

COMPARACIÓN DE ALGORITMOS PARA SLAM SOBRE UN ROBOT MÓVIL SANITIZADOR EN ENTORNOS CONTROLADOS

COMPARISON OF SLAM ALGORITHMS ON A MOBILE SANITIZING ROBOT IN CONTROLLED ENVIRONMENTS

Israel Soto

Universidad Autónoma de Ciudad Juárez, México
angel.soto@uacj.mx

Andrés de la Rosa García

Universidad Autónoma de Ciudad Juárez, México
andres.delarosa@uacj.mx

Francesco García

Universidad Autónoma de Ciudad Juárez, México
francesco.garcia@uacj.mx

Israel U. Ponce

Universidad Autónoma de Ciudad Juárez, México
israel.ulises@uacj.mx

Alma Guadalupe Rodríguez

Universidad Autónoma de Ciudad Juárez, México
alma.rodriguez.ram@uacj.mx

Recepción: 11/diciembre/2025

Aceptación: 28/diciembre/2025

Resumen

Este estudio presenta la evaluación comparativa de cuatro algoritmos SLAM 2D (Gmapping, Karto, Cartographer y Hector SLAM) en un robot omnidireccional de cuatro ruedas basado en ROS. La implementación experimental incluye pruebas sistemáticas en entornos controlados utilizando sensores del tipo LiDAR, IMU y encoders en los motores de las ruedas en entornos controlados. Los resultados mostraron diferencias significativas, lo que permitió realizar una evaluación en base a la precisión de localización, rendimiento computacional, robustez ante oclusiones, y en corrección de bucles. Los resultados presentan las variaciones de los errores de mapeo según el algoritmo y condiciones del entorno. Se establecen las bases para el desarrollo futuro de un sistema de sanitización autónoma mediante luz UV,

donde la precisión en navegación resulta crítica. Las conclusiones incluyen según requisitos de distintas aplicaciones en navegación autónoma recomendaciones para la selección de algoritmos SLAM.

Palabras Clave: Navegación autónoma, robot omnidireccional, ROS, SLAM 2D, sanitización UV.

Abstract

This study presents the comparative evaluation of four 2D SLAM algorithms (Gmapping, Karto, Cartographer, and Hector SLAM) on a ROS-based four-wheeled omnidirectional robot. The experimental implementation included systematic testing in controlled environments using LiDAR sensors, IMUs, and encoders on the wheel motors. The results showed significant differences, allowing for an evaluation based on localization accuracy, computational performance, robustness to occlusions, and loop correction. The results present the variations in mapping errors depending on the algorithm and environmental conditions. They are established lays the groundwork for the future development of an autonomous UV sanitization system, where navigation accuracy is critical. The conclusions include recommendations for the selection of SLAM algorithms, based on the requirements of different applications in autonomous navigation.

Keywords: 2D SLAM, autonomous navigation, omnidirectional robot, ROS, UV sanitization.

1. Introducción

La robótica móvil autónoma ha experimentado un desarrollo acelerado en las últimas décadas, impulsado por progresos notables en algoritmos de percepción sensorial, capacidades de procesamiento computacional y sistemas inteligentes de navegación. En el centro de esta transformación tecnológica se sitúa el desafío de Localización y Mapeo Simultáneo (SLAM), que constituye uno de los problemas fundamentales más complejos en el campo de la robótica móvil.

SLAM capacita a un robot para construir un mapa de su entorno desconocido mientras determina simultáneamente su ubicación precisa dentro de dicho mapa,

una habilidad esencial para lograr navegación autónoma eficaz [Durrant, 2006], [Bailey, 2006]. El significado de SLAM en robótica móvil autónoma se extiende más allá de las aplicaciones convencionales de exploración y navegación. En el contexto post-pandémico actual, ha emergido una demanda crítica de sistemas robotizados capaces de ejecutar tareas de desinfección autónoma en ambientes complejos como instalaciones de hospitales, edificios corporativos, centros comerciales y espacios públicos [Zhao, 2021], [Alatise, 2020]. Estos sistemas demandan capacidades de navegación extremadamente precisas para asegurar una cobertura integral y efectiva de las superficies que requieren desinfección, especialmente cuando se implementan tecnologías de sanitización mediante radiación ultravioleta (UV), donde la exactitud en el posicionamiento y la planificación de trayectorias resulta fundamental para la eficacia del proceso de descontaminación [Jiménez, 2019], [Zeng, 2021].

Los robots omnidireccionales, distinguidos por su habilidad de desplazarse en cualquier dirección sin requerir reorientación previa, ofrecen beneficios sustanciales para aplicaciones de sanitización en espacios restringidos y con configuraciones de obstáculos complejas [Rubio, 2019], [Siegwart, 2011]. Su capacidad de maniobra superior y flexibilidad de movimiento los posiciona como candidatos óptimos para navegación en entornos estructurados como hospitales y laboratorios, donde la precisión y la capacidad de evasión de obstáculos son elementos críticos [Lynch, 2017]. No obstante, la implementación de algoritmos SLAM en robots omnidireccionales presenta desafíos particulares relacionados con la estimación odométrica, la correlación de información sensorial y la integración de datos cinemáticos específicos de estas plataformas móviles [Borenstein, 1997], [Dudek, 2010]. La elección de algoritmos SLAM bidimensionales para esta investigación se fundamenta en múltiples consideraciones técnicas y prácticas. Primero, la mayoría de las aplicaciones de sanitización se desarrollan en ambientes predominantemente horizontales, donde la representación bidimensional del espacio resulta suficiente para una navegación efectiva [Thrun, 2005]. Segundo, los algoritmos SLAM 2D proporcionan ventajas computacionales significativas en términos de eficiencia procesual y operación en tiempo real, aspectos fundamentales para sistemas

autónomos que deben funcionar continuamente durante períodos extendidos [Cadena, 2016a], [Bresson, 2017]. Tercero, la madurez tecnológica de los algoritmos SLAM 2D y su amplia disponibilidad en plataformas como ROS (Robot Operating System) facilita su implementación y validación en aplicaciones del mundo real [Quigley, 2009].

Los cuatro algoritmos seleccionados para evaluación: Gmapping, Karto, Cartographer y Hector SLAM, representan aproximaciones metodológicas distintas y complementarias para resolver el problema SLAM [Stachniss, 2016]. Gmapping, fundamentado en filtros de partículas Rao-Blackwellized, ha demostrado robustez en entornos dinámicos y cambiantes [Grisetti, 2007]. Karto SLAM utiliza técnicas de optimización basadas en grafos para corrección de bucles cerrados y refinamiento progresivo de mapas [Konolige, 2010]. Cartographer emplea un enfoque híbrido que integra SLAM local y global para aplicaciones de tiempo real [Hess, 2016]. Hector SLAM ofrece la capacidad distintiva de operar sin información odométrica, utilizando exclusivamente datos del sensor LiDAR, lo que resulta particularmente relevante para robots omnidireccionales donde la estimación precisa de odometría puede presentar desafíos significativos [Kohlbrecher, 2011], [Santos, 2013].

Esta investigación se propone realizar una evaluación comparativa sistemática del desempeño de estos cuatro algoritmos SLAM en el contexto específico de robots omnidireccionales destinados a aplicaciones de desinfección autónoma. El estudio se enfoca en analizar métricas críticas de rendimiento incluyendo precisión de mapeo, consistencia temporal, eficiencia computacional y robustez ante incertidumbres odométricas características de las plataformas omnidireccionales. Los resultados de esta evaluación proporcionarán fundamentos empíricos para la selección óptima de algoritmos SLAM en sistemas robotizados de sanitización, contribuyendo al desarrollo de soluciones autónomas más eficaces y confiables para la desinfección de ambientes críticos. Adicionalmente, este trabajo busca identificar las limitaciones específicas y oportunidades de mejora de cada algoritmo cuando se implementa en robots con capacidades de movimiento omnidireccional, estableciendo así las bases para futuras investigaciones en la optimización de sistemas SLAM para esta clase particular de plataformas móviles autónomas.

2. Métodos

El concepto de Simultaneous Localization and Mapping (SLAM) es fundamental en robótica y vehículos móviles, permitiendo a los sistemas entender y navegar por entornos sin conocimiento previo. SLAM resuelve la tarea de construir un mapa del entorno mientras localiza al agente dentro de ese mapa [Cadena, 2016].

SLAM se puede describir como un problema de estimación de estado en el que se modela tanto el camino del robot (o agente) como el mapa del entorno como variables aleatorias. Matemáticamente, el problema SLAM puede formularse utilizando filtros bayesianos o métodos de optimización basados en gráficos. Los filtros bayesianos, como el filtro de Kalman extendido, evalúan iterativamente las estimaciones del estado del robot y del mapa. Por otro lado, los métodos de optimización basados en gráficos representan las estimaciones de estado como nodos y las mediciones como bordes, utilizando técnicas como la optimización gráfica para mejorar estas estimaciones [Anand, 2021], [Cadena, 2016].

Los algoritmos SLAM se clasifican en dos categorías:

- *Los basados en filtros*, utilizan métodos estadísticos para actualizar las estimaciones del estado del vehículo.

Ejemplos clásicos son el Filtro de Kalman Extendido y el Filtro de Partículas. Estos métodos son adecuados para entornos donde las mediciones y movimientos del sensor pueden modelarse de manera probabilística [Schmuck, 2018].

- *Los basados en grafos*. emplean una representación gráfica para modelar las relaciones entre diversas posturas del robot y las características observadas en el entorno. Este enfoque es más efectivo en la corrección de errores históricos a través de técnicas de optimización basada en gráficos, como la optimización no lineal [Cadenal, 2016].

Para aplicar los algoritmos SLAM 2D, se emplean diversos tipos de sensores, como lo son:

- El LiDAR (Light Detection and Ranging) es ampliamente utilizado debido a su precisión y capacidad para proporcionar mapas detallados de los entornos [Saat, 2020].

- Las Cámaras monocular y stereo utilizan datos visuales para extraer características del entorno que son utilizadas tanto para la localización como para la construcción de mapas [Makhubela, 2018]
- El Sensor de radar proporciona robustez en condiciones climáticas adversas y es útil en entornos exteriores a gran escala [Hong, 2022], [Hong, 2022].
- La Odometría (encoders de ruedas) es fundamental para GMapping y Karto, ya que ayuda a estimar el movimiento del robot entre escaneos. Sin embargo, Hector SLAM puede funcionar sin odometría, aunque su precisión mejora si se incluye [Zhao, 2022].
- IMU (Unidad de Medición Inercial): Proporciona información sobre aceleración y rotación, mejorando la robustez y precisión, especialmente en entornos dinámicos o cuando hay deslizamientos de ruedas.

Estos sensores son integrados a menudo de manera complementaria para aprovechar sus fortalezas individuales en un entorno dado, incrementando la robustez y precisión del sistema SLAM [Saat, 2020]. La Tabla 1 presenta una relación de los algoritmos con los sensores utilizados en su implementación:

- **Gmapping.** Es un algoritmo de SLAM que utiliza una combinación del filtro de partículas y el filtro de Kalman. Este enfoque permite el manejo eficaz de la incertidumbre en la estimación de la localización y el mapeo simultáneamente [Abdelrasoul, 2016]. Es muy efectivo en entornos 2D, ofreciendo una buena precisión y es ampliamente utilizado debido a su implementación en el sistema operativo de robots (ROS). Sin embargo, Gmapping puede enfrentar dificultades en entornos muy dinámicos o grandes debido a la necesidad intensiva de recursos de computación [Zhang, 2023].

Tabla 1 Sensores utilizados por algoritmo.

Algoritmo	LiDAR	Cámara	Odometría	IMU
GMapping	Sí	Opcional	Sí	Opcional
Hector	Sí	Opcional	No	Opcional
Karto	Sí	Opcional	Sí	Opcional
Cartographer	Sí	Opcional	Sí	Sí

Fuente: elaboración propia

- **Hector SLAM.** Es reconocido por operar sin odometría, aprovechando únicamente las capacidades de alta frecuencia y resolución de los escáneres láser para la correlación de escaneos [Harik, 2018]. Este enfoque lo hace especialmente ventajoso para robots que operan en entornos con cambios rápidos o poco predecibles, como drones o robots en ambientes resbaladizos donde los métodos tradicionales de odometría fallarían [Yagfarov, 2018].
- **Karto SLAM.** Es reconocido por su facilidad de uso y su eficacia en entornos semi-estructurados e interiores; utiliza una arquitectura basada en grafos de posturas, donde las posturas del robot y las mediciones se representan como nodos y las conexiones entre ellas como bordes dentro de un grafo [Xuexi, 2019]. Emplea técnicas avanzadas de optimización gráfica para garantizar mapas consistentes, utilizándose ampliamente en escenarios donde la corrección de la deriva acumulada es crucial. Esta capacidad de cierre de bucles mejora significativamente la precisión en entornos más grandes y con mapas complejos [Xuexi 2019].
- **Cartographer.** Desarrollado por Google, es un sistema de SLAM que utiliza un enfoque híbrido de SLAM local y global para proporcionar localización y mapeo en tiempo real [Dwijotomo, 2020]. Utiliza el algoritmo Branch and Bound para la optimización global, mejorando la precisión y la robustez del mapeo, especialmente en grandes entornos interiores [Wei, 2025]. Cartographer está diseñado para trabajar en tiempo real, lo que permite su uso en aplicaciones donde la rápida reacción y adaptación al entorno es esencial, como robots de desinfección autónomos durante la pandemia de COVID-19 [Wei, 2025].

Robots Omnidireccionales en SLAM

Los robots omnidireccionales poseen mecanismos de ruedas que les permiten moverse en cualquier dirección sin necesidad de girar hacia la nueva dirección de movimiento. Esto se debe a diseños innovadores como el uso de ruedas con juntas de deslizamiento pasivas que permiten que el robot mantenga movilidad omnidireccional mientras se asegura una estructura robusta [Terakawa, 2018].

Además, los robots omnidireccionales a menudo presentan redundancia en sus actuadores, lo que significa que tienen más actuadores de los necesarios para sus grados de libertad de movimiento, lo que permite evitar configuraciones de singularidad y realizar tareas secundarias [Yi, 2002]. La odometría en los robots omnidireccionales presenta desafíos particulares debido a la complejidad de sus mecanismos de movimiento. La incertidumbre en la odometría influye directamente en la localización y la creación de mapas, y el error en la posición puede crecer sin límite en sistemas de navegación basada en recuento de pasos (dead-reckoning) [Hidalgo-Carrio, 2017]. Adicionalmente, la calidad de los datos de odometría interna puede conducir a resultados poco confiables, lo cual complica el modelado principal inicial necesario para una buena estimación con SLAM [Valiente, 2017].

La movilidad omnidireccional confiere ventajas significativas en espacios confinados al permitir maniobras precisas y rápidas. Estos robots pueden cambiar de dirección sin necesidad de realizar giros intensos, facilitando su uso en entornos con restricciones espaciales tales como fábricas y almacenes [Terakawa, 2018]. El diseño de los robots omnidireccionales considerablemente mejora su capacidad para navegar eficientemente en estos ambientes complejos y dinámicos. Estas características hacen de los robots omnidireccionales una herramienta poderosa en aplicaciones de SLAM, especialmente en entornos donde el espacio es limitado y la flexibilidad de movimiento es crucial.

Metodología:

- **Plataforma robótica experimental.** Es un robot omnidireccional que comercialmente se vende con fines educativos (Figura 1). Esta plataforma es llamada comercialmente como ROSMASTER X3-PLUS, sin embargo, en este trabajo solo será nombrada robot experimental. La Figura 2 presenta una explosión de los componentes básicos del robot experimental.
- **Sistema de percepción.** Consiste en un Sensor LiDAR, una IMU de 9 ejes y los encoder que están unidos al eje trasero de los motores de CD, Figura 3. El Sensor LiDAR, llamado YDLIDAR 4ROS, tiene solo 7,58 cm * 3,47 cm de tamaño. Cuenta con un controlador de motor con función de regulación

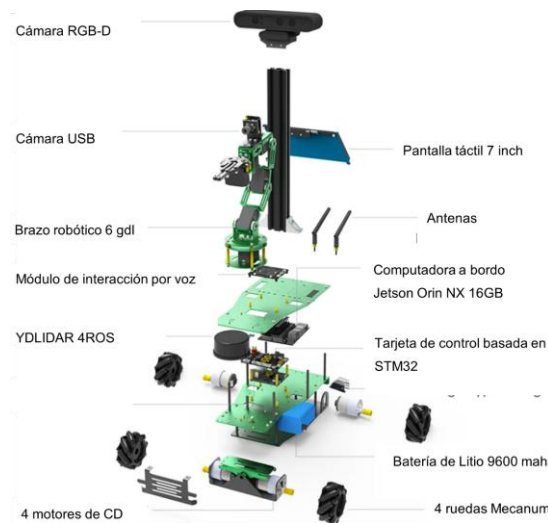
de velocidad en el interior y una cubierta protectora cerrada en el exterior, que es impermeable y a prueba de polvo. Utilizando la tecnología TOF, la frecuencia de muestreo es 2000 *veces/s*, el radio de escaneo puede alcanzar aproximadamente 30 *m*. Debido a que puede resistir el fuerte entorno de luz de 100 *KLux*, los usuarios pueden usarlo para lograr un rango y mapeo de alta precisión sin importar el interior o el exterior. La tarjeta de control robótico ROS es a la vez una placa de expansión para la unidad de control del vehículo robótico ROS y una placa de desarrollo STM32. Integra configuraciones importantes como la unidad de control STM32 y la IMU de 9 ejes, y proporciona 4 canales para control de motores con encoder.

- **Arquitectura de control.** La tarjeta de control principal consiste en una Jetson Orin NX con 16 *GB* de RAM y un SSD de 128 *GB*. Sobre esta computadora a bordo se tiene instalado Ubuntu 22.04 y ROS con su distribución Noetic.
- **Nodos y tópicos implementados.** Para implementar los algoritmos evaluados bajo el sistema operativo de robots (ROS), se deben considerar ciertos nodos y tópicos que faciliten la integración y ejecución. a continuación, se mencionan solo los principales. En cada caso, el uso de un sensor láser (LiDAR) es clave, junto con los datos de odometría y las transformaciones necesarias para asegurar la cooperación entre los diferentes elementos del sistema SLAM, facilitando así la localización y mapeo simultáneos [Harik, 2018], [Kamarudin, 2018], [Xuexi, 2019].



Fuente: elaboración propia

Figura 1 Plataforma robótica experimental ROSMASTER X3-PLUS.



Fuente: elaboración propia

Figura 2 Componentes de hardware del robot experimental.



Fuente: elaboración propia

Figura 3 Componentes del sistema de percepción.

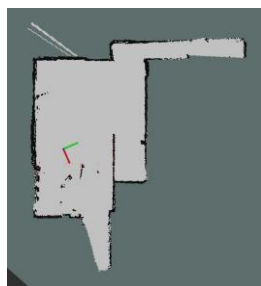
3. Resultados

Los resultados experimentales se obtuvieron mediante la implementación sistemática de los cuatro algoritmos SLAM 2D en el robot omnidireccional durante múltiples sesiones de prueba en entornos controlados:

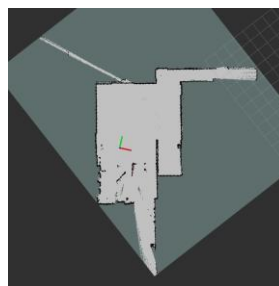
- **Evaluación cualitativa.** Los mapas 2D generados por cada algoritmo revelan diferencias significativas en la calidad de reconstrucción geométrica y la coherencia espacial. Las figuras presentadas muestran los resultados de mapeo para un entorno de prueba que incluye habitaciones conectadas, pasillos y obstáculos diversos. El mapa generado por Gmapping que se muestra en la Figura 4a presenta una reconstrucción geométrica sólida con

bordes bien definidos y una representación coherente de las paredes principales del entorno. Se observa una ligera rugosidad en los contornos, característica típica del enfoque basado en filtros de partículas.

El algoritmo logró una correcta detección de la geometría general del espacio, incluyendo la habitación principal y el pasillo de conexión. Sin embargo, se identificaron algunas inconsistencias menores en áreas con alta densidad de obstáculos pequeños. Los resultados de Karto SLAM (Figura 4b) demuestran una excelente capacidad de corrección de bucles, evidenciada en la consistencia geométrica superior del mapa final. Las líneas de las paredes aparecen más suaves y regulares comparadas con Gmapping, resultado de la optimización basada en grafos que caracteriza a este algoritmo. La representación del pasillo es particularmente precisa, mostrando una alineación correcta entre las diferentes secciones del entorno. La resolución de detalles finos es notable, especialmente en las esquinas y transiciones entre espacios. El mapa que se presenta en la Figura 5a, producido por Cartographer, exhibe la mayor precisión en términos de definición de bordes y coherencia espacial global. La representación del entorno muestra líneas extremadamente limpias y bien definidas, con una excelente preservación de las proporciones geométricas reales del espacio. El algoritmo demostró capacidad superior para manejar las transiciones entre diferentes áreas del entorno, manteniendo la consistencia topológica a lo largo de toda la sesión de mapeo. La calidad del mapa refleja la efectividad del enfoque híbrido local-global implementado en Cartographer.



a) Mapa por Gmapping.



b) Mapa por Karto SLAM.

Fuente: elaboración propia

Figura 4 Mapas generados por los algoritmos Gmapping y Karto SLAM.

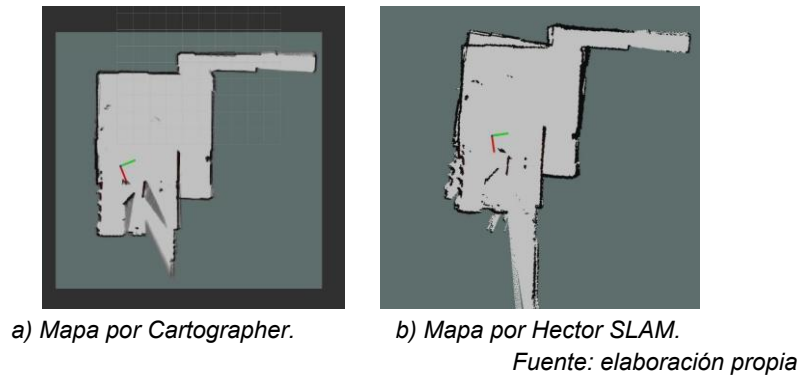


Figura 5 Mapas generados por los algoritmos Cartographer y Hector SLAM.

Los resultados de Hector SLAM muestran una reconstrucción geométrica adecuada (Figura 5b), aunque con algunas irregularidades en los bordes que reflejan la ausencia de información odométrica en el proceso de mapeo. A pesar de esta limitación, el algoritmo logró una representación funcional del entorno, con una correcta identificación de las características estructurales principales. La capacidad de operar únicamente con datos LIDAR resultó en un mapeo exitoso, aunque con menor refinamiento en comparación con los algoritmos que integran información odométrica.

El análisis de las cuatro figuras revela diferencias importantes en la consistencia topológica y la preservación de relaciones espaciales:

- ✓ Precisión en Esquinas y Ángulos: Cartographer demostró la mayor precisión en la preservación de ángulos rectos y esquinas, seguido por Karto SLAM. Gmapping mostró una tendencia a suavizar esquinas agudas, mientras que Hector SLAM presentó cierta variabilidad en la definición de ángulos.
- ✓ Continuidad de Paredes: Todos los algoritmos lograron mantener la continuidad de las paredes principales, aunque con diferentes niveles de suavidad. Karto SLAM y Cartographer produjeron las líneas más continuas y regulares.
- ✓ Representación de Espacios Abiertos: La representación de áreas abiertas fue consistente entre todos los algoritmos, aunque Cartographer mostró la mejor definición de los límites entre espacios navegables y obstáculos.

- Evaluación cuantitativa.** Los resultados cuantitativos se basaron en la comparación con mapas de referencia obtenidos mediante medición manual del entorno de prueba. Los resultados de localización mostraron que Cartographer alcanzó la mayor precisión con un error promedio de posición de 1.8 *cm* y error angular de 0.7 °. Karto SLAM siguió con errores de 2.4 *cm* y 1.1 °, mientras que Gmapping registró 3.2 *cm* y 1.8 °. Hector SLAM, operando sin odometría, mostró errores mayores de 4.6 *cm* y 2.9 °. Gmapping demostró la mayor robustez ante oclusiones temporales del sensor LIDAR, manteniendo la coherencia del mapa durante interrupciones de hasta 82.1 *segundos*. Karto SLAM mostró buena recuperación después de oclusiones, aunque requirió un tiempo mayor para re-localización. Cartographer presentó excelente robustez a corto plazo, pero mostró degradación en oclusiones prolongadas. Hector SLAM fue el más sensible a oclusiones debido a su dependencia exclusiva de datos LIDAR, Tabla 2.

Tabla 2 Comparación entre algoritmos.

Algoritmo	Error RMS Paredes (<i>cm</i>)	Error Angular Promedio (°)	Precisión Esquinas (<i>cm</i>)	Coherencia Global (%)
Gmapping	4.2 ± 1.8	2.1 ± 0.9	5.8 ± 2.3	87.3
Karto SLAM	3.1 ± 1.2	1.4 ± 0.6	3.9 ± 1.7	92.7
Cartographer	2.3 ± 0.8	0.9 ± 0.4	2.8 ± 1.1	96.2
Hector SLAM	5.7 ± 2.4	3.2 ± 1.4	7.1 ± 3.2	82.1

Fuente: elaboración propia

- Síntesis de resultados.** Los resultados experimentales revelan que Cartographer ofrece el mejor balance entre precisión de mapeo y localización, aunque con mayor costo computacional. Karto SLAM proporciona excelente precisión con eficiencia computacional moderada. Gmapping destaca por su robustez y eficiencia, siendo ideal para aplicaciones con recursos limitados. Hector SLAM ofrece la mayor eficiencia computacional y capacidad única de operación sin odometría, aunque con menor precisión absoluta. Estos resultados establecen las bases para la selección informada de algoritmos según los requisitos específicos de aplicaciones de sanitización autónoma, donde la precisión, eficiencia y robustez son factores críticos para el éxito operacional del sistema.

4. Discusión

La implementación de algoritmos SLAM 2D como Gmapping, Karto, Cartographer, y Hector sobre robots omnidireccionales para la sanitización de ambientes controlados presenta una serie de ventajas y desafíos únicos. Los robots omnidireccionales ofrecen la flexibilidad de movimiento necesaria para maniobrar en espacios confinados y de difícil acceso, lo cual es crucial en ambientes donde la exposición eficiente de luz ultravioleta es esencial para una desinfección efectiva. Estos entornos, que requieren una alta precisión en la navegación y mapeo, se benefician significativamente de la capacidad de los robots para realizar movimientos laterales y rotaciones complejas sin necesidad de maniobras de retroceso [Wei, 2025], [Xuexi, 2019]. Sin embargo, el éxito de estos algoritmos depende en gran medida de la calidad de los datos de seguridad obtenidos de los sensores, destacando la necesidad de integrar múltiples tecnologías de detección para mejorar la precisión y evitar fallas comunes asociadas con el uso exclusivo de tecnologías de percepción monocromática [Miettinen, 2007], [Wei, 2018]. Además, aunque los robots omnidireccionales permiten maniobras avanzadas, pueden verse afectados por problemas de calibración y mantenimiento debido a su complejidad mecánica y requisitos energéticos.

5. Conclusiones

La implementación de algoritmos SLAM 2D, incluidos Gmapping, Karto, Cartographer, y Hector, en robots omnidireccionales representa una solución prometedora para la sanitización de ambientes controlados con luz ultravioleta. Estos algoritmos, junto con la agilidad proporcionada por los robots omnidireccionales, permiten la creación de mapas de alta precisión, mejorando la eficiencia de las tareas de desinfección y optimizando la cobertura de luz UV en áreas clave. Aunque existen desafíos relacionados con la integración de sensores y el mantenimiento del hardware, las mejoras constantes en la tecnología de sensores y algoritmos de procesamiento prometen superar estas barreras, ofreciendo una herramienta robusta y eficaz para la desinfección de espacios complejos [Wei, 2025], [Xuexi, 2019]. Este enfoque tiene el potencial de mejorar

considerablemente las estrategias de control de infecciones en una variedad de entornos, desde hospitales hasta instalaciones industriales.

6. Referencias y Bibliografía

- [1] Abdelrasoul, Yassin, Patrick Sebastian, y Abu Bakar Sayuti Hm Saman, (2016). A quantitative study of tuning ROS gmapping parameters and their effect on performing indoor 2D SLAM, pp. 1-6. <https://doi.org/10.1109/roma.2016.7847825>.
- [2] Alatisse, M. B., & Hancke, G. P., (2020). A review on challenges of autonomous mobile robot and sensor fusion methods. *IEEE Access*, 8, pp. 39830-39846.
- [3] Anand, Ruchika, Andreas S Reichert, y Arun Kumar Kondadi, (2021). Emerging Roles of the MICOS Complex in Cristae Dynamics and Biogenesis. *Biology* 10 (7): 600. <https://doi.org/10.3390/biology10070600>.
- [4] Bailey, T., & Durrant-Whyte, H., (2006). Simultaneous localization and mapping (SLAM): part II. *IEEE Robotics & Automation Magazine*, 13(3), pp. 108-117.
- [5] Borenstein, J., Everett, H. R., Feng, L., & Wehe, D., (1997). Mobile robot positioning: Sensors and techniques. *Journal of Robotic Systems*, pp. 231-249.
- [6] Bresson, G., Alsayed, Z., Yu, L., & Glaser, S., (2017). Simultaneous localization and mapping: A survey of current trends in autonomous driving. *IEEE Transactions on Intelligent Vehicles*, 2(3), pp. 194-220.
- [7] Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., & Leonard, J. J., (2016). Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on Robotics*, 32(6), pp. 1309-1332.
- [8] Cadena, Cesar, John J Leonard, Ian Reid, et al., (2016). Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. *IEEE Transactions on Robotics* 32 (6), pp. 1309-32. <https://doi.org/10.1109/tro.2016.2624754>.
- [9] Dudek, G., & Jenkin, M., (2010). Computational principles of mobile robotics. Cambridge University Press.

- [10] Durrant-Whyte, H., & Bailey, T., (2006). Simultaneous localization and mapping: part I. *IEEE Robotics & Automation Magazine*, 13(2), pp. 99-110.
- [11] Dwijotomo, Abdurahman, Hairi Zamzuri, Wan Muhd Hafeez Wan Azree, Mohd Azizi Abdul Rahman, y Mohd Hatta Mohammed Ariff, (2020). Cartographer SLAM Method for Optimization with an Adaptive Multi-Distance Scan Scheduler. *Applied Sciences* 10. 347. <https://doi.org/10.3390/app10010347>.
- [12] Grisetti, G., Stachniss, C., & Burgard, W., (2007). Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Transactions on Robotics*, 23(1), pp. 34-46.
- [13] Harik, El Houssein, y Audun Korsæth, (2018). Combining Hector SLAM and Artificial Potential Field for Autonomous Navigation Inside a Greenhouse. *Robotics* 7 (2): 22. <https://doi.org/10.3390/robotics7020022>.
- [14] Hess, W., Kohler, D., Rapp, H., & Andor, D., (2016). Real-time loop closure in 2D LiDAR SLAM. *IEEE International Conference on Robotics and Automation*, pp. 1271-1278.
- [15] Hong, Ziyang, Andrew Wallace, Sen Wang, y Yvan Petillot, (2022). RadarSLAM: A robust simultaneous localization and mapping system for all weather conditions. *The International Journal of Robotics Research* 41 (5), pp. 519-42. <https://doi.org/10.1177/02783649221080483>.
- [16] Jiménez-Cano, A. E., Sanchez-Cuevas, P. J., Grau, P., Ollero, A., & Heredia, G., (2019). Contact-based bridge inspection multirotors: Design, modeling, and control considering the ceiling effect. *IEEE Robotics and Automation Letters*, 4(4), pp. 3561-3568.
- [17] Kohlbrecher, S., von Stryk, O., Meyer, J., & Klingauf, U., (2011). A flexible and scalable SLAM system with full 3D motion estimation. *IEEE International Symposium on Safety, Security, and Rescue Robotics*, pp. 155-160.
- [18] Konolige, K., Grisetti, G., Kümmerle, R., Burgard, W., Limketkai, B., & Vincent, R., (2010). Efficient sparse pose adjustment for 2D mapping. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 22-29.
- [19] Lynch, K. M., & Park, F. C., (2017). *Modern robotics: mechanics, planning, and control*. Cambridge University Press.

- [20] Makhubela, Jabulani K, Tranos Zuva, y Olusanya Yinka Agunbiade, (2018). A Review on Vision Simultaneous Localization and Mapping (VSLAM), 1-5. <https://doi.org/10.1109/iconic.2018.8601227>.
- [21] Miettinen, M., M. Ohman, A. Visala and P. Forsman, (2007). Simultaneous Localization and Mapping for Forest Harvesters, Proceedings 2007 IEEE International Conference on Robotics and Automation, Rome, Italy, pp. 517-522, doi: 10.1109/ROBOT.2007.363838.
- [22] Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., & Ng, A. Y., (2009). ROS: an open-source Robot Operating System. ICRA workshop on open source software, 3(3.2), 5.
- [23] Rubio, F., Valero, F., & Llopis-Albert, C., (2019). A review of mobile robots: Concepts, methods, theoretical framework, and applications. International Journal of Advanced Robotic Systems, 16(2), 1729881419839596.
- [24] Saat, Shahrizal, Mzm Tumari, Wn Abd Rashid, y Ms Saealal, (2020). HECTORSLAM 2D Mapping for Simultaneous Localization and Mapping (SLAM). Journal of Physics: Conference Series 1529 (4): 042032. <https://doi.org/10.1088/1742-6596/1529/4/042032>.
- [25] Santos, J. M., Portugal, D., & Rocha, R. P., (2013). An evaluation of 2D SLAM techniques available in robot operating system. IEEE International Symposium on Safety, Security, and Rescue Robotics, pp. 1-6.
- [26] Schmuck, Patrik, and Margarita Chli, (2018). CCM-SLAM: Robust and efficient centralized collaborative monocular simultaneous localization and mapping for robotic teams. Journal of Field Robotics 36 (4): pp. 763-81. <https://doi.org/10.1002/rob.21854>.
- [27] Siegwart, R., Nourbakhsh, I. R., & Scaramuzza, D., (2011). Introduction to autonomous mobile robots. MIT press.
- [28] Stachniss, C., Leonard, J. J., & Thrun, S., (2016). Simultaneous localization and mapping. In Springer handbook of robotics, pp. 1153-1176. Springer.
- [29] Thrun, S., Burgard, W., & Fox, D., (2005). Probabilistic robotics. MIT press.
- [30] Wei, H., X. -e. Li, Y. Shi, B. You and Y. Xu, (2018). Multi-sensor Fusion Glass Detection for Robot Navigation and Mapping, 2018 WRC Symposium on

- Advanced Robotics and Automation (WRC SARA), Beijing, China, pp. 184-188, doi: 10.1109/WRC-SARA.2018.8584213.
- [31] Wei, Su Jingyang Wen, Tao Qin, y Jia Wang, (2025). Indoor Localization and Path Planning of Mobile Disinfection and Epidemic Prevention Robot. *Recent Patents on Mechanical Engineering* 18 (3): pp. 319-29. <https://doi.org/10.2174/0122127976297666240510054304>.
- [32] Xuexi, Zhang, Lu Guokun, Fu Genping, Xu Dongliang, y Liang Shiliu, (2019). SLAM Algorithm Analysis of Mobile Robot Based on Lidar. 2019 Chinese Control Conference (CCC), pp. 4739-45. <https://doi.org/10.23919/ChiCC.2019.8866200>.
- [33] Yagfarov, Rauf, Mikhail Ivanou, y Ilya Afanasyev, (2018). Map Comparison of Lidar-based 2D SLAM Algorithms Using Precise Ground Truth, pp. 1979-83. <https://doi.org/10.1109/icarcv.2018.8581131>.
- [34] Zeng, Z., Lamata, P., Simaan, N., Giannarou, S., & Yang, G. Z., (2021). Collaborative robot-assisted endovascular catheterization with generative adversarial imitation learning. *IEEE Transactions on Medical Robotics and Bionics*, 3(1), pp.71-83.
- [35] Zhang, Bowen, Gang You, Lishuang Qu, Junting Qiu, y Shiyun Li, (2023). Application and Research on Improved Adaptive Monte Carlo Localization Algorithm for Automatic Guided Vehicle Fusion with QR Code Navigation. *Applied Sciences* 13 (21): 11913. <https://doi.org/10.3390/app132111913>.
- [36] Zhao, Jianwei, Shengyi Liu, y Jinyu Li, (2022). Research and Implementation of Autonomous Navigation for Mobile Robots Based on SLAM Algorithm under ROS. *Sensors* (Basel, Switzerland). <https://doi.org/10.3390/s22114172>.
- [37] Zhao, Y., Wang, L., Liu, Y., & Zhang, H., (2021). Autonomous disinfection robot for combating coronavirus: Design, development and experimental validation. *Robotics and Autonomous Systems*, 142, 103805.