoría de los estudiantes se describen como “Moderadamente motivados” o “Poco motivados”, lo cual podría estar influenciado por factores emocionales y académicos combinados.

Como se muestra en la Figura 8. Curiosamente, los estudiantes con niveles de estrés “Alto” o “Muy alto” obtuvieron promedios ligeramente superiores (media: 8.59) que los que reportaron niveles “Moderado” o “Poco”, aunque esto podría deberse a un factor de presión positiva en algunos casos.

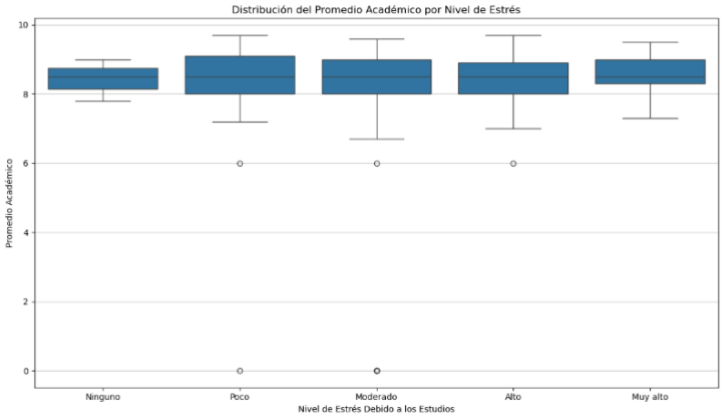


Figura 8. Relación de estrés con el promedio.

5 Conclusiones

El presente trabajo permitió demostrar que el estado emocional constituye un factor relevante que influye de manera significativa en el rendimiento académico de estudiantes universitarios. Mediante el uso de herramientas de cómputo en la nube y modelos de aprendizaje automático, fue posible recolectar, procesar, analizar y visualizar datos emocionales y académicos de forma escalable, automatizada y eficiente, lo que evidenció el potencial de estas tecnologías para el análisis integral del comportamiento estudiantil. El análisis exploratorio de los datos reveló patrones importantes en la relación entre variables emocionales y desempeño académico. En particular, los estudiantes que reportaron niveles elevados de ansiedad o estrés presentaron una mayor variabilidad en su rendimiento académico, mientras que aquellos con mejor calidad de sueño, mayores niveles de motivación y percepción de apoyo emocional tendieron a obtener promedios más altos. Estos hallazgos refuerzan la importancia de considerar la dimensión emocional como parte fundamental del proceso educativo. Por otra parte, los modelos predictivos desarrollados, a pesar de las limitaciones asociadas al tamaño y naturaleza del conjunto de datos, permitieron identificar perfiles de riesgo académico y establecer una base sólida para el diseño de futuras intervenciones personalizadas. En conjunto, los resultados obtenidos contribuyen al campo de la analítica educativa al integrar el análisis emocional, el cómputo en la nube y el aprendizaje automático, ofreciendo un enfoque innovador que puede apoyar la toma de decisiones orientadas al bienestar estudiantil y a la mejora del rendimiento académico.

Finalmente, este trabajo contribuye al campo de la analítica educativa al integrar la dimensión emocional, el cómputo en la nube y el aprendizaje automático en un entorno académico real, considerando además principios éticos y de protección de datos personales. Como trabajo futuro, se propone ampliar el conjunto de datos, incorporar variables adicionales y explorar modelos más avanzados que permitan mejorar la precisión predictiva y fortalecer el apoyo institucional al bienestar emocional y al rendimiento académico de los estudiantes.

Referencias

[1] Machine learning: Definición, Ventajas y Desventajas. (n.d.). Salesforce. https://www.salesforce.com/es/resources/definition/machine-learning/

[2] Agulló, M. J., Filella, G., Soldevila, A., & Ribes, R. (2011). Evaluación de la educación emocional en el ciclo medio de Educación Primaria. Revista de Educación, 354, 765-783.

[3] Machine learning: Definición, Ventajas y Desventajas. (n.d.). Salesforce. https://www.salesforce.com/es/resources/definition/machine-learning/

[4] Nubiral. (2024, July 15). Machine Learning con AWS | Nubiral. https://nubiral.com/machine-learning-aws/

[5] ¿Qué es el machine learning? - Explicación sobre el machine learning empresarial - AWS. (n.d.). Amazon Web Services, Inc. https://aws.amazon.com/es/what-is/machine-learning/

[6] Lee, M. (2017). Ethical Considerations in Emotion Research. Journal of Research Ethics, 13(3), 112-125.

[7] Acosta, F. P., & Clavero, F. H. (2017). LA INFLUENCIA DE LAS EMOCIONES SOBRE EL RENDIMIENTO ACADÉMICO. Ciencias Psicológicas, 29-39. https://doi.org/10.22235/cp.v11i2.1344

[8] Moreano Núñez, E. E., Saltos Salazar, L. F., Gavilanes López, W. L., & Andrade Alban, J. R. (2023). El estado emocional y el rendimiento académico una aproximación a la realidad actual. ConcienciaDigital, 6(1.4), 524-550. https://doi.org/10.33262/concienciadigital.v6i1.4.2012

[9] Silva, A., Riera, G., & Ríos, D. F. (2020). Cloud computing Applications for biomedical science. Reportes Científicos de la FACEN, 11(1), 39-50. https://doi.org/10.18004/rcfacen.2020.11.1.39

[10] Carlos, M. J. (2014, 1 noviembre). Big data. Un nuevo paradigma de análisis de datos. https://repositorio.comillas.edu/xmlui/handle/11531/4873

[11] Luis Gerardo Ramos Pérez. (2019). Computación en la nube – Análisis de proveedor Microsoft Azure para migración de aplicaciones y servicios, caso de estudio, Unidad para las Víctimas (UARIV). [Maestría en Ingeniería de Sistemas y Computación, Universidad Nacional de Colombia]. https://repositorio.unal.edu.co/bitstream/handle/unal/76338/LuisGerardoRamosPerez.2019.pdf?sequence=1&isAllowed=y

[12] Universidad Privada Dr. Rafael Belloso Chacín, URBE, Venezuela. (2017). LA NUEVA ERA DE LOS NEGOCIOS: COMPUTACIÓN EN LA NUBE. Revista Electrónica de Estudios Telemáticos, 15(2), 1856-4194. https://dialnet.unirioja.es/servlet/articulo?codigo=8577212

[13] Vera, R. A. A., Sánchez, N. G., Mendoza, J. C. D., & Ríos, B. L. F. (2023). Características de las Bases de Datos para el Cómputo en la Nube: Un Estudio Secundario. RISTI - Revista Ibérica de Sistemas E Tecnologias de Informação, 49, 115-130. https://doi.org/10.17013/risti.49.115-130

[14] Cruz, G. A. J., Piccirilli, D., & Florencia, P. C. M. (2016, 7 junio). Modelo de análisis relativo a la protección de datos personales para proyectos de cómputo en la nube. https://sedici.unlp.edu.ar/handle/10915/53258

Sistema de Gestión de Vehículos basada en Visión por Computadora para el Reconocimiento de Placas

Computer Vision-based Vehicle Management System for License Plate Recognition

Joselyne Janette Flores Emilio, Ana Luisa Ballinas Hernández, Judith Pérez Marcial, María Claudia Denicia Carral, Hector Medina Cruz

Benemérita Universidad Autónoma de Puebla, Complejo Regional Centro San José Chiapa. Calle 2 Sur, Ciudad Modelo, San José Chiapa, Puebla

joselyne.florese@alumno.buap.mx , {analuisa.ballinas, judith.perez, claudia.denicia, hector.medinacruz}@correo.buap.mx

Abstract

In the world of digital transformation, access automation has become central to operational efficiency for businesses. However, many establishments still rely on manual logs to manage vehicle access, which continues to directly impact operational efficiency and lead to data loss. This paper proposes the development of a management system that detects, recognizes, and records vehicle license plates in real time using ALPR (Automatic License Plate Recognition), as well as storing and managing the information through a web application. The proposed methodology comprises five phases: image acquisition, selection of a learning model, database creation, web system development, and functionality testing. The goal of this system is to provide an efficient and accurate vehicle access control solution for any facility.

Resumen

En el mundo de la transformación digital, la automatización de accesos se ha convertido en el eje central de la eficiencia operativa para las empresas. No obstante, gran parte de los establecimientos hacen uso de registros manuales para llevar el control de acceso vehicular, lo que sigue afectando directamente la eficiencia operativa y la perdida de información. En este trabajo se propone el desarrollo de un sistema de gestión que permite detectar, reconocer y registrar matriculas vehiculares en tiempo real a través el ALPR (reconocimiento automático de matrículas vehiculares), así como almacenar y gestionar la información a través de una aplicación web. La metodología propuesta abarca cinco fases: adquisición de imágenes, elección de modelo de aprendizaje, creación de una base de datos, desarrollo del sistema web y pruebas de funcionalidad. A través de este sistema se busca contar con un sistema eficiente y preciso para el control de acceso vehicular dentro de cualquier instalación.

Keywords and phrases: Visión por Computadora, Reconocimiento Automático de Placas, Acceso Vehicular, YOLO, Reconocimiento Óptico de Caracteres.

1 Introducción

Impulsados por la industria 4.0, la transformación digital exige modificar los procesos a pasos agigantados de tal manera que todos sean automatizados, sin embargo, muchas tareas cotidianas continúan utilizando métodos tradicionales.

En numerosos espacios como universidades, edificios, fraccionamientos, centros administrativos o estacionamientos los registros de entrada y salida en un principio se realizaba de forma manual, con el paso del tiempo se llevó a plantillas de Excel; lo que provocaba errores en los registros, como son, la pérdida o duplicidad de información, demora de tiempo y en muchos casos se pone en riesgo la seguridad de las instalaciones.

Los sistemas ALPR capturan automáticamente la imagen de la matrícula del vehículo, la transforman en caracteres alfanuméricos mediante reconocimiento óptico de caracteres o software similar, comparan el número de matrícula obtenido con una o más bases de datos de vehículos de interés para las fuerzas del orden y otras agencias, y alertan al agente cuando se observa un vehículo de interés [1]. Actualmente, con la llegada de las técnicas de la visión computacional el presente trabajo propone el desarrollo de un sistema de gestión automatizado basado en el reconocimiento de placas vehiculares en tiempo real.

El sistema tiene como objetivo detectar automáticamente las matrículas vehiculares para proceder a la extracción de información por medio del (OCR) reconocimiento óptico de caracteres y finalmente registrar los accesos a una base de datos que puede ser consultada en una aplicación web. Desde la adquisición de imágenes hasta la interacción con el usuario final, cada fase ha sido diseñada para garantizar que este sistema funcione en condiciones reales, reduciendo el margen de error del sistema.

2 Estado del Arte

El control de entradas y salidas de los vehículos ha sido motivo de investigación de varios autores, a continuación, se mencionan 6 artículos que han abordado el problema a través de la generación de contraseñas, códigos QR, RFID y mediante aprendizaje automático que son mencionadas de manera más detallada a continuación.

En el trabajo [2] se desarrolló un sistema de control de acceso mediante contraseñas que fue implementado en un estacionamiento en tiempos de pandemia, en donde el usuario tenía que ingresar una contraseña de 4 dígitos desde un teclado numérico.

En el trabajo [3] propuso un sistema digitalizado para mejorar la seguridad en una empresa por medio de lectores de código de barras utilizando una base de datos de Oracle, logrando una mejora en los tiempos de atención, reducción de reclamos internos y sobre todo mejoro en niveles de seguridad.

En el trabajo [4] diseñaron un sistema de control automatizado haciendo uso de códigos QR que permitían realizar la entrada y salida en la Universidad Tecnológica de la Habana (CUJAE), utilizando Raspherry y una cámara para la detección.

En el trabajo [5] hicieron uso del algoritmo OpenALPR reconocimiento automático de matrículas (ANPR), este algoritmo utiliza el reconocimiento óptico de caracteres lo que le permite leer las matrículas desde el flujo del video que se está captando en tiempo real, este sistema permitió automatizar el acceso para evitar robos en las instalaciones empresariales.

En el trabajo [6] aportan un sistema que optimiza el control de acceso vehicular a mediante la biometría, identificando datos de los empleados de identificación personal, a través de la huella dactilar. Finalmente, el sistema automatizado permitió la reducción en el tiempo de registro y aumento la seguridad de las instalaciones.

Finalmente, en el trabajo [7] presentaron un sistema combinando el ALPR inteligente con redes neuronales convolucionales (CNN) y mediaPipe para el reconocimiento facial. Desarrollando en Python con TensorFlow y OpenCV, el sistema mostró gran rendimiento en escenarios reales, y demostró que la función artificial y aprendizaje profundo es una tendencia efectiva de control vehicular.

3 Metodología Propuesta

Proponemos una metodología para el control automático de accesos vehiculares en instalaciones. Esta metodología consta de las fases que se muestran en la Figura 1 las cuales son descritas a continuación:

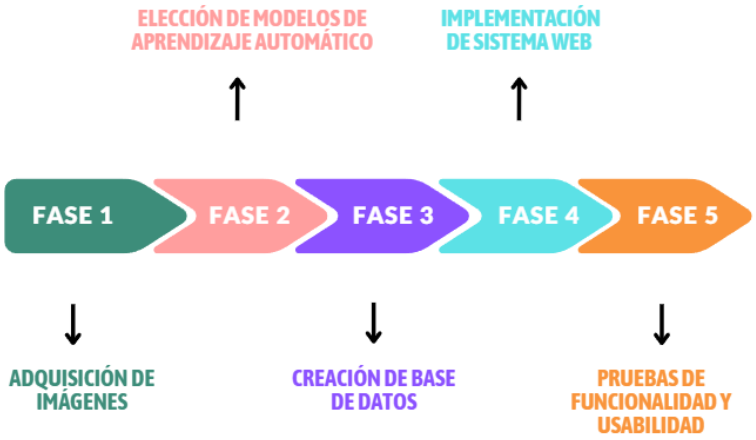


Figura 10. Metodología propuesta.

3.1 Adquisición de imágenes

Durante esta fase se recolecta un conjunto de imágenes con el objetivo de poder entrenar el algoritmo que permite hacer el reconocimiento de matrículas vehiculares. Las imágenes fueron obtenidas de las siguientes fuentes:

Data set público: se hizo uso de Kaggle, un data set público en donde se encontró un conjunto de imágenes de vehículos donde se podía distinguir perfectamente las placas vehiculares, el conjunto de estas imagines fue elegido de acuerdo con la calidad de la imagen.

Imágenes propias: la siguiente parte de las imágenes fueron recolectadas por cuenta propia utilizando la cámara de un teléfono celular. Para esto se realizaron diferentes tomas de vehículos, desde diversos ángulos y distancia, fueron tomadas de día. Con el objetivo de simular escenarios reales.

3.2 Elección de modelo de aprendizaje automático

Se realizaron pruebas con tres algoritmos, por un lado, se puso a prueba Easyocr, luego Pytesseract y finalmente YOLO; tras hacer un análisis comparativo de cada modelo y ponerlos a prueba con todas las imágenes que fueron recolectadas, se evaluaron los parámetros del número de errores que cada algoritmo tuvo, así como la velocidad, eficiencia y precisión del modelo y finalmente elegimos el modelo que tuvo menor número de errores.

YOLO resulto ser el método con la menor cantidad de errores, por lo que fue seleccionado como el más adecuado de acuerdo con su precisión. Una vez que se logró hacer la detección con YOLO se procedió a la etapa de reconocimiento de caracteres haciendo uso de Paddle OCR, una herramienta que hace uso de OCR. Existen diferentes razones por las que se eligió PaddleOCR, ya que permite reconocer los caracteres alfanuméricos con gran precisión, además de que es compatible con Python.

3.3 Creación de base de datos

Se desarrolló una base de datos relacional que nos permite gestionar la información que se obtendrá de cada vehículo, usuario y acceso. Entre los campos más importantes se encuentran el número de placa, la marca y modelo del vehículo, el tipo de unidad (ya sea automóvil, motocicleta, camioneta o autobús); datos del usuario como son el nombre.

Gracias a la tabla de registro se puede llevar un control más detallado donde se almacena la fecha y hora exacta de la entrada y salida del vehículo y finalmente el usuario tiene la opción de realizar una observación acerca del sistema, en donde se debe mencionar el id del acceso, la descripción y la hora.

El sistema de visión computacional se comunica con la base de datos lo que permite almacenar automáticamente los resultados de las placas obtenidas, así como la información de los accesos.

Esta información permite al usuario visualizar la información de la captura que se realizó por la cámara, facilitando la verificación de ello y realizar consultas posteriores.

3.4 Implementación de sistema Web

Para poder facilitar el monitoreo de la información recolectada por el algoritmo, se desarrolla un sistema web que a través de una interfaz los administradores pueden operar el sistema. El sistema

web permite visualizar los registros de entradas y salidas en tiempo real, así como realizar registros y modificar los datos de los vehículos y los usuarios.

La integración con la base de datos descrita anteriormente permite consultar, insertar y eliminar registros para poder tener una operación controlada desarrollado en Python con Visual Studio Code, el frontend se desarrolló con HTML5, CSS3 y JavaScript.

El sistema se organizó bajo la arquitectura cliente-servidor. El sistema posibilita gestionar tanto a los usuarios como a los vehículos permitiendo realizar registros, edición y eliminación de los datos.

3.5 Pruebas de validación y usabilidad

Una vez que elegimos el algoritmo se realizaron pruebas con el data set de 59 imágenes recolectadas de las cuales 29 fueron tomadas por la parte frontal del vehículo, 25 fueron tomadas por la parte trasera y 5 de motocicletas , midiendo su precisión y error con cada una de las imágenes; mediante el uso de una cámara Ip y el uso del protocolo RTSP (Real Time Streaming Protocol) se realizaron las pruebas de usabilidad a partir de medir la distancia máxima hasta donde el algoritmo es capaz de detectar las placas vehiculares, el tiempo que tarda para realizar dicha detección y la cantidad de errores que ha cometido en el reconocimiento de matrículas que son elementos que los usuarios evalúan a partir de su implementación.

4 Resultados

En la Figura 2 se presenta un esquema explicando las fases que lleva el proceso que realiza el algoritmo para la detección del reconocimiento de matrículas.

La figura 2.a muestra el primer paso que realiza el preprocesamiento de la imagen, en donde la imagen pasa por una serie de procesos, se convierte a escala de grises, se suaviza haciendo uso de un filtro bilateral para preservar los bordes y se aplica la binarización para mejorar el contraste,

La Figura 2.b permite ver la detección de objetos donde se extrae la caja delimitadora de cada objeto de interés, en este caso se marca en color verde la detección de la región de la matrícula y de azul cuando se detecta el vehículo (auto o motocicleta). La Figura 2.c posibilita la visualización del resultado, mostrando el texto de los caracteres alfanuméricos reconocido mediante el OCR.

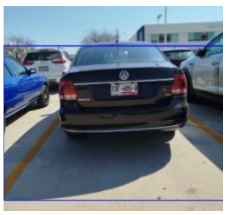
		
Procesamiento de la imagen	Detección de objetos	Visualización de resultados

Figura 2. Adquisición de imágenes.

Cada modelo fue probado con un total de 59 imágenes, de las cuales 29 son de placas delanteras, 25 de imágenes de placas traseras y 5 son de motocicletas. Se ha realizado un análisis de cada modelo, en donde a continuación se mostrará a través de porcentaje el número total de placas que fueron detectadas para cada uno, así como con datos numéricos, todo esto con la intención de poder encontrar el modelo más preciso. Después del análisis realizado, se concluyó que el modelo YOLO, al presentar mejores resultados, fue seleccionado para el procesamiento de las imágenes.

4.1 Sistema ANPR

ANPR es una tecnología de procesamiento de imágenes que captura imágenes de vehículos y, a partir de ellas, extrae información de sus matriculas y la traduce a formatos legibles por máquina, como cadenas de texto, que luego pueden procesarse e indexarse en una base de datos (ANPR International, 2021). Este proceso se compone de las fases que se pueden mostrar en la Figura 3:

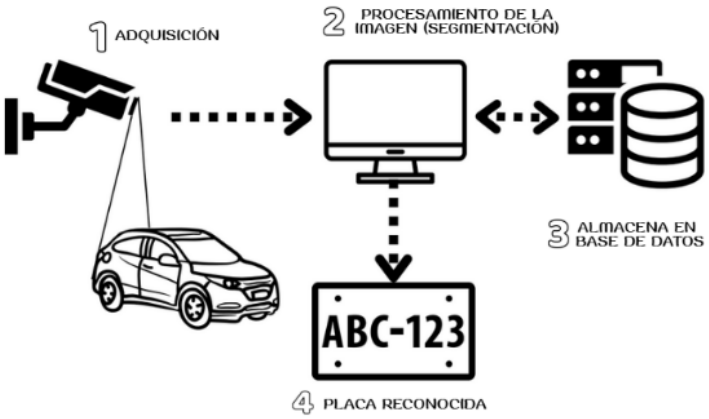


Figura 3. Esquema general de ANPR. Elaboración propia.

Como primer paso del método, se realiza la adquisición de la imagen, que por medio de una cámara captura al vehículo en tiempo real, después, la imagen capturada es enviada al sistema para su procesamiento donde se aplica la segmentación con el objetivo de identificar y usar el ROI (región de interés) es decir la matrícula.

Una vez que es localizada la matricula, el sistema procede a almacenar la información que ya fue registrada en la base de datos, para que finalmente, mediante el OCR (Reconocimiento Óptico de Caracteres) se puede extraer el texto de la placa vehicular.

4.2 Reconocimiento óptico de Caracteres (OCR)

El reconocimiento óptico de caracteres realiza la función de reconocer los caracteres alfanuméricos de una imagen y puede convertirlos en texto, el motor de OCR funciona de la siguiente manera como se muestra en la Figura 4. en donde se puede notar que en un principio se obtiene una imagen de entrada que va a ser analizada, posteriormente se realiza el procesamiento previo para realizar el reconocimiento de texto y se puedan encontrar la coincidencia de patrones y pueda extraer características y finalmente obtener el resultado.

Figura 4. Esquema general de ANPR. Elaboración propia.

5 Conclusiones

La implementación del proyecto demostró que la visión por computadora aplicada al reconocimiento automático de matrículas vehiculares es viable y eficiente para la gestión de accesos en el control vehicular. La implementación del modelo YOLOv8 combinado con PaddleOCR permitió tener resultados con alta precisión en el reconocimiento de caracteres, superando a otros enfoques como EasyOCR y Pytesseract, lo que lo hace más eficiente.

La integración del sistema de detección con una base de datos relacional posibilitó un registro estructurado y confiable de vehículos, usuarios, fechas y horarios, aportando trazabilidad y reduciendo la posibilidad de errores humanos propios de los registros manuales. A su vez, el

desarrollo de un sistema web cliente-servidor facilitó la consulta y gestión en tiempo real de la información, lo que incrementa la seguridad y eficiencia operativa.

Finalmente, se comprobó que la propuesta no solo resuelve una necesidad puntual de control vehicular, sino que también propone futuras mejoras como la incorporación de más condiciones de prueba (nocturnas, climáticas y en movimiento), la ampliación del dataset y la integración con otros mecanismos de seguridad.

Asimismo, como trabajo a futuro se propone como una aportación a la transformación digital e Industria 4.0, al integrar técnicas de inteligencia artificial, automatización y análisis de datos en tiempo real dentro de una arquitectura escalable y replicable.

Referencias

[1] International Association of Chiefs of Police, Automated License Plate Recognition (ALPR), The IACP. May (2025).

[2] Galindo, S., Hernández, H: Sistema de control de acceso vehicular basado en contraseña para estacionamientos, diseñado y simulado en Verilog. Universidad Distrital Francisco José de Caldas. Vol.17, pp. 61-68. Junio (2020).

[3] P. Huillca Canales: Innovación en seguridad para mejorar los controles de acceso en la empresa San Miguel industrias PET. Lima, Perú: Universidad San Ignacio de Loyola, (2021).

[4] L. M. Palmera Quintero, M. A. Rincón Pinzón, y L. O. Ramírez Orellano: Control de acceso vehicular mediante machine learning, Revista Colombiana de Tecnologías de Avanzada (RCTA), vol. 2, no. 44, pp. 1–10, jul. (2024).

[5] J. J. Moreno Miguel, E. Gutiérrez Jiménez, K. A. Jiménez Martínez, y B. R. Zamudio Rodríguez: Reconocimiento automático de matrículas para detección de vehículos con reporte de robo, Innovación y Desarrollo Tecnológico Revista Digital, vol. 16, no. 1, pp. 309–316, ene. (2024).

[6] J. A. Villarroel and J. M. Villarroel: Diseño e Implementación del Sistema Control e Identificación Vehicular en Parqueadero Ubicado en el Edificio Administrativo de la UPS Aplicando Tecnología RFID, Univ. Politécnica Salesiana, Guayaquil, Ecuador, (2010).

[7] A. J. Arellano Aray y C. B. Cayetano Tumbaco: Estudio y diseño geométrico del acceso de entrada y salida al recinto Cerecita provincia del Guayas, aplicando criterios de seguridad vial, ESPOL, Facultad de Ingeniería en Ciencias de la Tierra (FICT), (2023).

Computación de Próxima Generación en el Contexto de Tendencias Emergentes en Inteligencia Artificial, Analítica Avanzada y Defensa Digital

se terminó de editar en Diciembre de 2025 en la
Facultad de Ciencias de la Computación
Av. San Claudio y 14 Sur Jardines de San Manuel
Ciudad Universitaria
C.P. 72570

Computación de Próxima Generación en el Contexto de Tendencias Emergentes en Inteligencia Artificial, Analítica Avanzada y Defensa Digital

se terminó de editar en Diciembre de 2025 en la
Facultad de Ciencias de la Computación
Av. San Claudio y 14 Sur Jardines de San Manuel
Ciudad Universitaria
C.P. 72570

Computación de Próxima Generación en el Contexto de Tendencias Emergentes en Inteligencia Artificial, Analítica Avanzada y Defensa Digital

se terminó de editar en Diciembre de 2025 en la
Facultad de Ciencias de la Computación
Av. San Claudio y 14 Sur Jardines de San Manuel
Ciudad Universitaria
C.P. 72570

Computación de Próxima Generación en el Contexto de Tendencias Emergentes en Inteligencia Artificial, Analítica Avanzada y Defensa Digital

se terminó de editar en Diciembre de 2025 en la
Facultad de Ciencias de la Computación
Av. San Claudio y 14 Sur Jardines de San Manuel
Ciudad Universitaria
C.P. 72570

Computación de Próxima Generación en el Contexto de Tendencias Emergentes en Inteligencia Artificial, Analítica Avanzada y Defensa Digital

se terminó de editar en Diciembre de 2025 en la
Facultad de Ciencias de la Computación
Av. San Claudio y 14 Sur Jardines de San Manuel
Ciudad Universitaria
C.P. 72570

Computación de Próxima Generación en el Contexto de Tendencias Emergentes en Inteligencia Artificial, Analítica Avanzada y Defensa Digital

se terminó de editar en Diciembre de 2025 en la
Facultad de Ciencias de la Computación
Av. San Claudio y 14 Sur Jardines de San Manuel
Ciudad Universitaria
C.P. 72570

Computación de Próxima Generación en el Contexto de Tendencias Emergentes en Inteligencia Artificial, Analítica Avanzada y Defensa Digital

se terminó de editar en Diciembre de 2025 en la
Facultad de Ciencias de la Computación
Av. San Claudio y 14 Sur Jardines de San Manuel
Ciudad Universitaria
C.P. 72570

Computación de Próxima Generación en el Contexto de Tendencias Emergentes en Inteligencia Artificial, Analítica Avanzada y Defensa Digital

se terminó de editar en Diciembre de 2025 en la
Facultad de Ciencias de la Computación
Av. San Claudio y 14 Sur Jardines de San Manuel
Ciudad Universitaria
C.P. 72570

Computación de Próxima Generación en el Contexto de Tendencias Emergentes en Inteligencia Artificial, Analítica Avanzada y Defensa Digital

se terminó de editar en Diciembre de 2025 en la
Facultad de Ciencias de la Computación
Av. San Claudio y 14 Sur Jardines de San Manuel
Ciudad Universitaria
C.P. 72570

Computación de Próxima Generación en el Contexto de Tendencias Emergentes en Inteligencia Artificial, Analítica Avanzada y Defensa Digital

Coordinado por María del Carmen Santiago Díaz