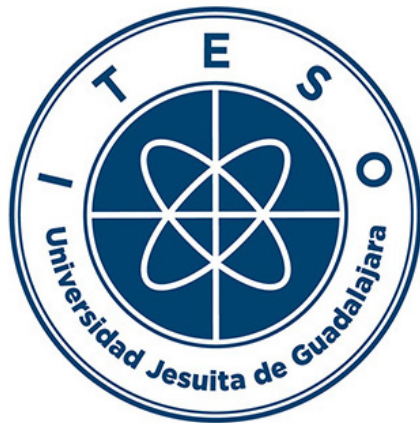


Instituto Tecnológico y de Estudios Superiores de Occidente

Reconocimiento de validez oficial de estudios de nivel superior según acuerdo secretarial 15018, publicado en el Diario Oficial de la Federación del 29 de noviembre de 1976.

Departamento de Electrónica, Sistemas e Informática
Maestría en Sistemas Computacionales



LOCALIZACIÓN INDOOR BASADA EN LA RECONSTRUCCIÓN DE MAPAS A PARTIR DE NUBES DE PUNTOS

TRABAJO RECEPCIONAL que para obtener el **GRADO** de
MAESTRO EN SISTEMAS COMPUTACIONALES

Presenta: **LI. RAMON HERNÁNDEZ AGUILAR**

Director de TOG Dr. Luis Fernando Gutiérrez Preciado

Tlaquepaque, Jalisco. 24 de Agosto de 2026.

AGRADECIMIENTOS

Agradezco al Instituto Tecnológico y de Estudios Superiores de Occidente (ITESO) por el apoyo institucional brindado durante mis estudios de maestría. De manera especial, reconozco a SECIHTI por el respaldo otorgado mediante la beca 2056159, el cual fue fundamental para la realización de este Trabajo de Obtención de Grado.

Expreso un agradecimiento particular al doctor Luis Fernando Gutiérrez Preciado por su orientación, acompañamiento y retroalimentación a lo largo del desarrollo del proyecto. Asimismo, agradezco a las y los profesores del programa por sus enseñanzas y aportaciones, que enriquecieron significativamente este trabajo de igual forma agradezco a mi familia y seres queridos por su paciencia, comprensión y apoyo constante, los cuales fueron un pilar importante para culminar esta etapa académica.

DEDICATORIA

Dedico este trabajo, en primer lugar, a mi familia, por acompañarme de manera incondicional durante cada etapa de este camino. Por su paciencia, comprensión y apoyo en los momentos de mayor exigencia, y por recordarme constantemente que cada esfuerzo, por difícil que pareciera, tenía un propósito.

A quienes han estado a mi lado durante estos años de estudio, les dedico también este logro. Su confianza, sus palabras de aliento y su presencia fueron fundamentales para continuar avanzando incluso en aquellos momentos en los que el cansancio, las responsabilidades y los desafíos propios de este proceso hicieron más difícil el camino.

Dedico especialmente este trabajo a quienes me enseñaron, con su ejemplo, el valor de la disciplina, la perseverancia, la responsabilidad y el compromiso. Cada uno de estos principios estuvo presente a lo largo de esta etapa y fue indispensable para transformar una idea inicial en un proyecto concluido.

A las personas que han formado parte de mi crecimiento personal y profesional, y que de alguna manera contribuyeron a que pudiera llegar hasta este momento, les expreso mi gratitud y les dedico también una parte de este logro.

Finalmente, dedico este trabajo a mi familia y a mis seres queridos como testimonio de que los objetivos que requieren tiempo, esfuerzo y constancia pueden alcanzarse cuando se cuenta con determinación, apoyo y la voluntad de seguir aprendiendo.

Que la culminación de esta maestría represente no solamente un logro académico, sino también el resultado de todas las experiencias, aprendizajes, sacrificios y personas que hicieron posible llegar hasta aquí.

RESUMEN

La localización y orientación en espacios interiores representa un desafío debido a la degradación de señales GPS, cambios de iluminación, variaciones de perspectiva y ambigüedad visual. Para abordar esto, en este Trabajo de Obtención de Grado se desarrolló un flujo de relocalización visual indoor apoyado en un mapa métrico-topológico previamente construido con RTAB-Map (herramienta de mapeo en tiempo real basada en apariencia que genera mapas 3D y 2D a partir de sensores como LiDAR o cámaras de profundidad). El propósito fundamental consiste en estimar la posición en un plano 2D y la orientación de un usuario o dispositivo a partir de una única imagen de consulta.

El sistema emplea una arquitectura híbrida que combina métodos clásicos de visión por computadora con técnicas de aprendizaje profundo, operando en dos etapas principales. En la fase offline de preparación, se organizó una base de datos en SQLite con imágenes de un recorrido previo asociadas a sus coordenadas espaciales, y se calcularon embeddings (vectores numéricos de alta dimensión que representan de forma matemática las características visuales de una imagen) que fueron normalizados e indexados mediante la biblioteca FAISS (biblioteca de software optimizada para la búsqueda eficiente de similitudes y el agrupamiento de vectores densos a gran escala). En la fase online, el vector de una nueva imagen de consulta capturada por una cámara RGB o RGB-D se contrasta eficientemente contra dicho índice; los nodos candidatos recuperados se refinan mediante verificación geométrica y análisis de correspondencias locales para seleccionar la hipótesis de pose más consistente. Aunque la construcción del mapa admitió datos complementarios de sensores como LiDAR o IMU para fortalecer la estimación espacial, la relocalización en tiempo real se centra en el procesamiento visual de las imágenes ingresadas.

Para lograr una interpretación clara de los datos, la posición y la orientación estimadas se proyectan sobre un plano 2D empleando una transformación afín (que considera alineación, rotación y escalado) renderizada con OpenCV, la cual permite visualizar el nodo localizado y el grafo del recorrido. Con el objetivo de facilitar el uso del sistema y habilitar pruebas integrales, se construyó una infraestructura compuesta por una API REST encargada de ejecutar el proceso de relocalización y una interfaz web interactiva desarrollada con HTML, CSS y JavaScript para cargar imágenes de consulta y explorar el mapa.

La precisión y viabilidad del sistema se evaluó mediante tres modalidades de prueba. En una evaluación controlada con 20 consultas sintéticas, ORB y SuperPoint+LightGlue alcanzaron una exactitud Top-1 del 100 %, con tiempos promedio de 0.283 s y 0.851 s, respectivamente. En una evaluación manual con 10 imágenes reales se obtuvo una exactitud Top-1 del 40 % y Top-5 del 70 %. Finalmente, como prueba principal, se realizó una evaluación cross-session con 490 consultas provenientes de un recorrido independiente capturado 18 días antes del mapa de referencia. SuperPoint+LightGlue alcanzó una tasa de localización exitosa del 94.7 %, con error espacial mediano de 0.615 m, mientras que ORB alcanzó 91.6 % y 0.764 m, respectivamente. Estos resultados muestran la viabilidad del enfoque híbrido y permiten identificar las condiciones en las que la variación visual y la similitud estructural del entorno afectan la precisión de la relocalización.

TABLA DE CONTENIDO

LOCALIZACIÓN INDOOR BASADA EN LA RECONSTRUCCIÓN DE MAPAS A PARTIR DE NUBES DE PUNTOS	1
1. INTRODUCCIÓN	10
1.1. ANTECEDENTES	12
1.2. JUSTIFICACIÓN	14
<i>Razones científicas y tecnológicas</i>	<i>15</i>
<i>Razones económicas</i>	<i>16</i>
<i>Razones sociales y de accesibilidad</i>	<i>17</i>
1.3. PROBLEMA	17
A. PROBLEMA PRÁCTICO: ACCESIBILIDAD PARA PERSONAS CON DISCAPACIDAD VISUAL	17
B. PROBLEMA CIENTÍFICO Y TECNOLÓGICO	18
C. ALCANCE DEL PROBLEMA A RESOLVER	19
1.4. OBJETIVOS	20
A. OBJETIVO GENERAL	20
B. OBJETIVOS ESPECÍFICOS:	20
C. NOVEDAD CIENTÍFICA, TECNOLÓGICA O APORTACIÓN	21
I) APORTACIÓN CIENTÍFICA	22
II) APORTACIÓN TECNOLÓGICA	23
III) APORTACIÓN SOCIAL	23
2. ESTADO DEL ARTE O DE LA TÉCNICA	25
2.1. RTAB-MAP: CIERRE DE CICLO POR APARIENCIA	26
2.2. SLAM FOR VISUALLY IMPAIRED PEOPLE	27
2.3. MAST3R-SLAM: SLAM DENSO MONOCULAR 3D	28
2.4. ORB-SLAM3: SLAM VISUAL	29
2.5. VINS-FUSION: ESTIMACIÓN DE ESTADO VISUAL-INERCIAL	30
2.6. RTAB-MAP PARA ESCENARIOS REALES	31
2.7. NETVLAD: VISUAL PLACE RECOGNITION	32
2.8. HF-NET: LOCALIZACIÓN JERÁRQUICA	33
2.9. INLOC: LOCALIZACIÓN INDOOR CON MATCHING DENSO	35
2.10. SUPERGLUE: EMPAREJAR PUNTOS ENTRE IMÁGENES	36
3. MARCO TEÓRICO/CONCEPTUAL	39
3.1. LOCALIZACIÓN Y ORIENTACIÓN EN INTERIORES	40
3.2. ESTIMACIÓN DE POSE Y ORIENTACIÓN	41
A. POSE EN SEIS GRADOS DE LIBERTAD (6-DOF)	42
B. REPRESENTACIÓN DE ORIENTACIÓN: CUATERNIONES Y ÁNGULOS DE EULER	42
C. PROYECCIÓN A POSE PLANAR (X, Y, YAW)	43
3.3. SLAM (LOCALIZACIÓN Y MAPEO SIMULTÁNEO):	43
A. FUNDAMENTOS DE SLAM	44
B. CIERRE DE CICLOS	45

C.	OPTIMIZACIÓN DE GRAFOS DE POSES	45
D.	RTAB-MAP: ARQUITECTURA Y BASE DE DATOS SQLITE.....	46
3.4.	REPRESENTACIÓN DEL ESPACIO INTERIOR	47
A.	MAPAS DE OCUPACIÓN 2D	47
B.	GRAFOS TOPOLÓGICOS DE POSES	49
C.	PLANO DE PLANTA Y TRANSFORMACIÓN AFÍN PARA PROYECCIÓN 2D	49
3.5.	RELOCALIZACIÓN VISUAL BASADA EN APRENDIZAJE PROFUNDO.....	50
A.	REDES NEURONALES CONVOLUCIONALES (CNN) PARA REPRESENTACIÓN VISUAL	51
3.6.	ACCESIBILIDAD Y NAVEGACIÓN INDOOR PARA PERSONAS CON DISCAPACIDAD VISUAL	56
A.	NECESIDADES DE NAVEGACIÓN AUTÓNOMA EN INTERIORES	56
B.	SISTEMAS DE LOCALIZACIÓN COMO SOPORTE A LA MOVILIDAD INCLUSIVA	57
4.	ARQUITECTURA DE LOCALIZACIÓN INDOOR	59
4.1.	METODOLOGÍA DE TRABAJO ELEGIDA	62
4.2.	LEVANTAMIENTO DE REQUERIMIENTOS.....	63
A.	REQUERIMIENTOS FUNCIONALES	64
B.	REQUERIMIENTOS NO FUNCIONALES.....	65
4.3.	ARQUITECTURA DE LOCALIZACIÓN.....	66
4.4.	PROCESO DETALLADO.....	71
A.	PREPARACIÓN DEL ENTORNO Y HERRAMIENTAS	71
B.	CONSTRUCCIÓN DEL MAPA BASE (RTAB-MAP)	72
C.	LOCALIZACIÓN / RE-LOCALIZACIÓN SOBRE EL MAPA.....	73
D.	EXTRACCIÓN DE POSE Y ORIENTACIÓN	74
E.	TRANSFORMACIÓN Y PROYECCIÓN A UN PLANO 2D.....	75
F.	EXPOSICIÓN DEL RESULTADO (API / UI).....	76
4.5.	EVALUACIÓN	81
5.	RESULTADOS Y DISCUSIÓN	83
5.1.	RESULTADOS	85
5.2.	DISCUSIÓN	100
6.	CONCLUSIONES	103
6.1.	CONCLUSIONES.....	104
6.2.	TRABAJO FUTURO	105
7.	BIBLIOGRAFÍA	106

LISTA DE FIGURAS

FIGURA 1: FASE OFFLINE - FASE ONLINE	61
FIGURA 2: CASO DE USO	69
FIGURA 3: DIAGRAMA DE CLASES	70
FIGURA 4: MODOS DE OPERACIÓN	78
FIGURA 5: FLUJO DE DATOS	79
FIGURA 6: FLUJO DE GENERACIÓN DE MAPA 2D	80
FIGURA 7: FLUJO GENERAL DE PREPARACIÓN DEL DATASET.	87
FIGURA 8: FLUJO DE EVALUACIÓN SINTÉTICA COMPARATIVA ORB vs. SPL.	92
FIGURA 9: ORB vs. SUPERPOINT+LIGHTGLUE EVALUACIÓN SINTÉTICA.	93
FIGURA 10: ARQUITECTURA DEL MÉTODO HÍBRIDO DE RELOCALIZACIÓN.	95
FIGURA 11: HISTOGRAMA DE ERROR DE ORIENTACIÓN - ORB vs. SUPERPOINT+LIGHTGLUE 490 CONSULTAS	98
FIGURA 12: PROYECCIÓN DE LOS 1 131 NODOS DEL MAPA SOBRE EL PLANO 2D DEL ENTORNO.	99
FIGURA 13: NODO LOCALIZADO Y PROYECTADO SOBRE EL PLANO 2D CON ORIENTACIÓN ESTIMADA.	100

LISTA DE TABLAS

TABLA 1: COMPARACIÓN DE ENFOQUES DE LA PROPUESTA DE RELOCALIZACIÓN VISUAL INDOOR.	38
TABLA 2: ENTORNO DE HARDWARE Y SOFTWARE UTILIZADO EN LAS PRUEBAS	84
TABLA 3: RESUMEN DEL DATASET GENERADO DESDE RTAB-MAP.	86
TABLA 4: ARTEFACTOS GENERADOS PARA LA LOCALIZACIÓN	88
TABLA 5: RESULTADOS DETALLADOS DE LA EVALUACIÓN MANUAL CON 10 IMÁGENES REALES.	89
TABLA 6: RESUMEN ESTADÍSTICO DE LA EVALUACIÓN MANUAL (10 CONSULTAS).....	90
TABLA 7: COMPARACIÓN DE DESCRIPTORES EN EVALUACIÓN SINTÉTICA (20 CONSULTAS, CUDA).....	91
TABLA 8: COMPONENTES DEL MÉTODO HÍBRIDO DE LOCALIZACIÓN.	94
TABLA 9: EVALUACIÓN CROSS-SESSION CON 490 CONSULTAS — SUPERPOINT+LIGHTGLUE VS. ORB.....	97
TABLA 10: VISUALIZACIONES GENERADAS POR EL SISTEMA.	99

LISTA DE ACRÓNIMOS Y ABREVIATURAS

API		Es un conjunto de reglas y protocolos que permite que dos aplicaciones de software se comuniquen entre sí, intercambiando datos y funciones sin que necesites conocer su código interno.
CSS		Lenguaje de diseño utilizado para presentar y dar estilo visual a un documento estructurado en HTML o XML.
FAISS		Es una biblioteca de software libre creada por Meta que permite buscar y agrupar vectores densos de alta dimensión de manera ultrarrápida.
GPS		Es un sistema de navegación que utiliza una red de satélites para determinar la ubicación aproximada, la velocidad y la hora de cualquier objeto o persona en la Tierra.
HTML		Es el código estándar que se utiliza para crear y estructurar la estructura básica de cualquier página o sitio web.
IMU		Inertial Measurement Unit. Sensor que mide aceleraciones lineales y velocidades angulares y puede utilizarse como fuente de información para estimación de movimiento y orientación.
LiDAR		Es una tecnología de teledetección que utiliza pulsos láser para medir distancias con precisión milimétrica.
ORB		<i>Oriented FAST and Rotated BRIEF. Detector y descriptor de características locales diseñado para identificar y describir puntos de interés de manera eficiente.</i>
RANSAC		RANdom SAMple Consensus (Consenso de Muestra Aleatoria). Es un algoritmo iterativo utilizado en matemáticas, visión artificial y aprendizaje automático
REST		<i>REpresentational State Transfer</i> (Transferencia de Estado Representacional). Es un conjunto de reglas de arquitectura de software que permite a diferentes aplicaciones y sistemas comunicarse entre sí a través de internet.
RGB		es un acrónimo en inglés que significa Rojo, Verde y Azul (<i>Red, Green, Blue</i>).
RGB-D		Es un acrónimo que hace referencia a las imágenes o sensores que combinan el modelo de color RGB (Rojo, Verde y Azul) con un canal adicional de Profundidad (Depth).
SIFT		Es un algoritmo muy utilizado en visión por computadora para detectar y describir características locales en imágenes, permitiendo a las computadoras reconocer objetos sin importar si la imagen está rotada, ampliada o si cambia de tamaño
SLAM		Simultaneous Localization and Mapping. Conjunto de técnicas que permiten estimar simultáneamente la pose de un sensor o agente y construir una representación del entorno.
SURF		Algoritmo de visión por computador. Se utiliza para detectar, describir y reconocer objetos o rasgos clave dentro de imágenes o videos de forma rápida y precisa.
Top K Accuracy		Precisión Top-K es una métrica de evaluación en Machine Learning e Inteligencia Artificial. Mide la frecuencia con la que la etiqueta correcta se encuentra entre las (K) respuestas

1. INTRODUCCIÓN

La localización y orientación en espacios interiores se ha convertido en un problema relevante para múltiples dominios donde la navegación precisa es necesaria, tales como hospitales, universidades, aeropuertos, centros comerciales y edificios corporativos. Este problema adquiere especial importancia en términos de accesibilidad, ya que puede apoyar la movilidad y orientación de personas con discapacidad visual al proporcionar información espacial dentro de edificios. A diferencia de los entornos exteriores, donde el Sistema de Posicionamiento Global, conocido como GPS, puede ofrecer una referencia aproximada de ubicación, en interiores su desempeño se ve limitado por la atenuación de la señal, la interferencia estructural y la falta de precisión suficiente para tareas de navegación fina. En estos contextos, conocer dónde se encuentra una persona, robot o dispositivo, así como hacia dónde está orientado, es fundamental para habilitar servicios como navegación indoor, búsqueda de destinos, asistencia contextual y supervisión de recorridos.

Aunque en muchos edificios existen planos arquitectónicos, estos no siempre representan el estado real del entorno. Pueden quedar desactualizados debido a modificaciones físicas, reubicación de mobiliario, cambios en accesos o adecuaciones internas. Además, incluso cuando los planos están disponibles, no necesariamente se encuentran en formatos adecuados para integrarse directamente a soluciones digitales de navegación o sistemas de localización. Por ello, resulta necesario contar con mecanismos que permitan construir representaciones del entorno a partir de recorridos reales y que, además, puedan relacionar dichas representaciones con la posición estimada de un usuario o dispositivo dentro del espacio.

En respuesta a estas limitaciones, han surgido enfoques basados en sensores como cámaras RGB, cámaras RGB-D y LiDAR, así como algoritmos de Localización y Mapeo Simultáneo, conocidos como SLAM. Estos métodos permiten estimar la trayectoria de un sensor mientras se construye una representación del entorno observado. Sin embargo, convertir estos recorridos en un sistema práctico de localización indoor implica resolver retos adicionales. Entre ellos se encuentran la consistencia geométrica del mapa, la tolerancia a cambios de iluminación, la presencia de objetos dinámicos, la ambigüedad visual en espacios repetitivos y la necesidad de ofrecer una salida reutilizable, interpretable y adecuada para aplicaciones finales. En particular, para una solución orientada al usuario, no basta con estimar una pose en coordenadas espaciales; también es necesario proyectar esa información sobre una representación comprensible, como una vista 2D del edificio que muestre la ubicación y orientación estimadas.

En este contexto, RTAB-Map, por sus siglas en inglés Real-Time Appearance-Based Mapping, se ha consolidado como una herramienta ampliamente utilizada para la creación de mapas y la localización en entornos interiores. Su enfoque combina seguimiento visual, reconocimiento de lugares previamente visitados mediante cierre de ciclo y generación de representaciones 2D y 3D del entorno. No obstante,

RTAB-Map no resuelve de forma completa el puente entre la estimación de localización generada por el sistema y una representación operativa para el usuario final. Además, en escenarios reales pueden presentarse variaciones de iluminación, perspectiva, oclusiones parciales y cambios estructurales que dificultan la relocalización visual usando únicamente métodos clásicos de comparación de características.

Por esta razón, este Trabajo de Obtención de Grado (TOG) se plantea un flujo de trabajo orientado a la relocalización visual indoor, tomando como punto de partida un mapa previamente generado con RTAB-Map. A partir de dicho mapa se utiliza un grafo en el que cada nodo está asociado con una imagen del entorno y una pose conocida, compuesta por posición y orientación. El problema central consiste en estimar la pose de una imagen de consulta capturada posteriormente dentro del mismo entorno, comparándola contra las imágenes de referencia del mapa y seleccionando el nodo o conjunto de nodos con mayor evidencia visual y geométrica.

La propuesta integra una arquitectura híbrida que combina métodos clásicos de visión por computadora con técnicas de aprendizaje profundo. Por un lado, los métodos clásicos permiten extraer y comparar características locales de las imágenes, además de aplicar verificación geométrica para validar correspondencias. Por otro lado, los métodos basados en Deep Learning permiten obtener representaciones visuales más robustas ante cambios de iluminación, perspectiva y apariencia. Esta combinación permite realizar una búsqueda inicial de candidatos visualmente similares y, posteriormente, verificar geoméricamente dichos candidatos para mejorar la confiabilidad de la estimación.

El objetivo de este trabajo es desarrollar una estructura de pasos reproducible que, a partir de un mapa indoor 3D previamente generado con RTAB-Map, permita estimar la posición y orientación de un usuario, robot o dispositivo mediante una imagen de consulta capturada dentro del mismo entorno, y representar el resultado de forma clara sobre una visualización 2D. Para ello, se integran componentes para la construcción y consulta del mapa, la asociación entre imágenes y poses, la búsqueda de candidatos visualmente similares, la validación de correspondencias mediante criterios geométricos y la proyección de la pose estimada sobre un plano de visualización. Finalmente, el sistema busca actualizar de manera automática la posición y orientación estimadas a partir de imágenes capturadas por una cámara externa, facilitando así su uso en aplicaciones de navegación indoor, accesibilidad y asistencia espacial.

1.1. Antecedentes

Históricamente, la representación de espacios interiores para navegación se abordaba mediante levantamientos manuales, planos arquitectónicos o mapas construidos de forma estática. Sin embargo, estos recursos no siempre reflejan el estado real del entorno, pueden omitir elementos relevantes para la navegación, como obstáculos, accesos, mobiliario o modificaciones recientes, y en muchos casos no se encuentran disponibles en formatos adecuados para su integración con aplicaciones digitales. Frente a estas limitaciones, los enfoques basados en SLAM, por sus siglas en inglés Simultaneous Localization and Mapping, permiten construir una representación del entorno a partir de recorridos reales, estimando simultáneamente la trayectoria del sensor y la estructura espacial observada.

Los métodos de SLAM han evolucionado hacia distintas aproximaciones según el tipo de sensor utilizado y el nivel de detalle requerido. Por una parte, el SLAM visual emplea cámaras monoculares, estéreo o RGB-D para detectar y rastrear características visuales entre imágenes. Estos métodos resultan atractivos por su bajo costo y facilidad de despliegue, pero en ambientes interiores enfrentan dificultades asociadas con baja textura, patrones repetitivos, variaciones de iluminación, oclusiones y cambios de perspectiva. Por otra parte, los enfoques basados en profundidad, como aquellos que integran sensores RGB-D o LiDAR, proporcionan mediciones métricas directas que favorecen la consistencia geométrica del mapa y facilitan la estimación de movimiento. No obstante, estos sistemas también presentan retos relacionados con calibración, ruido del sensor, sincronización, reflejos, superficies poco estructuradas y escenas dinámicas.

Dentro de las herramientas utilizadas en investigación y desarrollo para mapeo y localización, RTAB-Map se ha consolidado como una alternativa relevante para entornos interiores. Su enfoque basado en grafos, su capacidad de realizar cierre de ciclo para reducir la deriva acumulada, y su compatibilidad con sensores RGB-D, cámaras y LiDAR, lo convierten en una plataforma flexible para construir mapas topológicos, mapas 2D y representaciones 3D a partir de recorridos reales. Además, su estructura basada en nodos permite asociar observaciones visuales con poses estimadas, lo cual resulta especialmente útil para tareas posteriores de localización y relocalización.

Sin embargo, construir un mapa no resuelve por sí solo el problema completo de localización indoor. En aplicaciones prácticas, también es necesario que el sistema sea capaz de relocalizarse dentro de un mapa previamente construido. Esto implica tomar una nueva observación, por ejemplo una imagen capturada por una cámara, compararla contra las observaciones almacenadas en el mapa, identificar el nodo o región más probable y recuperar una estimación de posición y orientación dentro de un sistema de referencia coherente. Esta etapa es crítica, ya que la relocalización puede fallar cuando existen cambios de iluminación, modificaciones en el entorno, objetos dinámicos o ambigüedad visual entre zonas similares.

Tradicionalmente, la relocalización visual se ha apoyado en métodos clásicos de visión por computadora, como la detección y descripción de características locales mediante algoritmos como SIFT, SURF, ORB o variantes basadas en esquinas. Estos métodos permiten comparar imágenes a partir de puntos de interés y establecer correspondencias entre ellas. Aunque son eficientes y ampliamente utilizados, su desempeño puede degradarse cuando las imágenes presentan cambios significativos de iluminación, perspectiva, textura o contenido dinámico. Para reducir falsos positivos, estos enfoques suelen complementarse con

técnicas de filtrado de correspondencias y verificación geométrica, como el uso de RANSAC para validar relaciones espaciales entre pares de imágenes.

En años recientes, el aprendizaje profundo ha introducido nuevas posibilidades para mejorar la robustez de los sistemas de relocalización visual. Modelos como SuperPoint [1] permiten aprender detectores y descriptores visuales más resistentes a variaciones del entorno, mientras que métodos de matching basados en atención, como LightGlue [2], buscan correspondencias más consistentes entre imágenes. Asimismo, el uso de embeddings visuales globales permite representar una imagen completa mediante vectores numéricos que pueden compararse de manera eficiente en grandes bases de datos. En este contexto, bibliotecas como FAISS [3] facilita la búsqueda eficiente de vecinos más cercanos mediante índices exactos o aproximados, permitiendo recuperar rápidamente los candidatos visualmente más similares dentro de un mapa con múltiples nodos.

A pesar de estos avances, los métodos puramente clásicos pueden ser sensibles a cambios visuales, mientras que los enfoques exclusivamente basados en aprendizaje profundo pueden requerir entrenamiento especializado, altos recursos computacionales o presentar menor interpretabilidad. Por ello, una estrategia híbrida resulta adecuada para escenarios de localización indoor, ya que permite combinar la eficiencia e interpretabilidad de los métodos clásicos con la robustez de las representaciones aprendidas. Bajo este enfoque, el aprendizaje profundo puede utilizarse para recuperar candidatos probables dentro del mapa, mientras que la verificación geométrica clásica puede validar si las correspondencias encontradas son espacialmente consistentes.

El proyecto desarrollado en este Trabajo de Obtención de Grado se ubica precisamente en esta línea. A partir de la base de datos de un mapa generado con RTAB-Map, se propone una estructura de procesamiento que permite extraer imágenes asociadas a nodos, recuperar sus poses, construir índices de búsqueda visual y ejecutar un proceso de relocalización sobre nuevas imágenes de consulta. El sistema contempla una etapa offline de preparación del mapa, en la que se organizan las imágenes, poses e índices; y una etapa online de localización, en la que una imagen nueva se compara contra el mapa para estimar la pose más probable. Esta organización permite separar la construcción del mapa del proceso de consulta, facilitando su integración con aplicaciones externas.

En particular, el módulo de relocalización visual basado en aprendizaje profundo implementa una arquitectura híbrida que combina técnicas clásicas de emparejamiento, recuperación mediante embeddings visuales, búsqueda eficiente con FAISS [3] y verificación geométrica mediante RANSAC [4]. Esta integración busca mejorar la robustez de la relocalización visual en interiores y entregar como salida una estimación interpretable de posición, orientación, nivel de confianza y candidatos alternativos. De esta manera, el sistema no solo busca construir una representación del entorno, sino también habilitar mecanismos prácticos para consultar dicha representación y utilizarla en aplicaciones de navegación, asistencia, supervisión o integración con plataformas web y móviles.

1.2. Justificación

La geolocalización en entornos interiores constituye un problema de alta relevancia práctica y social, especialmente en edificaciones complejas donde la señal GPS resulta insuficiente o, en muchos casos, inexistente. En espacios como campus universitarios, hospitales, centros comerciales y edificios corporativos, contar con mecanismos que permitan estimar con precisión la ubicación de una persona y la dirección hacia la que se desplaza contribuye a mejorar la navegación, reducir tiempos de traslado y optimizar el uso y la gestión de los espacios. Esta necesidad adquiere una importancia aún mayor en el caso de las personas con discapacidad visual, para quienes la movilidad y la orientación en interiores representan una barrera cotidiana. La ausencia de referencias visuales accesibles las obliga con frecuencia a depender de la memoria espacial, de la asistencia de terceros o de sistemas de señalización que no siempre son inclusivos, consistentes o suficientes para garantizar un desplazamiento autónomo y seguro.

En este contexto, el presente Trabajo de Obtención de Grado se justifica por el diseño, implementación y validación de una metodología reproducible para la relocalización visual en interiores, sustentada en la base de datos de mapas generados con RTAB-Map y en un módulo complementario desarrollado en Python y herramientas de aprendizaje profundo. La propuesta integra extracción y asociación de imágenes con poses, construcción de índices de búsqueda, generación de embeddings visuales, recuperación de candidatos mediante FAISS o mecanismos equivalentes, validación geométrica mediante correspondencias visuales y RANSAC, así como una estrategia de ranking para seleccionar la hipótesis de localización más confiable.

La relevancia de este trabajo radica en que no se limita al uso directo de una herramienta de mapeo existente, sino que propone un flujo de procesamiento modular y evaluable que permite extender las capacidades de relocalización de RTAB-Map. El sistema desarrollado articula métodos clásicos, como ORB y GFTT, con técnicas modernas de aprendizaje profundo, como embeddings visuales y modelos de matching, permitiendo comparar estrategias, analizar su desempeño y establecer criterios objetivos de validación. Esta integración aporta una base técnica flexible para aplicaciones de robótica móvil, navegación asistida, realidad aumentada, inspección automatizada y sistemas de orientación en espacios cerrados.

Además, la metodología propuesta favorece la trazabilidad y evaluación experimental del proceso de localización, al considerar métricas como precisión Top-1 y Top-5, número de correspondencias válidas, número de inliers, razón de inliers, confianza, error de posición, error de orientación y latencia. Esto permite valorar no solo si el sistema identifica correctamente una ubicación, sino también qué tan confiable, preciso y eficiente resulta el proceso de relocalización bajo distintas condiciones.

Razones científicas y tecnológicas

Desde la perspectiva científica y tecnológica, este trabajo aporta conocimiento aplicado en áreas como la visión por computadora, la robótica móvil, la relocalización visual, el SLAM y la cartografía digital de espacios interiores. Si bien existen diversas herramientas y algoritmos para la construcción de mapas, la localización y la navegación autónoma, en escenarios reales persiste un desafío relevante: lograr que los resultados sean reproducibles, verificables y transferibles a un flujo de trabajo práctico que pueda ser utilizado por otros sistemas o aplicaciones.

En particular, la relocalización visual en interiores continúa siendo un problema técnico complejo debido a factores como los cambios de iluminación, la variabilidad de perspectiva, la presencia de objetos dinámicos, la ambigüedad visual en pasillos o zonas repetitivas, y los cambios estructurales del entorno. Estos factores afectan directamente la calidad del mapa, la estabilidad de la localización y la confiabilidad de la estimación de pose. Por ello, resulta necesario definir y validar una arquitectura que no dependa únicamente de un método aislado, sino que combine distintas estrategias de análisis visual y verificación geométrica.

En este contexto, el presente trabajo propone una integración híbrida entre RTAB-Map y un módulo desarrollado en Python y aprendizaje profundo para relocalización visual robusta. Esta integración permite aprovechar, por un lado, las capacidades de RTAB-Map para la construcción de mapas, gestión de nodos, poses y grafos; y por otro, la flexibilidad de Python para implementar procesos adicionales de análisis, búsqueda, comparación, validación y exposición de resultados. El módulo propuesto articula métodos clásicos de visión por computadora, como ORB, GFTT, matching de descriptores y validación mediante RANSAC, con técnicas modernas de aprendizaje profundo, como embeddings visuales, búsqueda aproximada con FAISS y modelos de correspondencia visual basados en Deep Learning.

La relevancia científica del trabajo radica en la comparación e integración de enfoques clásicos y profundos para resolver el problema de relocalización visual indoor. Mientras los métodos clásicos ofrecen eficiencia, interpretabilidad y bajo costo computacional, los métodos basados en aprendizaje profundo pueden mejorar la robustez frente a variaciones visuales complejas. La combinación de ambos enfoques permite construir un sistema más equilibrado, capaz de recuperar candidatos visuales relevantes mediante embeddings y posteriormente validar geométricamente las coincidencias antes de estimar la pose final.

Desde el punto de vista tecnológico, el trabajo contribuye mediante la definición de un flujo integral que incluye la extracción de imágenes desde una base de datos de RTAB-Map, el parseo de poses, la construcción de índices, la generación de embeddings, la búsqueda de candidatos visuales, la verificación geométrica, el cálculo de métricas de confianza y la entrega de resultados mediante una interfaz programable. Esta secuencia permite transformar la información generada por RTAB-Map representada en nodos, poses y relaciones espaciales en una salida estructurada, evaluable y susceptible de ser consumida por aplicaciones externas.

Un aspecto particularmente importante es la orientación del usuario o del agente móvil. Muchas soluciones de localización se enfocan principalmente en estimar la posición dentro del mapa, pero no siempre ofrecen una salida consistente respecto a la orientación. Este componente es indispensable para aplicaciones de asistencia a la navegación, robótica móvil, realidad aumentada o sistemas que requieren indicar no solo

dónde se encuentra el usuario, sino también hacia dónde está orientado. En este trabajo, la extracción y procesamiento de la pose permite representar tanto la posición como la orientación, facilitando su posterior transformación a modelos 2D, visualizaciones o interfaces de navegación.

Razones económicas

Desde el enfoque económico, la localización en entornos cerrados demanda soluciones que, además de ofrecer precisión técnica, sean viables en términos de implementación, operación, mantenimiento y escalabilidad. Los métodos convencionales para construir, actualizar y validar información espacial de interiores suelen requerir personal especializado, tiempos prolongados de levantamiento, procesamiento manual de datos y, en algunos casos, infraestructura propietaria o sensores de alto costo. Estos factores incrementan significativamente el costo total de adopción, especialmente en espacios donde la distribución física cambia con frecuencia, como oficinas, laboratorios, almacenes, hospitales, centros educativos o instalaciones industriales.

En este contexto, el desarrollo de una metodología automatizada para apoyar la relocalización visual indoor representa una alternativa económicamente relevante. El proyecto propone aprovechar mapas e imágenes generadas a partir de recorridos en interiores, integrando herramientas consolidadas como RTAB-Map con componentes desarrollados en Python para organizar datos, extraer poses, construir índices visuales, calcular embeddings y realizar procesos de búsqueda y verificación geométrica. Esta integración permite reducir la dependencia de soluciones cerradas o plataformas comerciales, favoreciendo una implementación más flexible, reproducible y adaptable a distintos escenarios de uso.

Asimismo, la arquitectura híbrida planteada en el proyecto, que combina métodos clásicos de visión por computadora con técnicas de aprendizaje profundo, contribuye a optimizar recursos computacionales. En lugar de realizar comparaciones exhaustivas contra todas las imágenes del mapa, el sistema puede emplear mecanismos de recuperación visual para seleccionar candidatos relevantes y posteriormente aplicar verificación geométrica sobre un subconjunto reducido. Esto permite disminuir tiempos de procesamiento y mejorar la eficiencia operativa, lo cual es un aspecto importante cuando se busca implementar la solución en sistemas reales o con recursos limitados.

Razones sociales y de accesibilidad

Desde una perspectiva social, este trabajo adquiere relevancia por su contribución potencial a la accesibilidad y a la inclusión de personas con discapacidad visual. En entornos interiores, la movilidad autónoma de esta población suele verse restringida por la ausencia de información espacial accesible, la complejidad estructural de los edificios, la limitada disponibilidad de señalización táctil o auditiva y la variabilidad constante en la disposición del entorno. En este contexto, disponer de un sistema capaz de estimar la posición y la orientación dentro de un edificio a partir de un mapa previo representa una base tecnológica valiosa para el desarrollo de herramientas de asistencia que ofrezcan indicaciones más seguras, claras y contextualizadas. Ello puede traducirse en una reducción de riesgos, un aumento de la autonomía personal y una mejora sustancial en la experiencia de desplazamiento.

Asimismo, en espacios como hospitales y universidades, donde convergen usuarios temporales, personal de nuevo ingreso, estudiantes y visitantes, una solución de localización en interiores puede disminuir la desorientación, optimizar los tiempos de búsqueda y evitar desplazamientos innecesarios. En consecuencia, el diseño de una arquitectura que genere resultados interpretables, reproducibles y exportables para su integración en plataformas web o aplicaciones móviles fortalece el acceso equitativo a la información espacial y amplía las posibilidades de uso en contextos reales. Su impacto, por tanto, no se limita al ámbito técnico, sino que incide directamente en la movilidad, la accesibilidad y la calidad de vida de las personas.

1.3. Problema

La navegación en espacios interiores constituye un desafío relevante, ya que los sistemas de posicionamiento global (GPS) no proporcionan una localización precisa y confiable dentro de edificios. A ello se suma la ausencia frecuente de mapas digitales indoor disponibles, actualizados o con el nivel de detalle necesario para apoyar la orientación de los usuarios. Esta problemática se intensifica en entornos complejos y de alta afluencia, como campus universitarios, hospitales, centros comerciales y edificios corporativos, donde la orientación cotidiana depende en gran medida de referencias visuales, tales como señalamientos, letreros, números de aula, mapas impresos y pantallas informativas. No obstante, estas referencias suelen presentar deficiencias de claridad, consistencia y accesibilidad, además de variar entre edificios y niveles, o modificarse con el tiempo debido a remodelaciones, reubicación de servicios y ajustes operativos.

a. Problema práctico: accesibilidad para personas con discapacidad visual

La navegación en entornos interiores representa un desafío significativo para las personas con discapacidad visual, debido a la presencia de barreras que dificultan la percepción del espacio, la orientación y la toma de decisiones durante el desplazamiento. Actividades aparentemente simples, como identificar pasillos, reconocer accesos, localizar puertas específicas, encontrar elevadores o escaleras, interpretar intersecciones, detectar desniveles o evitar obstáculos temporales, se convierten en tareas de

alta exigencia cognitiva y física. Esta problemática se agrava por la ausencia, en la mayoría de los edificios, de sistemas de localización interior comparables a las soluciones disponibles en exteriores, donde el GPS y la cartografía digital han permitido consolidar herramientas de navegación ampliamente accesibles. En interiores, los planos existentes suelen ser insuficientes, poco accesibles, no interoperables o rápidamente obsoletos frente a cambios en la configuración del espacio.

En este contexto, la autonomía de desplazamiento en interiores se ve comprometida. La falta de información espacial confiable no solo incrementa la dependencia de apoyo humano, sino que también aumenta los tiempos de traslado y la probabilidad de desorientación o incidentes. Por ello, desde un enfoque de accesibilidad, el problema no se resuelve únicamente con la disponibilidad de mapas, sino con la implementación de mecanismos capaces de estimar de forma consistente la localización y la orientación del usuario dentro del entorno. Esta capacidad es esencial para traducir la información espacial en instrucciones de navegación comprensibles y accionables, que faciliten un desplazamiento seguro, autónomo y eficiente.

b. Problema científico y tecnológico

Si bien los sistemas de SLAM han demostrado ser herramientas eficaces para la reconstrucción de entornos interiores mediante cámaras RGB-D, cámaras monoculares o sensores LiDAR, su adopción como base de una solución reproducible, verificable y funcional para re-localización visual plantea desafíos científicos y tecnológicos significativos. En la práctica, la calidad del mapa generado y la estabilidad de la localización no dependen únicamente del algoritmo SLAM empleado, sino de la interacción de múltiples variables: condiciones de captura, trayectoria recorrida, velocidad de desplazamiento, oclusiones parciales, calibración y sincronización de sensores, ruido de medición, variaciones de iluminación, superficies con baja textura, presencia de objetos dinámicos y parámetros asociados a la odometría, el cierre de ciclo y la optimización del grafo. Como consecuencia, recorridos distintos dentro de un mismo entorno pueden producir mapas con diferencias en consistencia geométrica, escala, alineación, densidad visual y conectividad topológica, lo que compromete directamente la precisión de la re-localización y limita la confiabilidad del sistema en escenarios reales.

Aun cuando una herramienta como RTAB-Map permite construir mapas basados en grafos, detectar cierres de ciclo y asociar imágenes con poses estimadas, la disponibilidad de una reconstrucción tridimensional o de un grafo de poses no garantiza por sí sola una solución operativa para navegación, orientación o asistencia. Para que la salida de un sistema SLAM pueda ser utilizada por una aplicación externa, es necesario transformar sus resultados en una representación estructurada, consultable e interpretable. Esto implica resolver de manera integrada varias tareas: construir un mapa base consistente del entorno, extraer y organizar los nodos visuales relevantes, asociar cada nodo con una pose en un sistema de referencia estable, re-localizar nuevas observaciones visuales dentro del mapa y proyectar la pose estimada hacia una representación comprensible para el usuario o para otros sistemas de navegación.

El problema se vuelve más complejo en entornos interiores donde suelen existir pasillos repetitivos y escenas visualmente ambiguas “fenómeno conocido como aliasing perceptual”, que dificultan la distinción entre nodos geográficamente distintos pero visualmente similares. En estos casos, los métodos clásicos de

comparación visual como ORB pueden ser eficientes pero sensibles a variaciones de apariencia y perspectiva; por otro lado, los métodos basados en aprendizaje profundo ofrecen mayor robustez ante cambios visuales, pero requieren mecanismos de integración, indexación, validación geométrica y control de confianza para ser utilizados de forma confiable dentro de un flujo de localización reproducible. Por tanto, el reto no consiste únicamente en generar un mapa, sino en diseñar una arquitectura capaz de buscar, comparar, verificar y seleccionar la posición más probable de una imagen de consulta dentro de un mapa previamente construido.

En este contexto, RTAB-Map se adopta en el presente trabajo como motor base para la construcción del mapa y la generación del grafo de poses, debido a su arquitectura basada en nodos, su capacidad de cierre de ciclo y su compatibilidad con distintas modalidades sensoriales. Sin embargo, el problema científico-tecnológico abordado no se reduce a la ejecución de RTAB-Map, sino a la integración de una capa complementaria de re-localización visual implementada en Python, capaz de automatizar el flujo completo de trabajo: extracción de imágenes y poses desde el mapa, construcción de índices de búsqueda, generación de representaciones visuales mediante métodos clásicos y profundos, búsqueda eficiente de candidatos, verificación geométrica de correspondencias, asignación de puntajes de confianza y exportación de resultados verificables.

De esta manera, el desafío central de este trabajo consiste en diseñar, implementar y validar un módulo reproducible de re-localización visual indoor que combine RTAB-Map con técnicas de visión por computadora clásica y aprendizaje profundo, permitiendo estimar la pose de una observación nueva dentro de un mapa previamente construido. La solución debe ser capaz de operar sobre datos extraídos del sistema SLAM, organizar la información visual y geométrica en estructuras consultables, comparar imágenes de consulta contra nodos del mapa y retornar una hipótesis de ubicación con evidencia cuantitativa, como similitud visual, número de correspondencias, inliers geométricos y nivel de confianza.

c. Alcance del problema a resolver

Con base en lo expuesto, el alcance del presente trabajo se delimita en el diseño, implementación y validación de un flujo técnico reproducible que permita transformar datos capturados en entornos interiores mediante sensores de profundidad en representaciones espaciales útiles para localización y navegación. En este marco, el reto no se limita a la generación de mapas, sino que abarca la integración de un proceso completo capaz de asegurar consistencia geométrica, estabilidad en la localización y utilidad operativa de las salidas producidas.

De manera específica, el trabajo se orienta a: generar mapas interiores consistentes en representaciones 2D y 3D a partir de recorridos capturados con sensores de profundidad; mejorar la coherencia del mapa y reducir los efectos de la deriva mediante estrategias de configuración, cierre de ciclo y optimización del grafo; ejecutar procesos de re-localización sobre el mapa previamente construido a partir de nuevas observaciones; extraer y transformar la pose, entendida como posición y orientación, hacia un sistema de referencia estable que permita su proyección en un plano bidimensional; y, finalmente, producir salidas exportables, verificables e interpretables, como visualizaciones 2D de localización y orientación, que sirvan como base para futuras herramientas de orientación accesible.

El alcance incluye también la evaluación de estrategias de búsqueda vectorial de imágenes visualmente similares mediante índices exactos o aproximados, la verificación geométrica de correspondencias mediante técnicas robustas como RANSAC, y el cálculo de métricas de confianza que permitan seleccionar la hipótesis de localización más probable. Como resultado, el sistema debe ser capaz de recibir una imagen de consulta, compararla contra las observaciones previamente indexadas del mapa, identificar los candidatos visualmente más cercanos, validar geométricamente las correspondencias y devolver una estimación de pose compuesta por posición, orientación y nivel de confianza.

1.4. Objetivos

En este capítulo se presentan los objetivos que orientaron el desarrollo del presente Trabajo de Obtención de Grado. Se expone el objetivo general, que consiste en diseñar e implementar una metodología computacional para la relocalización visual en entornos interiores integrando mapas generados con RTAB-Map, técnicas clásicas de visión por computadora y métodos de aprendizaje profundo. A partir de este propósito central, se desglosan los objetivos específicos, los cuales abarcan la definición de requerimientos del sistema, el diseño de la arquitectura híbrida, la construcción del mapa de referencia, la implementación del pipeline de preparación offline, el desarrollo del módulo de relocalización online, la exposición de resultados mediante una API REST, la evaluación cuantitativa del desempeño y la documentación del sistema para su reproducibilidad. Finalmente, se describe la novedad científica y tecnológica del trabajo, centrada en cerrar la brecha entre las capacidades de mapeo de RTAB-Map y su aplicación práctica como base para soluciones de asistencia a la navegación indoor.

a. Objetivo General

Desarrollar e implementar una metodología computacional automatizada, reproducible y evaluable para la relocalización visual en entornos interiores, integrada con los datos obtenidos de la base de datos de mapa RTAB-Map y herramientas desarrolladas en Python, que combine técnicas clásicas de visión por computadora y métodos de aprendizaje profundo para estimar de manera robusta la posición y orientación (pose) de un usuario, dispositivo o agente móvil a partir de imágenes de consulta. La metodología deberá permitir la construcción y consulta de índices visuales, la verificación geométrica de coincidencias, la generación de métricas de desempeño y la producción de salidas visuales y exportables en formatos 2D y 3D, con el propósito de constituir una base funcional para futuras aplicaciones de asistencia a la navegación indoor, con especial énfasis en accesibilidad y apoyo a personas con discapacidad visual.

b. Objetivos Específicos:

- i. Definir los requerimientos funcionales, no funcionales y de datos para un sistema de relocalización visual en entornos interiores, considerando las necesidades de una solución orientada a navegación asistida, así como criterios de efectividad, eficiencia, robustez, interpretabilidad y facilidad de integración con aplicaciones externas.

- ii. Diseñar una arquitectura híbrida de relocalización visual indoor basada en RTAB-Map y módulos desarrollados en Python, integrando las etapas de construcción del mapa, extracción de imágenes y poses, generación de índices de búsqueda, recuperación de candidatos, verificación geométrica, estimación de pose y entrega estructurada de resultados.
- iii. Construir un mapa indoor de referencia mediante RTAB-Map a partir de recorridos capturados con sensores RGB-D, incorporando procesos de extracción de nodos visuales, asociación de poses y organización del dataset necesario para la etapa de relocalización.
- iv. Implementar un pipeline de preparación offline que permita procesar la información generada por RTAB-Map, incluyendo la lectura de poses, asociación imagen-pose, generación de archivos índice, extracción de embeddings visuales y construcción de estructuras de búsqueda eficiente mediante FAISS.
- v. Desarrollar un módulo de relocalización online que, a partir de una imagen de consulta, recupere candidatos visualmente similares mediante técnicas de búsqueda profunda, aplique métodos clásicos o aprendidos de matching, realice verificación geométrica mediante RANSAC y estime la posición y orientación más probable dentro del mapa.
- vi. Implementar un mecanismo de consulta y consumo externo de resultados mediante una API REST, que retorne la pose estimada, nivel de confianza, número de candidatos evaluados y evidencia visual estructurada, facilitando su integración futura en aplicaciones de asistencia a la navegación en interiores.
- vii. Evaluar el desempeño de la metodología propuesta mediante métricas cuantitativas, considerando precisión Top-1 y Top-5, número de inliers geométricos, confianza de la predicción, error de orientación, latencia del sistema y comparativa de desempeño entre matchers sobre consultas reales y sintéticas.
- viii. Documentar los lineamientos de implementación, configuración y reproducibilidad del sistema, incluyendo estructura del proyecto, parámetros críticos, procedimiento de captura, construcción de índices, ejecución del pipeline, limitaciones identificadas y recomendaciones para su adaptación a distintos edificios o escenarios de uso.

c. Novedad científica, tecnológica o aportación

La aportación principal de este Trabajo de Obtención de Grado consiste en integrar las capacidades de mapeo y relocalización de RTAB-Map con un módulo desarrollado en Python, basado en aprendizaje profundo, para estimar la posición y orientación de un usuario o dispositivo en entornos interiores. La novedad no está en crear un nuevo algoritmo de SLAM, sino en construir un flujo de trabajo modular, reproducible y evaluable que permita transformar un mapa visual en una herramienta útil para aplicaciones de asistencia, particularmente orientadas a personas con discapacidad visual.

El sistema combina métodos clásicos de visión por computadora, como ORB, GFTT, ratio test y RANSAC, con técnicas de aprendizaje profundo y recuperación visual mediante embeddings e índices FAISS. Esta arquitectura híbrida busca mejorar la robustez de la relocalización ante cambios de iluminación, perspectiva y apariencia visual, manteniendo al mismo tiempo interpretabilidad y eficiencia computacional.

Además, el trabajo aporta una metodología estandarizada para gestionar imágenes, poses, nodos, configuraciones y resultados derivados de RTAB-Map, generando salidas estructuradas que pueden ser

evaluadas mediante métricas como precisión Top-K, número de correspondencias válidas, inliers, confianza, error de posición, error de orientación y latencia.

En conjunto, el valor tecnológico del proyecto radica en cerrar la brecha entre los sistemas de mapeo visual indoor y su posible aplicación práctica en soluciones de accesibilidad, ofreciendo una base técnica extensible para futuros sistemas de orientación, guiado o asistencia contextual en espacios interiores.

I) Aportación científica

La aportación científica de este trabajo radica en la formalización de un procedimiento metodológico para la relocalización visual en entornos interiores a partir de un mapa de referencia generado con RTAB-Map. El procedimiento propuesto integra la extracción de imágenes y poses del mapa, la construcción de índices de búsqueda, la comparación visual de imágenes de consulta y la estimación de una pose asociada dentro de un sistema de coordenadas conocido. Con ello, se establece una relación clara entre los nodos del grafo de RTAB-Map, sus poses asociadas y la interpretación espacial de la localización obtenida, lo cual permite analizar el problema de relocalización indoor de manera estructurada, reproducible y evaluable.

Asimismo, este trabajo aporta una estrategia de análisis basada en la combinación de métodos clásicos de visión por computadora y técnicas de aprendizaje profundo. En lugar de depender exclusivamente de un único enfoque, se propone un esquema híbrido que utiliza recuperación visual mediante embeddings profundos para seleccionar candidatos relevantes, y verificación geométrica basada en correspondencias locales y RANSAC para validar la consistencia espacial de dichos candidatos. Esta combinación permite estudiar la relocalización indoor considerando tanto la similitud visual global como la coherencia geométrica local, lo que resulta pertinente frente a condiciones comunes en interiores, tales como cambios de iluminación, variaciones de perspectiva, objetos dinámicos, repetición estructural y ambigüedad visual.

De igual forma, la investigación contribuye con la definición de criterios de evaluación orientados a medir el desempeño del sistema de relocalización. Dichos criterios incluyen métricas como exactitud Top-1 y Top-5, número de correspondencias válidas, número de inliers, razón de inliers, confianza de la estimación, error de posición, error de orientación y latencia de procesamiento. Esta propuesta de evaluación permite valorar no solo si el sistema identifica correctamente una ubicación, sino también la calidad, estabilidad y confiabilidad de la estimación obtenida.

El trabajo establece un vínculo conceptual entre la relocalización visual indoor y su posible aplicación en sistemas de orientación asistida. Al relacionar variables técnicas como robustez, estabilidad, latencia, precisión espacial e interpretabilidad de la salida con necesidades de navegación en espacios cerrados, se aporta una base metodológica para el diseño posterior de soluciones orientadas a accesibilidad. En este sentido, la contribución científica no se limita a la implementación de un sistema computacional, sino que organiza el problema de relocalización indoor como un componente evaluable dentro de aplicaciones de navegación asistida.

II) Aportación tecnológica

La aportación tecnológica de este trabajo consiste en el desarrollo de un módulo computacional en Python, un módulo computacional de relocalización visual basado en aprendizaje profundo, orientado a extender las capacidades de relocalización de RTAB-Map mediante una arquitectura modular, configurable e integrable con otros sistemas. El módulo implementa un flujo de trabajo dividido en dos etapas principales: una fase offline de preparación del mapa y construcción de índices, y una fase online de localización a partir de imágenes de consulta. Esta separación permite reutilizar la información previamente procesada del mapa y reducir el costo computacional durante la operación del sistema.

En la fase offline, el sistema automatiza la extracción y organización de imágenes asociadas a nodos del mapa, el procesamiento de poses provenientes de RTAB-Map, la generación de archivos estructurados como `node_index.csv`, el cálculo de embeddings visuales y la construcción de índices de búsqueda mediante FAISS o, en su defecto, mediante un mecanismo de respaldo basado en NumPy. En la fase online, el sistema procesa una imagen de consulta, obtiene candidatos visualmente similares mediante búsqueda exacta con FAISS IndexFlatIP para el mapa evaluado, verifica geométricamente las correspondencias y entrega como resultado una hipótesis de localización con pose, nivel de confianza y candidatos alternativos.

Otra aportación tecnológica relevante es la integración de múltiples estrategias de comparación visual dentro de una misma arquitectura. El sistema contempla métodos clásicos como ORB y GFTT+ORB, así como métodos basados en aprendizaje profundo, tales como SuperPoint, LightGlue, ResNet50 o CLIP para la generación de embeddings. Esta flexibilidad permite comparar configuraciones, adaptar el sistema a distintos recursos computacionales y seleccionar estrategias de localización según las características del entorno o los requerimientos de la aplicación.

Además, el trabajo aporta una estructura de configuración centralizada mediante archivos YAML, lo que facilita modificar parámetros de rutas, métodos de matching, umbrales geométricos, número de candidatos, tipo de índice FAISS, modelo de embeddings y dispositivo de ejecución. Esta característica favorece la reproducibilidad, la experimentación controlada y la adaptación del sistema a diferentes conjuntos de datos o mapas indoor.

Finalmente, el módulo contempla salidas estructuradas y mecanismos de integración, incluyendo resultados en formatos consumibles por otros sistemas, visualizaciones de correspondencias y una API REST para su uso desde aplicaciones externas. Esto permite que la relocalización estimada pueda incorporarse posteriormente en interfaces web, móviles o sistemas de asistencia, sin requerir que toda la lógica de RTAB-Map o ROS esté presente en la capa de visualización. Por ello, la aportación tecnológica del trabajo se centra en una solución modular, portable, extensible y preparada para integrarse en prototipos funcionales de navegación indoor.

III) Aportación social

La aportación social de este trabajo consiste en sentar una base tecnológica para el desarrollo de soluciones de orientación y navegación asistida en espacios interiores. La posibilidad de estimar la localización y

orientación de una persona, robot o dispositivo dentro de un edificio permite avanzar hacia sistemas capaces de transformar una posición y orientación (pose) en información útil para la toma de decisiones, la guía de desplazamiento y la interacción con el entorno.

En particular, el enfoque propuesto puede contribuir al diseño de herramientas de apoyo para personas con discapacidad visual o con necesidades específicas de movilidad en interiores. Aunque el alcance principal del trabajo se centra en la relocalización visual y no en el despliegue completo de una aplicación asistiva final, la obtención de una pose confiable representa un componente fundamental para generar indicaciones accesibles, advertencias contextuales o rutas guiadas dentro de edificios, oficinas, laboratorios, hospitales, escuelas u otros espacios cerrados.

Asimismo, la naturaleza modular, documentada y reproducible del sistema incrementa su potencial de transferencia a distintos contextos institucionales. Al estar basado en herramientas configurables y en una arquitectura abierta, el trabajo reduce la dependencia de soluciones propietarias y facilita que futuros desarrollos puedan adaptarlo a diferentes mapas, condiciones de operación o necesidades de usuarios. Esto favorece la continuidad del proyecto y su posible aprovechamiento en escenarios educativos, administrativos, de investigación o de servicio.

2. ESTADO DEL ARTE O DE LA TÉCNICA

A continuación, se presentan trabajos y enfoques relevantes organizados en tres grupos progresivos. El primero aborda los sistemas SLAM visuales y visual-inerciales: se revisan RTAB-Map en su formulación original de cierre de ciclo por apariencia, ORB-SLAM3 como referencia en SLAM visual multi-sensor, VINS-Fusion como solución de fusión visual-inercial robusta, y MAST3R-SLAM como ejemplo de SLAM denso monocular basado en modelos aprendidos. Este grupo establece el contexto técnico del mapa que sirve como entrada al sistema desarrollado. El segundo grupo examina RTAB-Map en escenarios reales y su evolución como librería integral de SLAM, lo que fundamenta su adopción como motor de mapeo en el presente trabajo. El tercer grupo se enfoca en técnicas de recuperación visual y localización indoor: NetVLAD como referencia en reconocimiento visual de lugares mediante embeddings aprendidos, HF-Net como propuesta de localización jerárquica coarse-to-fine, InLoc como sistema específico para interiores con matching denso y verificación por síntesis de vistas, y SuperGlue como método de emparejamiento profundo de puntos entre imágenes.

Adicionalmente, se incluye una revisión del área de SLAM orientado a personas con discapacidad visual, que sitúa el problema de la navegación asistida indoor como motivación aplicada del trabajo.

Esta revisión permite ubicar el presente Trabajo de Obtención de Grado dentro de un marco más amplio: el desarrollo de un módulo de relocalización visual indoor que aprovecha mapas generados con RTAB-Map y combina embeddings profundos con ResNet para recuperación eficiente mediante FAISS, matching con ORB o SuperPoint+LightGlue, y verificación geométrica con RANSAC, para estimar y visualizar la posición y orientación de un usuario o dispositivo en un entorno previamente mapeado.

2.1. RTAB-Map: Cierre de ciclo por apariencia

El cierre de ciclo constituye un proceso fundamental en los sistemas SLAM, ya que permite identificar cuándo el sistema retorna a un lugar previamente visitado. Esta capacidad es esencial para reducir el error acumulado en la estimación de la trayectoria y mejorar la consistencia global del mapa. Sin embargo, a medida que el entorno modelado crece, la comparación de la observación actual con todas las vistas almacenadas incrementa considerablemente el costo computacional, lo que compromete la operación en tiempo real. En este contexto, RTAB-Map (Real-Time Appearance-Based Mapping) plantea un enfoque orientado al cierre de ciclo por apariencia, diseñado específicamente para escenarios de gran escala y operación prolongada.

La contribución central de este trabajo radica en un mecanismo de gestión de memoria que limita de forma eficiente el conjunto de lugares utilizados en la detección de cierres de ciclo. Para ello, mantiene en una memoria de trabajo las ubicaciones recientes o frecuentemente observadas, mientras transfiere el resto a una memoria de largo plazo. De este modo, el sistema concentra la comparación en la información más relevante para la detección inmediata y reactiva recuerdos antiguos únicamente cuando existe evidencia de un posible reencuentro. Integrado en un marco bayesiano, este esquema permite preservar restricciones de tiempo real sin sacrificar la capacidad de reconocer lugares previamente visitados, incluso en mapas extensos. Por ello, RTAB-Map se consolidó como una propuesta relevante para aplicaciones de localización y mapeo en interiores, donde la escalabilidad y la robustez del cierre de ciclo son requisitos críticos. [5]

No obstante, el mecanismo de cierre de ciclo de RTAB-Map está concebido como parte del proceso de mapeo continuo, acoplado al movimiento del robot durante la construcción del mapa. Esto implica que no está diseñado para recibir imágenes de consulta externas una vez que el mapa ha sido construido, ni para operar como módulo de relocalización independiente accesible desde aplicaciones externas. Su representación visual se basa en descriptores locales agregados mediante modelos de bolsa de palabras visuales, lo que puede resultar menos discriminativo que los embeddings globales profundos en entornos con pasillos repetitivos o zonas visualmente ambiguas. Adicionalmente, la gestión de memoria en modo trabajo/largo plazo, aunque eficiente para SLAM en línea, no garantiza que todos los nodos del mapa estén disponibles de forma equitativa para una búsqueda exhaustiva en tiempo de consulta. Finalmente, RTAB-Map no expone un mecanismo estructurado de confianza que combine similitud visual, correspondencias geométricas e inliers verificados, ni genera visualizaciones de la localización proyectadas sobre un plano 2D. Estas limitaciones motivan el desarrollo del módulo complementario propuesto en el presente trabajo, el cual reutiliza el mapa generado por RTAB-Map para construir un pipeline de relocalización visual externo, basado en embeddings profundos, búsqueda eficiente mediante FAISS, verificación geométrica con RANSAC y proyección sobre plano 2D.

2.2. SLAM for Visually Impaired People

Este trabajo presenta una revisión sistemática del uso de técnicas de Simultaneous Localization and Mapping (SLAM) en tecnologías de asistencia dirigidas a personas ciegas o con discapacidad visual. En lugar de proponer un único sistema, el estudio examina de manera comparativa 54 investigaciones recientes, publicadas principalmente a partir de 2017, con el propósito de identificar los enfoques de localización y mapeo más empleados en este ámbito. Asimismo, analiza los tipos de sensores utilizados, los requerimientos computacionales y la incorporación de métodos de aprendizaje automático en las distintas soluciones desarrolladas.

También evalúa las principales fortalezas y limitaciones de estas propuestas en contextos reales de navegación asistida. Entre sus aportaciones más relevantes se encuentra la identificación de desafíos aún no resueltos, como la operación en entornos dinámicos y complejos, la robustez ante cambios del escenario, la usabilidad de los sistemas y su adopción efectiva por parte de los usuarios. En este sentido, el estudio constituye una base de referencia para comprender el estado actual del área y para orientar futuras investigaciones en el diseño de herramientas de navegación más seguras, accesibles y funcionales para personas con discapacidad visual. [6]

Sin embargo, al tratarse de una revisión sistemática, el trabajo no propone ni implementa un sistema concreto, lo que limita su utilidad directa como referencia técnica para el desarrollo de soluciones específicas. Los 10 sistemas analizados están orientados, en su mayoría, a guiar al usuario durante el desplazamiento en tiempo real, sin contemplar el escenario de relocalización a partir de una imagen de consulta capturada de forma independiente sobre un mapa previamente construido. Asimismo, ninguno de los enfoques revisados reporta una integración modular que combine recuperación visual mediante embeddings profundos, búsqueda eficiente sobre un índice vectorial, verificación geométrica de correspondencias y proyección de la pose estimada sobre un plano 2D accesible desde una API externa. La usabilidad y la adopción efectiva son identificadas como desafíos abiertos, pero no se ofrece una arquitectura de referencia que los aborde. Estas brechas refuerzan la pertinencia del presente trabajo, que propone un módulo de relocalización visual funcional, consultable mediante una API REST y con salida visual interpretable, orientado a constituir una base técnica para futuras aplicaciones de asistencia a la navegación indoor para personas con discapacidad visual.

2.3. MAST3R-SLAM: SLAM denso monocular 3D

MASt3R-SLAM propone un sistema de SLAM denso monocular 3D que emplea una única cámara y se apoya en un modelo aprendido, MAST3R, para fortalecer la reconstrucción de la escena y la obtención de correspondencias tridimensionales entre múltiples vistas. A diferencia de los enfoques que dependen exclusivamente de características visuales tradicionales, este método incorpora un conocimiento previo aprendido que mejora la robustez del seguimiento en secuencias reales, incluso bajo condiciones complejas. El sistema integra un pipeline completo que abarca el seguimiento de la cámara, la fusión local, la construcción del grafo, el cierre de ciclo y la optimización global. Como resultado, reporta desempeño en tiempo real y una estimación de poses globalmente consistente, lo que lo posiciona como una alternativa relevante para aplicaciones de localización y mapeo en entornos interiores. [7]

No obstante, MAST3R-SLAM está concebido como un sistema de mapeo y seguimiento continuo, donde la localización ocurre de forma acoplada al movimiento de la cámara durante la construcción del mapa. Este diseño no contempla el escenario de relocalización discreta, en el que una imagen de consulta independiente debe localizarse contra un mapa previamente construido y almacenado. La reconstrucción densa tridimensional, aunque precisa, implica un costo computacional considerablemente mayor al requerido para recuperación visual por similitud, lo que puede resultar excesivo cuando el objetivo se reduce a estimar la posición y orientación de una imagen de consulta sobre un plano 2D. Adicionalmente, el sistema depende de un modelo aprendido de gran escala que requiere hardware especializado y no expone un mecanismo de consulta externo mediante API.

2.4. ORB-SLAM3: SLAM visual

ORB-SLAM3 es una de las librerías de referencia en SLAM visual para la estimación de la posición y orientación (pose) de un sistema en tiempo real. Su relevancia radica en que admite múltiples configuraciones sensoriales, entre ellas cámaras monoculares, estéreo, RGB-D y esquemas visual-inerciales que combinan cámara e IMU. Uno de sus aportes más significativos es la integración estrecha de la información visual e inercial, lo que mejora la precisión y robustez de la estimación, especialmente en escenarios con movimientos rápidos o con baja textura visual. Asimismo, incorpora una arquitectura de multi-mapa, mediante la cual el sistema puede iniciar un nuevo mapa cuando pierde el seguimiento y, posteriormente, fusionarlo con mapas previos al reconocer zonas ya visitadas. Esta capacidad favorece la relocalización y la corrección de errores acumulados, lo que le permite mantener un desempeño robusto tanto en entornos interiores como exteriores y lo convierte en una base sólida para aplicaciones de navegación y localización en espacios indoor [8].

Sin embargo, ORB-SLAM3 está diseñado para operar de forma continua y acoplada al flujo de la cámara: su mecanismo de relocalización actúa para recuperar el seguimiento dentro de una sesión SLAM en curso, no para recibir imágenes de consulta arbitrarias contra un mapa ya construido y almacenado. El reconocimiento de lugares se basa en un modelo de bolsa de palabras visuales construido con descriptores ORB, lo que puede resultar menos discriminativo que los embeddings globales profundos en entornos interiores con pasillos repetitivos o escenas visualmente ambiguas. Adicionalmente, el sistema no incorpora una etapa de recuperación basada en aprendizaje profundo ni expone una interfaz de consulta externa mediante API, lo que dificulta su uso como servicio de localización independiente integrable en aplicaciones de terceros.

2.5. VINS-Fusion: estimación de estado visual-inercial

En entornos interiores, el desempeño de los sistemas de localización basados únicamente en visión puede verse comprometido por factores como el desenfoque, la escasez de textura en las superficies y las variaciones en las condiciones de iluminación. En este contexto, VINS-Fusion se plantea como una solución orientada a la estimación precisa de la posición y la orientación mediante fusión visual-inercial, con el propósito de incrementar la estabilidad y confiabilidad del sistema en escenarios reales. Sus autores lo describen como un optimization-based multi-sensor state estimator, es decir, un estimador de estado multisensor basado en optimización. A diferencia de los métodos que actualizan la trayectoria de forma secuencial y acumulativa, VINS-Fusion formula el problema como un proceso de optimización sobre las observaciones disponibles, lo que permite refinar de manera conjunta la trayectoria estimada. En términos prácticos, esta estrategia aprovecha la complementariedad entre sensores: cuando la información visual se degrada, la IMU (Unidad de Medición Inercial) contribuye a mantener el seguimiento; y cuando la IMU comienza a acumular deriva, la cámara aporta información que permite corregirla. Esta integración mejora la robustez del sistema frente a las limitaciones propias de los entornos interiores. [9]

VINS-Fusion presenta limitaciones relevantes cuando se considera su aplicación en un escenario de relocalización visual basada en consultas discretas. Al depender de la presencia de una IMU calibrada y sincronizada con la cámara, el sistema introduce una restricción de hardware que eleva el costo de despliegue y la complejidad de configuración, en contraste con enfoques que operan exclusivamente sobre imágenes RGB. Asimismo, su diseño está orientado a la estimación continua del estado durante el movimiento del agente, por lo que no contempla de forma directa el problema de localizar una imagen de consulta aislada contra un mapa previamente construido y almacenado. La fusión visual-inercial, aunque robusta para seguimiento continuo, no incorpora mecanismos de recuperación visual global “como índices de embeddings o modelos de reconocimiento de lugares” que permitan resolver la ambigüedad posicional cuando el sistema se reinicia o recibe una observación sin contexto temporal previo. Para escenarios de relocalización donde la entrada es una imagen de consulta independiente y el objetivo es identificar su correspondencia dentro de un mapa de referencia, resulta más adecuado un enfoque basado en recuperación por similitud visual y verificación geométrica, sin dependencia de sensores inerciales adicionales.

2.6. RTAB-Map para Escenarios Reales

RTAB-Map se consolida como una biblioteca open source orientada al mapeo y la localización en escenarios reales, particularmente en contextos donde el entorno puede crecer de manera considerable y exige operación en línea a gran escala y durante periodos prolongados. Originalmente concebido como un enfoque de detección de cierre de ciclo basado en apariencia y gestión de memoria, el sistema evolucionó hasta convertirse en una librería integral de SLAM, capaz de operar sobre diferentes plataformas robóticas y configuraciones sensoriales.

De acuerdo con Labbé y Michaud, una de las principales extensiones de RTAB-Map consiste en incorporar, dentro de un mismo paquete, soporte tanto para SLAM visual como para SLAM basado en LiDAR, con el propósito de facilitar la implementación y la comparación de soluciones en distintos robots, sensores y aplicaciones. Esta integración resulta especialmente relevante en escenarios reales, donde la selección del enfoque más adecuado depende de variables como el costo, la precisión, la capacidad de cómputo y la facilidad de integración.

La contribución central del trabajo radica en que RTAB-Map deja de ser únicamente un método de cierre de ciclo para convertirse en una herramienta completa de SLAM, compatible con sensores visuales “como cámaras RGB-D y sistemas estéreo”, sensores LiDAR y combinaciones sensoriales de uso frecuente en navegación autónoma. Asimismo, la librería permite evaluar distintas configuraciones de mapeo y localización en 2D y 3D dentro de una misma plataforma, lo que aporta flexibilidad metodológica y valor práctico al momento de definir la solución más conveniente para un caso de uso específico.

RTAB-Map resulta particularmente relevante porque ofrece una base tecnológica madura para desarrollar sistemas de localización y orientación en interiores a partir de mapas previamente generados o actualizados en operación. Su capacidad para integrar múltiples sensores, comparar configuraciones y operar en condiciones cercanas a escenarios reales lo convierte en una referencia técnica útil para propuestas orientadas a la asistencia de usuarios en entornos indoor, incluyendo aplicaciones dirigidas a personas que desconocen el lugar o a personas con discapacidad visual. [10]

Como librería integral de SLAM, RTAB-Map mantiene la localización acoplada al proceso de mapeo activo, lo que implica que su uso como servicio de relocalización independiente “capaz de recibir imágenes de consulta externas sobre un mapa ya construido” requiere adaptaciones que no forman parte de su diseño original. El reconocimiento de lugares se basa en descriptores locales agregados mediante modelos de bolsa de palabras visuales, sin incorporar representaciones globales aprendidas que podrían ofrecer mayor robustez ante escenas visualmente ambiguas. Asimismo, RTAB-Map no expone de forma nativa una interfaz de consulta ligera y desacoplada “como una API REST” que permita integrar sus capacidades de localización en aplicaciones externas sin necesidad de ejecutar el pipeline completo de SLAM. Finalmente, aunque la librería genera mapas 2D y 3D, no proporciona un mecanismo directo para proyectar la pose estimada de una imagen de consulta sobre un plano 2D personalizado con visualización de orientación. Estas características motivan el diseño de un módulo complementario que, apoyándose en el mapa generado por RTAB-Map, incorpore recuperación visual profunda, verificación geométrica y una capa de servicio accesible desde aplicaciones externas.

2.7. NetVLAD: Visual Place Recognition

NetVLAD aborda el problema del reconocimiento visual de lugares (Visual Place Recognition, VPR), cuya finalidad es identificar de manera rápida y precisa la ubicación asociada a una imagen de consulta mediante su comparación con una base de datos de imágenes georreferenciadas o previamente registradas del mismo entorno. Este problema resulta especialmente relevante en escenarios de localización visual a gran escala, donde la eficiencia y la capacidad de discriminación del descriptor son factores determinantes. Antes de esta propuesta, era habitual emplear representaciones basadas en bolsas de palabras visuales (Bag of Visual Words) o en descriptores agregados manualmente; sin embargo, estos enfoques presentaban limitaciones de robustez y precisión frente a cambios de apariencia, variaciones de iluminación, o similitudes estructurales entre distintos lugares.

La principal contribución de NetVLAD consiste en una arquitectura de red neuronal convolucional diseñada específicamente para reconocimiento de lugares y entrenable de extremo a extremo. Su componente central es una capa de agregación denominada NetVLAD, inspirada en el descriptor clásico VLAD (Vector of Locally Aggregated Descriptors), pero reformulada para ser diferenciable e integrarse dentro de una CNN mediante retropropagación. En términos funcionales, el proceso parte de una imagen de entrada (un pasillo, una entrada o un corredor), a partir de la cual la CNN extrae características locales de alto nivel. Posteriormente, la capa NetVLAD agrega dichas características y las transforma en un vector global compacto, que actúa como una firma numérica representativa de toda la imagen. Durante la etapa de inferencia, este vector se compara con los vectores globales de las imágenes almacenadas en el mapa o base de datos, de modo que la localización se estima a partir de las coincidencias más cercanas en el espacio de características.

Un aspecto particularmente relevante del trabajo es que el modelo no fue concebido únicamente como una CNN genérica para extracción de características, sino como una arquitectura optimizada específicamente para VPR. Para ello, los autores proponen un esquema de entrenamiento débilmente supervisado, sustentado en una función de pérdida de ranking, con el objetivo de acercar en el espacio descriptor las imágenes que representan un mismo lugar y alejar aquellas correspondientes a ubicaciones distintas. El entrenamiento se realiza con imágenes del mismo sitio capturadas en diferentes momentos, obtenidas de Google Street View Time Machine, lo que permite al modelo aprender invariancias útiles ante cambios temporales y de apariencia. De acuerdo con los autores, esta estrategia supera tanto a representaciones no aprendidas como a descriptores extraídos directamente de CNN de propósito general, consolidando a NetVLAD como una referencia fundamental en reconocimiento visual de lugares a gran escala. [11]

NetVLAD presenta limitaciones relevantes cuando se considera su integración en un sistema completo de relocalización visual indoor. Al haber sido entrenado con imágenes de exteriores a nivel de calle “obtenidas de Google Street View”, el modelo aprende invariancias propias de escenas urbanas, lo que puede reducir su capacidad de discriminación en entornos interiores con características visuales distintas, como iluminación artificial, superficies planas poco texturizadas y estructuras repetitivas a corta distancia. Adaptar el modelo a un nuevo entorno indoor requeriría reentrenamiento o ajuste fino con datos del dominio objetivo, lo que implica disponer de pares de imágenes anotados geográficamente para ese entorno específico. Adicionalmente, NetVLAD opera exclusivamente como etapa de recuperación global:

identifica las imágenes más similares en la base de datos, pero no incorpora una etapa de verificación geométrica que confirme la consistencia de las correspondencias identificadas, lo que puede producir predicciones incorrectas con alta similitud aparente en zonas visualmente ambiguas. Esta limitación es especialmente crítica en pasillos repetitivos, donde distintos nodos pueden producir vectores similares en el espacio de embeddings. Finalmente, el sistema no genera métricas de confianza compuestas ni proyecciones de la pose estimada sobre un plano 2D, funcionalidades necesarias para constituir una solución de relocalización completa e interpretable. Estas observaciones sugieren que, para un sistema de relocalización indoor funcional, la etapa de recuperación visual global debe complementarse con verificación geométrica y un mecanismo de estimación de confianza que permita seleccionar y validar la hipótesis de localización más probable.

2.8. HF-Net: localización jerárquica

En el proyecto HF-Net, Sarlin et al. abordan el problema de la localización visual jerárquica en mapas previamente construidos, con el objetivo de estimar la posición de una imagen de consulta dentro de un entorno conocido. Los autores señalan que la localización visual a gran escala sigue siendo un desafío, especialmente en escenarios con cambios significativos de apariencia, como variaciones de iluminación, condiciones estacionales, presencia de personas o modificaciones en la disposición del mobiliario. Asimismo, destacan que numerosos métodos de alta precisión resultan demasiado costosos en términos computacionales para aplicaciones en tiempo real.

Para responder a estas limitaciones, HF-Net propone una estrategia jerárquica de tipo coarse-to-fine. En una primera etapa, el sistema realiza una búsqueda global (global retrieval) para identificar las ubicaciones candidatas más probables dentro del mapa. Posteriormente, restringe el análisis a dichas hipótesis y ejecuta un emparejamiento local (local matching), con el fin de refinar la estimación de la pose con mayor precisión. Este esquema reduce de manera importante el costo computacional, ya que evita comparar exhaustivamente la imagen de entrada con todo el mapa y concentra el procesamiento fino únicamente en regiones relevantes.

La principal aportación de HF-Net radica en el uso de una red neuronal convolucional monolítica capaz de predecir simultáneamente descriptores globales y características locales, integrando en una sola arquitectura las dos etapas fundamentales del proceso de localización. Gracias a esta representación jerárquica, el método combina eficiencia y precisión, y resulta adecuado para tareas de localización en tiempo real con seis grados de libertad (6-DoF). De acuerdo con los autores, el uso de descriptores aprendidos también mejora la robustez frente a variaciones severas de apariencia y permite alcanzar resultados de referencia en dos benchmarks exigentes de localización a gran escala. [12]

HF-Net presenta limitaciones relevantes para su adopción en un sistema de relocalización indoor de propósito general. La arquitectura monolítica que produce simultáneamente descriptores globales y características locales requiere un proceso de entrenamiento especializado con grandes volúmenes de datos anotados, lo que dificulta su adaptación directa a entornos interiores específicos sin reentrenamiento o ajuste fino. Asimismo, la estimación de pose con seis grados de libertad presupone la disponibilidad de un

mapa 3D denso construido mediante reconstrucción por estructura desde el movimiento (SfM), lo que implica una etapa de preparación del mapa de mayor complejidad que la requerida por sistemas basados en grafos de nodos con poses asociadas. El acoplamiento de las etapas de recuperación global y emparejamiento local dentro de una sola red neuronal, aunque eficiente, reduce la modularidad del sistema: sustituir el componente de recuperación o el de emparejamiento local de forma independiente requeriría modificar o reentrenar la arquitectura completa. Por otro lado, el enfoque no define explícitamente un mecanismo de confianza compuesto que integre evidencia de múltiples fuentes —como similitud visual, número de correspondencias geométricas e inliers verificados— para seleccionar y validar la hipótesis de localización más probable. Estas consideraciones sugieren que, para entornos donde el mapa de referencia es un grafo de nodos con poses asociadas y no una nube de puntos densa, resulta pertinente explorar arquitecturas más modulares que desacoplen la etapa de recuperación visual global de la verificación geométrica, facilitando la adaptación a distintos entornos sin necesidad de reentrenamiento completo del sistema.

2.9. InLoc: localización indoor con matching denso

InLoc fue desarrollado específicamente para el problema de localización visual en interiores y tiene como objetivo estimar la pose de seis grados de libertad (6DoF) de una imagen de consulta con respecto a un mapa 3D indoor. Su metodología se estructura en tres etapas principales: en primer lugar, realiza la recuperación de poses candidatas (retrieval), lo que permite escalar el proceso a entornos interiores de gran tamaño; en segundo lugar, estima la pose mediante matching denso, sustituyendo el uso exclusivo de características locales por correspondencias a nivel de píxel, una decisión especialmente adecuada para escenas interiores con regiones poco texturizadas; y, por último, incorpora una etapa de verificación basada en síntesis de vistas, en la que se renderiza una vista esperada del entorno y se compara con la imagen de consulta para mejorar la robustez frente a cambios de punto de vista, oclusiones, modificaciones en la disposición de la escena y presencia de objetos interpuestos. Adicionalmente, el trabajo introduce un conjunto de datos orientado a condiciones reales de uso, compuesto por imágenes de consulta capturadas con teléfonos móviles distintos y en fechas diferentes a las empleadas para construir el mapa 3D de referencia, con poses 6DoF conocidas. Este diseño experimental fuerza variaciones de iluminación, oclusiones y cambios del entorno, lo que convierte al dataset en una base exigente y representativa para la evaluación de métodos de localización visual en interiores. [13]

InLoc presenta limitaciones significativas que condicionan su viabilidad en escenarios donde la construcción del mapa y la complejidad del pipeline deben mantenerse accesibles. La metodología depende de la disponibilidad de un mapa 3D denso del entorno, generalmente construido con escáneres de alta precisión o mediante reconstrucción densa a partir de múltiples vistas, lo que implica un proceso de captura y procesamiento considerablemente más costoso que el derivado de un recorrido con cámara RGB-D integrado en un sistema SLAM convencional. La etapa de verificación mediante síntesis de vistas requiere renderizar imágenes artificiales del entorno desde poses candidatas, operación que demanda acceso al modelo 3D en tiempo de consulta y capacidad de cómputo adicional para el renderizado, lo que aumenta la latencia del sistema y su dependencia de recursos de hardware especializados. El matching denso a nivel de píxel, aunque robusto en zonas poco texturizadas, es computacionalmente más intensivo que el emparejamiento basado en puntos clave, lo que puede representar un obstáculo para aplicaciones con restricciones de tiempo de respuesta. Adicionalmente, el sistema está orientado a la estimación de pose con seis grados de libertad, nivel de precisión que requiere el mapa 3D como referencia geométrica y que puede resultar innecesario en aplicaciones donde la salida requerida es la posición en un plano 2D y la orientación aproximada del usuario. Para escenarios donde el mapa disponible es un grafo de nodos con poses asociadas “sin reconstrucción densa”, resulta pertinente explorar enfoques que combinen recuperación visual global con verificación geométrica ligera, sin depender de síntesis de vistas ni de mapas tridimensionales densos como requisito operativo.

2.10. SuperGlue: emparejar puntos entre imágenes

SuperGlue plantea un método de aprendizaje profundo para el emparejamiento de puntos entre dos imágenes, orientado a reducir correspondencias incorrectas y a mejorar la estimación de pose. Su principal aporte consiste en modelar el problema de correspondencia de manera global, en lugar de decidir cada emparejamiento de forma aislada. Para ello, emplea una red neuronal sobre grafos con mecanismos de atención que enriquecen la descripción de cada punto a partir de su contexto local y de sus relaciones con los puntos de la otra imagen. En particular, utiliza self-attention para capturar dependencias dentro de cada imagen y cross-attention para razonar sobre las posibles correspondencias entre ambas, generando representaciones más discriminativas e informadas por la estructura de la escena.

Además, SuperGlue incorpora una capa de asignación basada en transporte óptimo diferenciable, resuelta mediante iteraciones de Sinkhorn, con el fin de obtener una correspondencia globalmente consistente entre los conjuntos de puntos. Este planteamiento permite imponer restricciones de asignación y evitar decisiones ambiguas o localmente inconsistentes. De igual manera, el método introduce contenedores de descarte para aquellos puntos que no deben ser emparejados, ya sea porque no tienen correspondencia visible en la otra imagen o porque resultan poco confiables. En conjunto, esta arquitectura supera las heurísticas tradicionales de emparejamiento al aprender regularidades geométricas y estructurales a partir de pares de imágenes, mostrando un desempeño sobresaliente en tareas de estimación de pose y una integración viable en sistemas modernos de SfM y SLAM. [14]

SuperGlue presenta limitaciones relevantes que condicionan su rol dentro de un sistema completo de relocalización visual. Al ser exclusivamente un módulo de emparejamiento local, el método opera sobre pares de imágenes previamente seleccionados y no aborda el problema de recuperación global: no determina cuáles imágenes del mapa son candidatas plausibles para una imagen de consulta, por lo que debe integrarse necesariamente con una etapa de recuperación previa que reduzca el espacio de búsqueda. El uso de mecanismos de atención sobre grafos y la resolución iterativa mediante el algoritmo de Sinkhorn implican un costo computacional considerablemente mayor al de los matchers clásicos, lo que puede representar un obstáculo cuando se requiere verificar múltiples candidatos en condiciones de baja latencia. Adicionalmente, SuperGlue fue diseñado para operar conjuntamente con SuperPoint como detector y descriptor de puntos clave; su aplicación con otros detectores requiere adaptaciones que pueden afectar el desempeño. La complejidad del mecanismo de atención crece cuadráticamente con el número de puntos clave, lo que puede limitar su eficiencia en imágenes con alta densidad de características. Finalmente, el método produce correspondencias con puntajes individuales por par, pero no define un mecanismo de confianza compuesto que integre la evidencia geométrica con la similitud visual global, aspecto necesario para seleccionar de forma fundamentada la hipótesis de localización más probable dentro de un pipeline completo. Estas características refuerzan la idea de que un sistema robusto de relocalización indoor debe combinar una etapa de recuperación eficiente con una etapa de verificación geométrica como SuperGlue, sin depender de este último como componente único de decisión.

La revisión de trabajos relacionados permite identificar que la localización indoor ha sido abordada desde distintas perspectivas: sistemas basados en infraestructura inalámbrica, como Wi-Fi, BLE y beacons; métodos de SLAM visual y visual-inercial; soluciones basadas en mapas topológicos; y enfoques recientes apoyados en aprendizaje profundo. Entre sus principales fortalezas se encuentran la disponibilidad de tecnologías de bajo costo, la posibilidad de construir mapas del entorno, la estimación de pose en tiempo real y el incremento de robustez mediante modelos aprendidos.

Sin embargo, también se observan áreas de oportunidad relevantes. Las soluciones basadas en beacons o Wi-Fi dependen de infraestructura instalada, calibración y mantenimiento. Los sistemas SLAM tradicionales pueden verse afectados por cambios de iluminación, oclusiones, escenas repetitivas o pérdida temporal de seguimiento. Por su parte, los métodos basados únicamente en aprendizaje profundo pueden requerir entrenamiento especializado, mayor capacidad computacional y una integración más compleja con sistemas robóticos existentes.

A partir de esta revisión, se identifica que no existe una integración directa y modular que combine, en un mismo flujo de trabajo, mapas indoor previamente generados, relocalización visual robusta, recuperación eficiente de imágenes candidatas, validación geométrica y visualización de la posición y orientación estimada. En particular, se observa una oportunidad para complementar herramientas como RTAB-Map con métodos modernos de Deep Learning, sin sustituir por completo los métodos clásicos que aportan interpretabilidad y validación geométrica.

El reto principal de este trabajo de obtención de grado consiste en proponer e implementar un módulo híbrido de relocalización visual indoor que permita reutilizar mapas generados con RTAB-Map, estimar la pose de una imagen de consulta y visualizar la localización y orientación del usuario o dispositivo dentro del entorno. Para ello, el proyecto plantea integrar técnicas de extracción de características, embeddings profundos, búsqueda eficiente mediante índices, ranking de candidatos y verificación geométrica, buscando una solución práctica, modular y extensible para escenarios interiores. Esta propuesta se alinea con el objetivo técnico del módulo `dl_relocalization`, el cual combina métodos clásicos, Deep Learning, FAISS y RANSAC para robustecer la estimación de poses en ambientes indoor.

Como se muestra en la Tabla 1 la comparativa se evidencia que la contribución del sistema propuesto no radica en sustituir los métodos existentes de SLAM o reconocimiento visual de lugares, sino en integrar una base de datos de una mapa RTAB-Map previamente construida, a una etapa de recuperación visual global, verificación geométrica configurable y una capa de exposición de resultados. Esta combinación permite utilizar desde una imagen independiente hasta n número de imágenes como consulta sin requerir la ejecución continua del proceso SLAM.

Trabajo	Recuperación global	Verificación local	Mapa requerido	Consulta independiente	API / integración	Diferencia con esta propuesta
RTAB-Map	BoW / apariencia	Sí, dentro del SLAM	Grafo SLAM	Limitada	No REST nativa	Relocalización acoplada al SLAM
ORB-SLAM3	BoW con ORB	Sí	Mapa SLAM	Orientada a recuperar tracking	No REST nativa	No integra recuperación profunda
NetVLAD	Sí, descriptor global	No	Base de imágenes	Sí	No	Requiere verificación geométrica adicional
HF-Net	Sí	Sí	Mapa 3D/SfM	Sí	No	Arquitectura monolítica y menos modular
InLoc	Sí	Matching denso + síntesis	Mapa 3D denso	Sí	No	Mayor costo de preparación y consulta
Propuesta	ResNet50 + FAISS	ORB/SPL + RANSAC	Grafo RTAB-Map	Sí	API REST	Pipeline modular y salida 2D

Tabla 1: Comparación de enfoques de la propuesta de relocalización visual indoor.

3. MARCO TEÓRICO/CONCEPTUAL

Para comprender la propuesta presentada en este trabajo, es necesario establecer un conjunto de conceptos teóricos y técnicos que permitan explicar cómo es posible estimar la posición y la orientación de una persona, robot o dispositivo dentro de un edificio, en escenarios donde la señal GPS no se encuentra disponible o no ofrece la precisión requerida. Esta problemática exige revisar los fundamentos de la localización en interiores, la estimación de pose, el uso de sensores visuales y la construcción de representaciones del entorno que permitan vincular una imagen capturada por una cámara con una ubicación específica dentro de un espacio físico.

A diferencia de los enfoques clásicos de localización visual que comparan exhaustivamente cada imagen de consulta contra la totalidad de las referencias almacenadas, la propuesta desarrollada en este trabajo adopta una arquitectura híbrida en dos etapas. En la primera, se emplea una red neuronal convolucional profunda “específicamente una ResNet50 con la capa de clasificación eliminada” para codificar cada imagen como un vector de representación de 2048 dimensiones en un espacio métrico de alta dimensión, sobre el cual se construye un índice de búsqueda vectorial mediante FAISS, que permite recuperar eficientemente los lugares visualmente más similares. En la segunda etapa, cada candidato recuperado es sometido a una verificación geométrica mediante estimación de homografía con RANSAC, empleando descriptores de puntos de interés “ORB como línea base o SuperPoint con emparejamiento LightGlue como opción de mayor precisión” para confirmar la coherencia espacial de la correspondencia visual. Adicionalmente, el sistema incorpora un módulo de clasificación semántica de escenas basado en el modelo CLIP en modalidad zero-shot, que permite asignar una etiqueta contextual al lugar identificado sin requerir entrenamiento específico sobre el entorno.

En este marco, el presente capítulo expone los fundamentos conceptuales que sustentan cada componente de la arquitectura propuesta, organizados de acuerdo con la estructura que se describe a continuación.

En primer lugar, se aborda la localización en interiores como problema general, sus características distintivas frente a la localización en exteriores y las principales familias de soluciones existentes en la literatura. A continuación, se revisa la estimación de pose en seis grados de libertad “posición (x, y, z) y orientación en cuaternión” y su proyección a una representación planar bidimensional (x, y, yaw), que constituye la salida del sistema propuesto.

Seguidamente, se describe el rol de los sensores visuales empleados en este tipo de sistemas, con énfasis en cámaras RGB y RGB-D, y se presentan los fundamentos de SLAM visual, el cierre de ciclos y la optimización de grafos, que permiten construir y mantener mapas consistentes del entorno. Estos conceptos fundamentan el uso de RTAB-Map como herramienta de construcción del mapa de referencia, cuya base de datos SQLite constituye el único insumo externo del sistema de localización.

Posteriormente, se revisan las formas de representación del espacio interior, desde mapas de ocupación 2D hasta grafos topológicos de poses, necesarias para asociar la estimación de pose con un plano del edificio mediante una transformación afín. A continuación, se presentan en profundidad los fundamentos de la relocalización visual basada en aprendizaje profundo, que constituye el eje central de la propuesta. Esta revisión incluye las redes neuronales convolucionales para la extracción de representaciones visuales globales (ResNet50), los índices de búsqueda vectorial eficiente (FAISS, con las variantes IndexFlatIP e IndexHNSWFlat), los descriptores de características locales clásicos y neuronales (ORB, SuperPoint, LightGlue), la verificación geométrica por homografía (RANSAC) y la clasificación semántica de escenas mediante modelos multimodales (CLIP).

Finalmente, dado que el sistema propuesto puede servir como base para aplicaciones de asistencia a la navegación, se incorporan consideraciones relacionadas con la accesibilidad y la navegación indoor para personas con discapacidad visual, que contextualizan la relevancia práctica del trabajo y vinculan los fundamentos técnicos con una aplicación de impacto social. [15]

3.1. Localización y orientación en interiores

La localización en interiores (indoor localization) tiene como objetivo determinar la posición de un usuario, robot o dispositivo dentro de un edificio respecto de un sistema de referencia local. Esta posición puede expresarse en coordenadas bidimensionales (x, y) cuando se trabaja sobre el plano de una planta, o en coordenadas tridimensionales (x, y, z) cuando se requiere distinguir entre niveles o se trabaja con modelos volumétricos del entorno. Complementada con la orientación del agente, la posición define su pose completa, que en el caso general comprende seis grados de libertad: traslación en los tres ejes espaciales y rotación expresada como cuaternión o como ángulos de Euler (cabeceo, alabeo y guiñada). Para aplicaciones de navegación y visualización en planta baja, es habitual proyectar esta representación completa a una pose planar compuesta por la posición (x, y) y el ángulo de guiñada (yaw), que indica la dirección hacia la que apunta el dispositivo sobre el plano horizontal.

A diferencia de los entornos exteriores, donde el Sistema de Posicionamiento Global (GPS) proporciona cobertura continua con precisiones de entre 2 y 10 metros, en interiores la señal satelital sufre atenuaciones severas por las estructuras del edificio que la hacen inutilizable para aplicaciones que requieren precisión métrica. Esta limitación ha impulsado el desarrollo de una amplia variedad de tecnologías alternativas de localización indoor, que pueden agruparse en tres grandes familias según el tipo de señal o sensor empleado:

Basadas en señales de radiofrecuencia: WiFi, Bluetooth Low Energy (BLE), RFID y Ultra-Wideband (UWB), que estiman la posición a partir de la intensidad de señal recibida (RSSI) o de diferencias de

tiempo de llegada. Ofrecen despliegue sencillo pero requieren infraestructura adicional y presentan susceptibilidad a interferencias y multipath.

Basadas en campos físicos: sistemas de localización magnética que explotan anomalías en el campo magnético terrestre generadas por la estructura metálica del edificio, o sistemas de presión barométrica para estimación de planta.

Basadas en sensores de percepción: cámaras RGB y RGB-D, LiDAR y unidades inerciales (IMU), que construyen o consultan representaciones del entorno a partir de observaciones sensoriales directas. Este enfoque no requiere infraestructura instalada en el edificio y es el adoptado en el presente trabajo.

La localización basada en percepción visual presenta desafíos específicos que no están presentes en las tecnologías de radiofrecuencia. El primero es el fenómeno de aliasing perceptual (perceptual aliasing): la existencia de zonas del entorno visualmente indistinguibles “pasillos de apariencia homogénea, habitaciones con distribución similar” que pueden inducir asociaciones incorrectas entre la observación actual y una ubicación errónea del mapa. El segundo es la variabilidad ambiental: cambios de iluminación entre la fase de construcción del mapa y la fase de consulta, así como la presencia de elementos dinámicos (personas, mobiliario móvil) que alteran la apariencia de la escena. Ambos factores comprometen la estabilidad de la estimación, entendida como la ausencia de saltos abruptos entre localizaciones sucesivas espacialmente cercanas.

En función del nivel de precisión requerido por la aplicación, los sistemas de localización indoor pueden clasificarse en tres categorías: localización por zona (detección de la habitación o sector), localización métrica (posición continua con precisión de decímetros a metros) y localización con pose completa (posición y orientación). El presente trabajo se enmarca en esta última categoría, produciendo como salida una estimación de posición (x , y) y ángulo de orientación (yaw) del dispositivo sobre el plano de planta, lo que permite no solo ubicar al usuario en el edificio sino también determinar la dirección hacia la que se desplaza o apunta, información esencial para sistemas de navegación guiada e interfaces de asistencia. [16]

3.2. Estimación de pose y orientación

La estimación de pose constituye el problema central de cualquier sistema de localización, ya que determina de manera completa el estado geométrico de un agente o dispositivo en el espacio. En el contexto de la localización indoor, conocer la pose de un dispositivo implica conocer tanto su posición en el entorno como la dirección hacia la que apunta, información que es indispensable para generar instrucciones de navegación, proyectar resultados sobre un plano de planta y mantener coherencia espacial entre observaciones sucesivas. Las subsecciones que siguen describen la representación formal de la pose, los distintos esquemas para codificar la orientación y la forma en que el sistema propuesto proyecta la pose tridimensional a una representación planar adecuada para la visualización indoor.

a. Pose en seis grados de libertad (6-DOF)

En robótica y visión por computadora, la pose completa de un cuerpo rígido en el espacio tridimensional se describe mediante seis grados de libertad (6-DOF, six degrees of freedom): tres correspondientes a la traslación y tres a la rotación. [17]

Formalmente, la pose se representa como un par (t, R) , donde $t \in \mathbb{R}^3$ es el vector de posición en el sistema de referencia global y $R \in SO(3)$ es la matriz de rotación que describe la orientación del cuerpo respecto a dicho sistema. Esta representación es la base del problema de relocalización de cámara (camera relocalization), cuyo objetivo es estimar la pose 6-DOF de una cámara a partir de una única imagen de la escena. [18]

En el contexto del presente trabajo, las poses de los nodos del mapa son generadas por RTAB-Map tras la optimización del grafo de poses, y se almacenan en el archivo de poses con el siguiente formato por registro:

Donde t_s es la marca temporal, (t_x, t_y, t_z) es la posición en metros en el sistema de coordenadas global del mapa, y (q_x, q_y, q_z, q_w) es el cuaternión unitario que codifica la orientación. Para el entorno de prueba evaluado, el rango de posiciones cubrió aproximadamente 37.25 m en el eje X y 32.18 m en el eje Y, correspondiente a un área aproximada de 1 199 m².

La estimación de pose 6-DOF es inherentemente más compleja que la localización bidimensional, tanto por el mayor número de parámetros a determinar como por la necesidad de representar de forma no ambigua las rotaciones en el espacio tridimensional, aspecto que se aborda en la subsección siguiente [19].

b. Representación de orientación: cuaterniones y ángulos de Euler

La orientación de un cuerpo rígido en el espacio puede representarse de múltiples formas equivalentes, cada una con ventajas e inconvenientes según el contexto de uso .

Ángulos de Euler (RPY). La representación más intuitiva descompone la rotación en tres rotaciones sucesivas alrededor de ejes del sistema de referencia: alabeo (roll, ϕ), cabeceo (pitch, θ) y guiñada (yaw, ψ). Esta parametrización es fácilmente interpretable, pero presenta una singularidad conocida como bloqueo de cardán (gimbal lock) cuando el ángulo de cabeceo se aproxima a $\pm 90^\circ$, situación en la que dos de los tres grados de libertad de rotación colapsan y la representación pierde unicidad [4]. En aplicaciones de navegación indoor en plano horizontal, el bloqueo de cardán generalmente no ocurre, pero la representación RPY sigue siendo problemática para la composición eficiente de rotaciones.

Cuaterniones unitarios. La alternativa más empleada en robótica y SLAM es el cuaternión unitario $\mathbf{q} = q_w + q_x i + q_y j + q_z k$, con la restricción $|\mathbf{q}| = 1$ [20]. Los cuaterniones ofrecen tres ventajas fundamentales frente a los ángulos de Euler: eliminan el bloqueo de cardán, permiten la composición de rotaciones mediante producto de cuaterniones sin singularidades, y admiten interpolación esférica suave

(SLERP) entre orientaciones. En contrapartida, la interpretación geométrica directa es menos inmediata, ya que cada rotación es representada por dos cuaterniones antipodales (q y $-q$) equivalentes.

La conversión del cuaternión unitario (q_x, q_y, q_z, q_w) a la matriz de rotación $R \in SO(3)$ se obtiene mediante:

Esta representación es la empleada internamente por RTAB-Map para almacenar las poses optimizadas del grafo, y es el formato en que el sistema de localización propuesto recibe y procesa la orientación de cada nodo del mapa.

c. Proyección a pose planar (x, y, yaw)

Para la mayoría de las aplicaciones de navegación indoor en un edificio de un solo nivel, la pose tridimensional completa puede reducirse sin pérdida significativa de información a una pose planar de tres grados de libertad: posición horizontal (x, y) y ángulo de orientación ψ (yaw) [21]. Esta reducción es posible porque el desplazamiento vertical t_z varía de forma mínima en un recorrido por una planta del edificio” en el entorno evaluado, el rango de t_z fue de tan solo 1.77 m entre su valor mínimo y máximo” y porque el alabeo ϕ y el cabeceo (θ) son despreciables en el movimiento de un dispositivo que se desplaza sobre una superficie aproximadamente horizontal. El ángulo yaw se extrae directamente del cuaternión de orientación mediante la siguiente relación, que corresponde a la componente de rotación alrededor del eje Z vertical:

El resultado es un ángulo en radianes en el intervalo $(-\pi, \pi)$, que indica la dirección de orientación del dispositivo sobre el plano horizontal respecto al sistema de referencia del mapa. En el entorno evaluado, el rango de ángulos yaw registrado cubrió el intervalo completo $(-179.88^\circ, 179.98^\circ)$, confirmando que el recorrido de mapeo incluyó rotaciones en todas las direcciones.

La pose planar (t_x, t_y, ψ) constituye la salida del módulo de estimación de pose del sistema propuesto, y sirve de entrada directa para la transformación afín que proyecta la posición al sistema de píxeles del plano de planta, tal como se describe en la sección de Proceso Detallado.

3.3. SLAM (Localización y Mapeo Simultáneo):

Este método permite que un sistema móvil se desplace en un entorno desconocido mientras estima de manera simultánea su propia ubicación y construye un mapa del espacio recorrido. En otras palabras, SLAM resuelve de forma conjunta dos problemas fundamentales: determinar la trayectoria del sistema y generar una representación del entorno.

De manera general, el proceso SLAM se compone de varios elementos interrelacionados. Uno de ellos es la odometría, cuya función consiste en estimar el movimiento del sistema entre dos instantes consecutivos, ya sea a partir de imágenes, lecturas de sensores u otras mediciones. Esta estimación permite inferir cuánto se ha desplazado el sistema y en qué dirección. Sin embargo, dado que dichas mediciones no son perfectas, los errores se acumulan progresivamente a lo largo del recorrido, generando un fenómeno conocido como deriva (drift), el cual afecta la precisión de la trayectoria y del mapa construido.

Otro componente esencial es el reconocimiento de lugares, también denominado cierre de ciclo (loop closure). Este proceso ocurre cuando el sistema identifica que ha regresado a un sitio previamente visitado. Al detectar esta coincidencia, es posible comparar la estimación actual con la información almacenada anteriormente y corregir las desviaciones acumuladas por la odometría. Finalmente, el sistema incorpora una etapa de optimización, orientada a ajustar de manera conjunta las poses estimadas y la estructura del mapa para garantizar una representación más consistente del entorno.

a. Fundamentos de SLAM

Desde una perspectiva probabilística, el problema SLAM consiste en estimar la distribución de probabilidad conjunta sobre la trayectoria del agente $x_{1:t} = \{x_1, x_2, \dots, x_t\}$ y el mapa del entorno m , condicionada a las observaciones sensoriales $z_{1:t}$ y los comandos de movimiento $u_{1:t}$

$$p(x_{1:t}, m \mid z_{1:t}, u_{1:t})$$

Esta formulación evidencia que SLAM no produce una estimación determinista, sino una distribución de probabilidad que refleja la incertidumbre inherente tanto en el movimiento como en la percepción del sistema .

Según el tipo de sensor empleado, los sistemas SLAM se clasifican en LiDAR-SLAM, que utilizan escáneres láser para construir mapas de puntos de alta densidad, y Visual SLAM (V-SLAM), que emplean cámaras monoculares, estéreo o RGB-D. Los sistemas V-SLAM presentan la ventaja de operar con hardware de bajo costo y capturar información de apariencia visual del entorno, lo que los hace especialmente adecuados para la relocalización basada en reconocimiento de lugares.

Desde el punto de vista algorítmico, los enfoques SLAM se dividen en dos grandes familias:

Basados en filtros: emplean el Filtro de Kalman Extendido (EKF-SLAM) o filtros de partículas (FastSLAM) para mantener y actualizar la distribución de probabilidad de la pose y el mapa de forma incremental. Su complejidad computacional crece cuadráticamente con el número de características del mapa, lo que limita su escalabilidad.

Basados en grafos (Graph-Based SLAM): representan el historial de poses como un grafo y resuelven el problema de estimación mediante optimización no lineal sobre dicho grafo. Este enfoque es el dominante en los sistemas modernos de SLAM de largo plazo y gran escala.

En este trabajo, RTAB-Map opera como sistema de SLAM basado en grafos, empleando cámaras RGB-D para la construcción del mapa de referencia utilizado en la fase offline del sistema de localización.

b. Cierre de ciclos

El cierre de ciclo es el mecanismo mediante el cual un sistema SLAM detecta que el agente ha retornado a una posición previamente visitada, permitiendo incorporar una restricción adicional en el grafo de poses que corrige el error acumulado por la odometría [22]. Sin este mecanismo, la deriva acumulada a lo largo de trayectorias largas produce inconsistencias severas en el mapa, manifestadas como duplicaciones de zonas o deformaciones en la geometría del espacio reconstruido.

La detección del cierre de ciclo puede abordarse desde dos perspectivas complementarias:

Detección geométrica: compara la posición actual estimada con posiciones previas del mapa y verifica si la distancia entre ellas es menor que un umbral. Este enfoque es computacionalmente eficiente pero depende de la precisión de la odometría acumulada, lo que lo hace poco fiable en trayectorias largas con deriva significativa.

Detección por apariencia (appearance-based loop closure): compara la imagen actual con imágenes almacenadas en el historial del mapa utilizando representaciones visuales compactas, como bolsas de palabras visuales (Bag of Visual Words, BoVW) construidas a partir de descriptores de características locales. Este enfoque es robusto frente a la deriva de la odometría, ya que la decisión de cierre de ciclo se basa únicamente en la similitud visual entre observaciones, independientemente de la posición estimada.

RTAB-Map implementa detección de cierre de ciclo por apariencia mediante un índice de bolsas de palabras visuales construido incrementalmente. Cuando la similitud entre la imagen actual y una imagen histórica supera un umbral configurable, el sistema verifica la correspondencia mediante comparación de características locales y, si se confirma geoméricamente, añade una arista de cierre de ciclo al grafo de poses. En el contexto del presente trabajo, este mecanismo garantizó la coherencia global del mapa de referencia durante el recorrido de captura, reduciendo el error en las poses de los 1131 nodos almacenados.

c. Optimización de grafos de poses

En el enfoque de SLAM basado en grafos, el problema de estimación se formula como un grafo

$G = (V, E)$ donde los nodos $V = x_1, x_2, \dots, x_n$ representan las poses estimadas del agente en distintos instantes de tiempo, y las aristas E representan restricciones espaciales entre pares de poses. Estas restricciones provienen de dos fuentes: las mediciones de odometría consecutiva, que generan aristas entre nodos adyacentes, y los cierres de ciclo detectados, que generan aristas de largo alcance entre nodos distantes en el tiempo, pero cercanos en el espacio.

Cada arista $e_{ij} \in E$ codifica la transformación relativa medida $\widetilde{z}_{ij}$ entre las poses x_i y x_j , junto con su matriz de información Ω_{ij} que cuantifica la confianza en dicha medición. El problema de optimización consiste en encontrar la configuración de poses $x^* = x_1^* x_2^* \dots x_n^*$ que minimiza la suma ponderada de los errores cuadráticos residuales sobre todas las aristas del grafo [19]:

$$x^* = \arg \min_x \sum_{(i,j) \in E} e_{ij}(x)^T \Omega_{ij} e_{ij}(x)$$

Donde $e_{ij}(x)$ es la función de error que mide la discrepancia entre la transformación predicha por las poses actuales y la restricción observada; Ω_{ij} representa la matriz de información asociada a dicha restricción. Este problema de mínimos cuadrados no lineales se resuelve iterativamente mediante algoritmos como Gauss-Newton o Levenberg-Marquardt, aprovechando la estructura dispersa (sparse) de la matriz de información del sistema para lograr eficiencia computacional. El resultado de la optimización es un conjunto de poses refinadas que distribuyen el error acumulado de forma globalmente coherente. En RTAB-Map, esta etapa se ejecuta tras cada cierre.

d. RTAB-Map: arquitectura y base de datos SQLite

RTAB-Map (Real-Time Appearance-Based Mapping) es un sistema de SLAM visual y LiDAR de código abierto diseñado para operar en tiempo real con restricciones de memoria, capaz de construir y mantener mapas de entornos de gran escala durante largos períodos de operación. Su arquitectura distingue tres niveles de memoria que gestionan el flujo de información durante el recorrido:

- Memoria de trabajo (Working Memory, WM): contiene los nodos activos recientemente procesados, sobre los cuales se ejecuta la detección de cierre de ciclo en tiempo real.
- Memoria a corto plazo (Short-Term Memory, STM): buffer de los nodos más recientes antes de ser transferidos a la memoria de trabajo.
- Memoria a largo plazo (Long-Term Memory, LTM): almacén persistente en disco de todos los nodos del mapa, consultado cuando se detecta una posible revisita a una zona archivada.

Este esquema de gestión de memoria permite que RTAB-Map mantenga una frecuencia de procesamiento constante independientemente del tamaño del mapa, transfiriendo nodos poco relevantes desde la memoria de trabajo a la memoria a largo plazo cuando la carga computacional supera un umbral configurable.

Base de datos SQLite. Toda la información generada durante el recorrido de mapeo se persiste en una base de datos relacional en formato SQLite (archivo .db). La estructura interna de esta base de datos incluye, entre otras, dos tablas principales relevantes para el sistema de localización propuesto: la tabla Data, que almacena las imágenes RGB de cada nodo en formato comprimido (zlib), y la tabla Node, que almacena la pose optimizada de cada nodo en formato binario (pose_blob) con la estructura de traslación y cuaternión descrita en la sección 3.2.1. En el entorno evaluado en este trabajo, la base de datos en archivo .db presentó una estructura de 9 tablas internas y contenía 1131 registros de nodos, cada uno con su imagen

RGB de 720×960 píxeles y su posición y orientación (pose) en 6-DOF, con cobertura total de 100 % entre imágenes y poses.

Es importante destacar que, en la arquitectura propuesta en este trabajo, RTAB-Map actúa exclusivamente como herramienta de construcción del mapa en la fase offline: una vez generada la base de datos, el sistema de localización extrae las imágenes y poses mediante procesamiento propio y no requiere que RTAB-Map permanezca en ejecución durante la fase de consulta. Esta separación arquitectónica permite desplegar el módulo de localización de forma independiente de la infraestructura de SLAM [23].

3.4. Representación del espacio interior

Para que un sistema de localización indoor pueda asociar una estimación de pose con un lugar concreto dentro de un edificio, es necesario disponer de una representación del espacio que capture la geometría, la topología o la apariencia del entorno de forma computable. La elección de esta representación determina qué información queda disponible para el módulo de localización, qué operaciones pueden realizarse sobre el mapa y qué nivel de precisión es alcanzable en la estimación de pose [24].

En la práctica, los sistemas modernos de SLAM y localización visual suelen emplear representaciones complementarias a distintos niveles de abstracción: una representación métrica densa para la descripción geométrica del espacio, una representación topológica para modelar la conectividad entre lugares, y un plano de referencia 2D para la visualización y la interacción con el usuario.

a. Mapas de ocupación 2D

El mapa de ocupación (occupancy grid map) es una representación discreta del entorno en la que el espacio se divide en celdas de tamaño fijo, y a cada celda se le asigna una probabilidad de estar ocupada por un obstáculo. Formalmente, dado el conjunto de observaciones sensoriales $\mathbf{z}_{1:t}$ y la trayectoria conocida $\mathbf{x}_{1:t}$, el mapa de ocupación $\mathbf{m}$ se estima como un campo de variables de Bernoulli independientes $m_i \in \{0,1\}$, donde $m_i = 1$ indica celda ocupada. La estimación se actualiza de forma incremental mediante la regla de Bayes aplicada a cada celda [21]:

$$P(m_i | \mathbf{z}_{1:t}, \mathbf{x}_{1:t}) \propto P(z_t | m_i, x_t) P(m_i | \mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1})$$

En entornos indoor, la resolución habitual de la cuadrícula oscila entre 0.05 y 0.10 m por celda, lo que permite representar con suficiente detalle puertas, pasillos y paredes. RTAB-Map genera este tipo de mapa como producto adicional del proceso de SLAM, proyectando las lecturas del sensor de profundidad sobre el plano horizontal y acumulando las evidencias de ocupación en una cuadrícula 2D [21].

Aunque el mapa de ocupación proporciona una representación geométrica precisa del espacio navegable, su uso directo como índice de localización presenta limitaciones: no captura información de apariencia visual y su consulta requiere comparar observaciones con un modelo geométrico del entorno, lo que exige sensores métricos y estimaciones de pose suficientemente precisas. Por esta razón, en el sistema propuesto el mapa de ocupación no se emplea como mecanismo de localización, sino únicamente como referencia visual para verificar la coherencia geométrica del recorrido de mapeo.

b. Grafos topológicos de poses

Una representación complementaria al mapa de ocupación es el grafo topológico de poses, en el cual el espacio del entorno se modela mediante un grafo $G = (V, E)$ cuyos nodos V

representan lugares o posiciones relevantes del recorrido, y cuyas aristas E

representan las conexiones navegables entre ellos. A diferencia del mapa de ocupación, que modela el espacio de forma densa y continua, el grafo topológico es una representación discreta y compacta que captura la estructura de conectividad del entorno [25].

En los sistemas de SLAM modernos, la representación topológica y la métrica no son mutuamente excluyentes, sino que se combinan en estructuras denominadas mapas topométricos (topometric maps): cada nodo del grafo almacena no solo la identidad del lugar sino también su pose métrica en el sistema de referencia global, lo que permite recuperar información geométrica precisa a partir de consultas topológicas. RTAB-Map implementa esta representación híbrida de forma nativa: el grafo de poses que construye durante el recorrido asigna a cada nodo un identificador único, una imagen RGB y una pose 6-DOF optimizada, conformando un mapa topométrico en el que cada lugar queda caracterizado tanto por su posición en el espacio como por su apariencia visual [26].

En el contexto del presente trabajo, el grafo topométrico generado por RTAB-Map constituye la estructura de referencia sobre la que opera el sistema de localización. Los 1131 nodos del grafo conforman el índice visual del sistema: cada nodo es un punto de referencia espacial identificable tanto por su posición métrica (t_x, t_y, t_z) como por su representación vectorial profunda obtenida mediante ResNet50. La localización de una imagen de consulta equivale, desde esta perspectiva, a identificar el nodo del grafo topométrico cuya apariencia visual es más coherente con la observación actual, problema que se aborda mediante la recuperación por similitud en el índice FAISS descrita en la sección 3.5.

c. Plano de planta y transformación afín para proyección 2D

Para la visualización e interpretación de los resultados de localización por parte de usuarios no especializados, la representación más adecuada del espacio interior es el plano de planta (floor plan): una imagen bidimensional del edificio obtenida mediante proyección ortogonal vertical, que muestra la distribución de espacios, tabiques, puertas y corredores en una escala conocida. A diferencia del mapa de ocupación “generado automáticamente por el sistema de SLAM” el plano de planta es habitualmente un documento arquitectónico elaborado previamente, cuya escala y orientación deben ponerse en correspondencia con el sistema de coordenadas del mapa RTAB-Map mediante una calibración explícita.

Transformación afín de coordenadas. La correspondencia entre el sistema de referencia métrico del mapa (t_x, t_y) y el sistema de píxeles del plano de planta (u, v) se establece mediante una transformación afín que combina tres operaciones geométricas en el plano [26]:

Rotación por un ángulo θ , que alinea los ejes del mapa RTAB-Map con la orientación del plano de planta.

Escala uniforme por un factor s (expresado en píxeles por metro), que convierte distancias métricas en distancias en píxeles.

Traslación $(\Delta u, \Delta v)$, que desplaza el origen del sistema de coordenadas del mapa al punto correspondiente en la imagen del plano.

$$P(m_i | z_{1:t}, x_{1:t}) \propto P(z_t | m_i, x_t) P(m_i | z_{1:t-1}, x_{1:t-1})$$

donde (u, v) son las coordenadas del punto proyectado en píxeles sobre el plano de planta. Los cuatro parámetros $\theta, s, \Delta u, \Delta v$ se determinan mediante un proceso de calibración que identifica al menos dos puntos de correspondencia entre el sistema de coordenadas del mapa y posiciones conocidas en el plano de planta, y se almacenan en el archivo de configuración del sistema para su reutilización en todas las consultas posteriores.

De forma análoga, el ángulo yaw ψ de orientación del dispositivo, obtenido según la Ecuación, se transforma al sistema angular de la imagen sumando la rotación de calibración θ , de modo que la flecha de orientación representada sobre el plano de planta refleje correctamente la dirección hacia la que apunta el dispositivo en el espacio físico. Esta transformación conjunta de posición y orientación constituye la última etapa del pipeline de localización antes de la publicación del resultado mediante la capa de servicio REST.

3.5. Relocalización visual basada en aprendizaje profundo

La relocalización visual es el proceso mediante el cual un sistema estima la pose de un dispositivo a partir de una imagen de consulta, comparándola con un conjunto de imágenes de referencia cuyas poses son conocidas [18]. A diferencia de la odometría visual, que estima el movimiento incremental entre fotogramas consecutivos, la relocalización opera sobre imágenes aisladas sin continuidad temporal, lo que la hace especialmente adecuada para escenarios donde el sistema ha perdido el hilo de seguimiento o donde el usuario inicia la consulta desde una posición arbitraria.

Los enfoques clásicos de relocalización visual se basaban en la extracción y comparación de descriptores de características locales sobre la totalidad del mapa de referencia, con complejidad computacional lineal en el número de nodos. Los métodos basados en aprendizaje profundo superan esta limitación al aprender representaciones visuales compactas de alta dimensión que capturan la apariencia global de los lugares y permiten la recuperación eficiente mediante índices de búsqueda vectorial, reduciendo el conjunto de candidatos que posteriormente requiere verificación geométrica, reduciendo el espacio de candidatos antes de la verificación geométrica [18]. La arquitectura propuesta en este trabajo integra ambos paradigmas en un pipeline híbrido de dos etapas que se describe a lo largo de esta sección.

a. Redes neuronales convolucionales (CNN) para representación visual

Las redes neuronales convolucionales (Convolutional Neural Networks, CNN) son arquitecturas de aprendizaje profundo diseñadas para procesar datos con estructura espacial, como imágenes. Su principio de operación consiste en aplicar sucesivas capas de convolución, activación no lineal y reducción de resolución (pooling) para extraer representaciones jerárquicas del contenido visual: las capas iniciales detectan bordes y texturas de bajo nivel, mientras que las capas profundas codifican estructuras semánticas de alto nivel como formas, objetos y escenas. Esta jerarquía de representación hace que las activaciones de las capas profundas sean especialmente útiles como descriptores visuales globales para la recuperación de imágenes similares.

3.5.1.1 Arquitectura ResNet y transferencia de aprendizaje

Las redes residuales (Residual Networks, ResNet) fueron propuestas por He et al [27] para resolver el problema del desvanecimiento del gradiente (vanishing gradient) que limita el entrenamiento de redes muy profundas. La idea central es incorporar conexiones de atajo (skip connections) que suman la entrada de un bloque directamente a su salida, forzando a cada

bloque a aprender únicamente la función residual respecto a la identidad:

$$y = \mathcal{F}(x, \{W_i\}) + x$$

donde x es la entrada del bloque, $\mathcal{F}(x, W_i)$ es la transformación aprendida por las capas convolucionales y y es la salida. Este mecanismo permite entrenar redes de hasta 152 capas con mayor estabilidad y mejor rendimiento que las arquitecturas previas.

La transferencia de aprendizaje (transfer learning) es la práctica de reutilizar los pesos de una red entrenada en una tarea de gran escala “habitualmente clasificación de imágenes sobre ImageNet, con 1.2 millones de imágenes y 1 000 categorías” como punto de partida para una tarea distinta. En el contexto de la relocalización visual, las representaciones aprendidas en ImageNet capturan propiedades visuales generales (texturas, patrones, estructuras) que resultan transferibles a la descripción de lugares indoor, sin necesidad de reentrenamiento sobre el entorno específico [27].

3.5.1.2 Extracción de embedding global (ResNet50, 2048D, L2-norm)

En el sistema propuesto se utiliza la arquitectura ResNet50 preentrenada sobre ImageNet como extractor de características visuales. La red emplea conexiones residuales que facilitan el entrenamiento de arquitecturas profundas. Para la generación de los embeddings se elimina la capa final de clasificación (fully connected), conservando como salida la activación de la capa de agrupación promedio global (global average pooling, GAP), que produce un vector de 2048 dimensiones por imagen[27]. Este vector constituye el embedding visual global del lugar: una representación compacta que codifica la apariencia de la imagen en un espacio métrico de alta dimensión.

Para que la similitud entre embeddings sea independiente de la magnitud de los vectores y pueda expresarse como un producto interno eficiente, cada vector se normaliza a norma L2 unitaria antes de ser almacenado en el índice:

$$\hat{\mathbf{v}} = \frac{\mathbf{v}}{|\mathbf{v}|_2}$$

Esta normalización garantiza que $|\hat{\mathbf{v}}|_2 = 1$ y transforma la similitud coseno en un simple producto interno: $\cos(\hat{\mathbf{v}}_i, \hat{\mathbf{v}}_j) = \hat{\mathbf{v}}_i \cdot \hat{\mathbf{v}}_j$, condición que habilita el uso directo del índice IndexFlatIP de FAISS. En el entorno evaluado, este proceso generó una matriz de $1\,131 \times 2048$ componentes, correspondiente a 1 131 embeddings L2-normalizados, con un tamaño aproximado de 8.84 MiB (9.27 MB en notación decimal).

3.5.2 Búsqueda eficiente de similitud en alta dimensión

3.5.2.1 Similitud coseno en espacios vectoriales

La similitud coseno mide el coseno del ángulo θ entre dos vectores en un espacio de alta dimensión, independientemente de su magnitud:

$$\cos(\theta) = \frac{a \cdot b}{\|a\|_2 \|b\|_2}$$

Su valor está acotado en el intervalo $[-1,1]$: un valor de 1 indica vectores con idéntica dirección (máxima similitud), 0 indica ortogonalidad (sin correlación) y -1 indica direcciones opuestas. Para vectores L2-normalizados, la expresión se reduce al producto interno $a \cdot b$, lo que hace la computación especialmente eficiente.

En los resultados obtenidos con la imagen de consulta evaluada, el candidato más similar recuperado por el índice obtuvo una similitud coseno de 0.9562, mientras que el nodo correcto "que superó la verificación geométrica" presentó una similitud de 0.9027, ocupando la quinta posición en el ranking de recuperación. Este resultado evidencia que la similitud coseno en el espacio de embeddings globales no es condición suficiente para garantizar la localización correcta, y justifica la necesidad de la etapa de verificación geométrica que se describe en la sección 3.5.4.

3.5.2.2 Índices FAISS: IndexFlatIP e IndexHNSWFlat

FAISS (Facebook AI Similarity Search) es una biblioteca de código abierto para búsqueda eficiente de vecinos más cercanos en espacios vectoriales de alta dimensión, con soporte para ejecución en CPU y GPU. Para el caso de la relocalización visual, FAISS permite recuperar eficientemente los k nodos del mapa más similares a una imagen de consulta mediante distintas estructuras de indexación, cuya elección depende del tamaño de la colección y de los requisitos de precisión y latencia.

El sistema propuesto implementa dos estrategias de indexación según el número de nodos del mapa:

IndexFlatIP (búsqueda exacta por producto interno): realiza una comparación exhaustiva entre el vector de consulta y todos los vectores del índice, garantizando la recuperación del vecino más cercano exacto.

Su complejidad es $O(n \cdot d)$ por consulta, donde n es el número de vectores y d su dimensión. Es la opción óptima para mapas con menos de 5 000 nodos, donde el costo lineal es computacionalmente asumible. En el entorno evaluado (1 131 nodos < 5 000), se empleó este tipo de índice.

IndexHNSWFlat (Hierarchical Navigable Small World, HNSW): construye un grafo jerárquico de múltiples capas sobre los vectores del índice, con parámetros $M = 32$ (conexiones por nodo) y $efSearch=64$ (amplitud de búsqueda). Permite búsqueda aproximada con complejidad $O(\log n)$, sacrificando una pequeña fracción de precisión en la recuperación a cambio de tiempos de respuesta sublineales. Se activa automáticamente cuando el número de nodos supera el umbral de 5 000.

3.5.3 Descriptores de características locales

3.5.3.1 ORB (Oriented FAST and Rotated BRIEF)

ORB es un descriptor de características locales binario propuesto por Rublee et al. [6] como alternativa eficiente a SIFT y SURF, libre de restricciones de patentes y diseñado para aplicaciones en tiempo real. Su funcionamiento combina dos componentes:

Detector FAST orientado: detecta puntos de interés basándose en la diferencia de intensidad entre un píxel y su vecindario circular, e incorpora un esquema de orientación basado en el centroide de intensidad del parche que dota al descriptor de invarianza a la rotación.

Descriptor BRIEF rotado (rBRIEF): genera una cadena binaria de longitud configurable comparando la intensidad de pares de píxeles seleccionados aleatoriamente en el parche alineado según la orientación estimada. La distancia de Hamming entre descriptores permite el emparejamiento eficiente mediante operaciones de bits.

En el sistema propuesto, ORB actúa como descriptor de línea base en la etapa de verificación geométrica, con un máximo de 2 000 características por imagen. Su ventaja principal es la velocidad de extracción y emparejamiento, lo que lo hace apropiado para escenarios con restricciones de cómputo, aunque a costa de menor precisión frente a descriptores neuronales en condiciones de cambio de iluminación o punto de vista.

3.5.3.2 SuperPoint y emparejamiento LightGlue

SuperPoint es un detector-descriptor de características locales completamente convolucional entrenado mediante supervisión autosupervisada [7]. Su arquitectura comparte un encoder convolucional que genera representaciones intermedias, y dos cabezas de decodificación independientes: una para la detección de puntos de interés (interest point detector) y otra para el cálculo de descriptores flotantes de 256 dimensiones. El entrenamiento emplea adaptación homográfica (Homographic Adaptation): se aplican homografías aleatorias a imágenes reales para generar pares con correspondencias conocidas, lo que permite aprender detectores robustos sin anotaciones manuales [28].

LightGlue es una red neuronal transformer para el emparejamiento de características locales entre pares de imágenes [8]. Recibe como entrada los conjuntos de puntos y descriptores de dos imágenes y produce una asignación óptima entre sus características, aprendiendo a considerar contexto global mediante mecanismos de atención cruzada. A diferencia de su predecesor SuperGlue, LightGlue incorpora un mecanismo de terminación anticipada (early stopping) que adapta la profundidad de procesamiento a la dificultad del emparejamiento específico, lo que se traduce en tiempos de inferencia significativamente menores sin pérdida de precisión [28].

En el sistema propuesto, SuperPoint con emparejamiento LightGlue constituye el descriptor principal configurado para la verificación geométrica, con 2 000 características máximas por imagen, umbral de ratio de Lowe de 0.75 y mínimo de 20 correspondencias requeridas. Los experimentos de evaluación sobre la imagen de consulta disponible produjeron 91 correspondencias y 36 inliers RANSAC con este descriptor, lo que evidencia su capacidad para establecer correspondencias precisas y densas incluso en condiciones de ligeras variaciones de punto de vista.

3.5.4 Verificación geométrica por homografía

3.5.4.1 Homografía planar y sus condiciones de validez

Una homografía es una transformación proyectiva que relaciona los puntos de una escena planar vistos desde dos puntos de cámara distintos, o bien dos vistas de cualquier escena cuando la cámara experimenta exclusivamente una rotación pura. Formalmente, la homografía H es una matriz 3×3 de rango completo, definida salvo escala, que relaciona puntos homogéneos $p = (u, v, 1)^T$ en la primera imagen con sus correspondencias $p' = (u', v', 1)^T$ en la segunda [29]:

$$p' \sim Hp, \quad H \in \mathbb{R}^{3 \times 3}, \quad \det(H) \neq 0$$

La homografía tiene 8 grados de libertad y puede estimarse con un mínimo de 4 correspondencias de puntos no colineales. En entornos indoor, la suposición de planaridad aproximada es razonable para imágenes de pasillos, paredes y zonas de suelo, ya que los principales elementos visuales de la escena se encuentran predominantemente en planos paralelos al plano imagen [29]. Esta propiedad justifica el uso de la homografía como modelo geométrico en la etapa de verificación del sistema propuesto.

3.5.4.2 Estimación robusta mediante RANSAC

La estimación directa de la homografía a partir del conjunto completo de correspondencias detectadas por el descriptor es sensible a correspondencias erróneas (outliers), es decir, pares de puntos que han sido emparejados incorrectamente. El algoritmo RANSAC (Random Sample Consensus) [30] aborda este problema de forma robusta mediante el siguiente esquema iterativo:

Seleccionar aleatoriamente un subconjunto mínimo de correspondencias (4 pares para la homografía).

Estimar la homografía H a partir de ese subconjunto mediante el algoritmo DLT (Direct Linear Transform). Evaluar cuántas correspondencias del conjunto total son inliers, definidos como aquellas cuyo error de reproyección es menor que un umbral ϵ : $|H\mathbf{p} - \mathbf{p}_2'| < \epsilon$.

Conservar la hipótesis con mayor número de inliers y repetir durante un número fijo de iteraciones. En el sistema propuesto, el umbral de reproyección es $\epsilon = 5.0$ píxeles, y una correspondencia se considera geoméricamente válida si cumple simultáneamente dos criterios: número de inliers $N_{\text{inliers}} \geq 15$ y razón de inliers $r_{\text{inliers}} = N_{\text{inliers}}/N_{\text{matches}} \geq 0.25$. En la evaluación realizada, la imagen de consulta produjo 36 inliers de 91 correspondencias totales (razón 0.396), superando ambos umbrales y confirmando la validez geométrica de la localización con el nodo.

3.5.5 Clasificación semántica de escenas

3.5.5.1 Modelos multimodales visión-lenguaje (CLIP)

CLIP (Contrastive Language–Image Pretraining) es un modelo multimodal propuesto por Radford et al. [31] que aprende representaciones visuales y lingüísticas en un espacio de embedding compartido, entrenado mediante aprendizaje contrastivo sobre 400 millones de pares imagen-texto recopilados de Internet. La función objetivo del entrenamiento maximiza la similitud entre los embeddings de imágenes y sus descripciones de texto correspondientes, y minimiza la similitud con las descripciones no correspondientes dentro de cada lote de entrenamiento.

Formalmente, dado un encoder visual f_V y un encoder textual f_T , CLIP aprende a minimizar la pérdida contrastiva simétrica sobre pares (I_i, T_i) :

$$\mathcal{L} = -\frac{1}{2N} \sum_{i=1}^N \left[\log \left[\frac{\exp(\langle f_V(I_i), f_T(T_i) \rangle / \tau)}{\sum_{j=1}^N \exp(\langle f_V(I_i), f_T(T_j) \rangle / \tau)} + \log \left[\frac{\exp(\langle f_T(T_i), f_V(I_i) \rangle / \tau)}{\sum_{j=1}^N \exp(\langle f_T(T_j), f_V(I_i) \rangle / \tau)} \right] \right]$$

Donde f_V y f_T representan los codificadores visual y textual, respectivamente; N corresponde al número de pares imagen-texto del lote; τ es un parámetro de temperatura aprendible y $\langle f_V(I_i), f_T(T_i) \rangle$ denota el producto interno entre embeddings normalizados.. Este entrenamiento produce representaciones alineadas entre modalidades que permiten comparar directamente imágenes con descripciones textuales [31].

3.5.5.2 Clasificación zero-shot para etiquetado de lugares indoor

La capacidad más notable de CLIP para las aplicaciones de localización es la clasificación zero-shot: dado un conjunto de etiquetas de clase expresadas como texto, CLIP puede clasificar una imagen sin haber sido entrenado explícitamente en esas categorías . [31]. El procedimiento consiste en calcular el embedding visual de la imagen de consulta y compararlo mediante similitud coseno con los embeddings de las descripciones textuales de cada clase candidata, seleccionando la clase con mayor similitud.

En el sistema propuesto, este mecanismo se emplea para asignar a cada imagen de consulta una etiqueta semántica del tipo de lugar indoor a partir de un vocabulario de ocho categorías: pasillo, sala de reuniones, laboratorio, área de descanso, recepción, escalera, baño y zona común. Esta clasificación no requiere ningún proceso de entrenamiento o ajuste específico sobre el entorno evaluado, lo que la hace inmediatamente transferible a nuevos edificios sin costo de anotación. La etiqueta semántica resultante se añade a la respuesta del servicio REST como información contextual complementaria a la pose estimada, enriqueciendo la descripción del lugar localizado más allá de sus coordenadas numéricas.

3.6. Accesibilidad y navegación indoor para personas con discapacidad visual

La pérdida de visión representa una de las discapacidades sensoriales con mayor impacto en la autonomía cotidiana de las personas. Según la Organización Mundial de la Salud, al menos 2,200 millones de personas en el mundo padecen alguna forma de deterioro visual, y una proporción significativa de ellas enfrenta barreras concretas para desplazarse de manera independiente en entornos cerrados [32]. A diferencia del espacio exterior, donde el GPS y las tecnologías de posicionamiento satelital proveen referencias geográficas continuas, el interior de edificios” hospitales, centros comerciales, universidades, estaciones de transporte” carece de infraestructura de navegación estandarizada. Esta ausencia convierte la movilidad indoor en un desafío técnico y social que los sistemas de localización visual basados en aprendizaje profundo tienen el potencial de abordar de forma eficaz.

a. Necesidades de navegación autónoma en interiores

La navegación autónoma en espacios interiores implica la capacidad de un sistema para determinar la posición y orientación de un usuario dentro de un entorno estructurado, sin depender de señales satelitales externas. Para personas con discapacidad visual, esta capacidad no es simplemente un complemento de comodidad, sino un requisito fundamental para la participación activa en la vida social, laboral y educativa [33].

Los entornos interiores presentan características que los hacen intrínsecamente más complejos que los espacios abiertos. La variabilidad de la iluminación artificial, la presencia de estructuras repetitivas” como pasillos simétricos o áreas de espera homogéneas”, los cambios dinámicos en la disposición de obstáculos y la ausencia de puntos de referencia naturales dificultan el proceso de auto-localización [34]. A ello se suma la necesidad de ofrecer estimaciones de posición con latencia reducida, dado que un sistema de guiado en tiempo real debe actualizarse con suficiente frecuencia para resultar útil durante la marcha.

Desde la perspectiva del usuario con discapacidad visual, los requerimientos de un sistema de navegación indoor se agrupan en cuatro categorías principales:

Precisión posicional. La estimación de la posición debe ser suficientemente aproximada para distinguir entre puntos de interés próximos, como la entrada de dos salas adyacentes o el inicio de una escalera. Estudios en el área de navegación asistiva indican que un error de posición inferior a 1.5 metros es aceptable para la mayoría de las tareas de orientación [35].

Estimación de orientación. Conocer la posición sin saber hacia dónde se está orientando el usuario resulta insuficiente para generar instrucciones de giro. Los sistemas avanzados integran la estimación del ángulo de orientación (yaw) para producir instrucciones del tipo "gire 30 grados a la izquierda" con respecto al sistema de referencia del mapa [36].

Bajo acoplamiento con infraestructura. Las soluciones que requieren instalación previa de balizas Bluetooth, etiquetas RFID o redes Wi-Fi densas imponen costos de despliegue que muchas instituciones no pueden asumir. Los sistemas basados puramente en visión” que operan con la cámara del dispositivo móvil del usuario” eliminan esta dependencia y resultan más viables para entornos de recursos limitados [37].

Operación en tiempo real. El sistema debe producir una estimación de localización en el rango de segundos para ser funcional durante la navegación activa. Latencias superiores a 10 segundos degradan la experiencia de usuario y pueden comprometer la seguridad en entornos con obstáculos dinámicos [35].

Satisfacer simultáneamente estas cuatro condiciones exige combinar recuperación eficiente de imágenes de referencia, verificación geométrica de correspondencias visuales y proyección de la pose estimada sobre una representación comprensible del entorno, como un plano de planta en dos dimensiones.

b. Sistemas de localización como soporte a la movilidad inclusiva

La movilidad inclusiva se entiende como el conjunto de condiciones tecnológicas, físicas e institucionales que permiten a todas las personas, independientemente de sus capacidades físicas o sensoriales, desplazarse de manera autónoma y segura [38]. En el contexto de la discapacidad visual, los sistemas de localización indoor representan una capa tecnológica habilitante que, integrada con interfaces de audio o hápticas, puede transformar el acceso a espacios públicos complejos.

Los primeros sistemas de navegación para invidentes en interiores se basaron en tecnologías de radiofrecuencia “Wi-Fi fingerprinting, Bluetooth Low Energy (BLE) beacons” que ofrecían precisión de 2 a 5 metros con infraestructura dedicada [39]. Si bien estos enfoques demostraron viabilidad conceptual, su adopción masiva fue limitada por los costos de instalación y mantenimiento, así como por la degradación de precisión ante cambios en la disposición del entorno o actualizaciones en el firmware de los dispositivos emisores.

El surgimiento de la localización basada en reconocimiento de imagen marcó un cambio de paradigma. Sistemas como NetVLAD [9] y DBoW2 introdujeron la posibilidad de recuperar la posición mediante la comparación de descriptores visuales extraídos de la imagen capturada con un banco de imágenes georreferenciadas del entorno. Este enfoque no requiere infraestructura adicional y se adapta naturalmente

a dispositivos móviles de consumo masivo, ampliando significativamente el potencial de acceso para personas con discapacidad visual.

La integración de redes neuronales profundas como ResNet [40] para la generación de representaciones visuales densas y compactas elevó la robustez de estos sistemas frente a cambios de iluminación y variaciones en el ángulo de captura, condiciones frecuentes en entornos hospitalarios o universitarios. La combinación de recuperación semántica por similitud de embeddings con verificación geométrica mediante RANSAC [41] permite además distinguir entre candidatos visualmente similares pero espacialmente distintos, reduciendo las tasas de falso positivo que afectaban a los sistemas puramente basados en apariencia.

Desde una perspectiva de diseño centrado en el usuario, la interfaz de resultados es tan determinante como la precisión del algoritmo. La proyección de la posición estimada sobre un plano de planta 2D, accesible a través de una API REST, facilita la integración con lectores de pantalla, aplicaciones de guiado por voz y plataformas de asistencia cognitiva. Este modelo de arquitectura desacoplada “donde el motor de localización expone sus resultados como servicio independiente” favorece la reutilización en múltiples interfaces sin necesidad de modificar la lógica central de estimación de pose [35].

Los avances recientes en clasificación semántica cero-disparo, apoyados en modelos del tipo CLIP [42], abren además la posibilidad de enriquecer el resultado de localización con etiquetas de contexto "entrada principal", "sala de espera", "escalera norte" que complementan la posición numérica con información orientada al usuario. Esta característica resulta especialmente valiosa para personas con discapacidad visual, para quienes los nombres de los lugares constituyen referencias más intuitivas que las coordenadas cartesianas.

En síntesis, la evolución desde sistemas dependientes de infraestructura hacia arquitecturas de localización puramente visual, con capacidades semánticas y latencias compatibles con el uso en tiempo real, sitúa a los sistemas de relocalización basados en aprendizaje profundo como una tecnología de alto impacto para la inclusión de personas con discapacidad visual en entornos interiores complejos.

4. ARQUITECTURA DE LOCALIZACIÓN INDOOR

Como se expuso anteriormente, el objetivo de este proyecto es estimar la localización y la orientación de un usuario o dispositivo dentro de un edificio, tomando como referencia un mapa previamente generado. Asimismo, en el capítulo anterior se presentaron los conceptos fundamentales que permiten comprender la arquitectura de localización indoor propuesta.

La solución planteada se organiza en dos fases claramente diferenciadas. La primera corresponde a una fase offline de preparación, ejecutada una única vez sobre la base de datos del mapa: a partir de ella se extraen las imágenes de cada nodo y sus poses asociadas, se construye un índice estructurado que relaciona cada nodo con su imagen y su pose, y se generan los embeddings visuales de 2048 dimensiones mediante una red ResNet50, los cuales se almacenan junto con un índice FAISS para búsqueda eficiente por similitud coseno. Todos estos artefactos se persisten en disco y quedan disponibles para ser reutilizados en cada consulta posterior.

La segunda es una fase online de localización, que se ejecuta de forma independiente por cada imagen de consulta recibida. Esta fase se estructura a su vez en dos etapas encadenadas. En la Fase offline se procesa la Recuperación profunda, la imagen de consulta es procesada por la misma red ResNet50 para obtener su embedding L2-normalizado, el cual se compara contra el índice FAISS para recuperar los K nodos candidatos con mayor similitud. En la Fase online se hace una verificación geométrica, cada candidato es evaluado mediante matching de puntos clave con SuperPoint+LightGlue (prueba de ratio 0.75) y posteriormente validado con RANSAC sobre una homografía, exigiendo un mínimo de 15 inliers y una razón de inliers de al menos 0.25. La combinación de la puntuación de matching y el valor de confianza “ponderado sobre similitud coseno, razón de inliers e inliers absolutos” determina el nodo definitivo. Adicionalmente, una clasificación semántica cero-disparo basada en CLIP enriquece el resultado con una etiqueta de lugar.

El resultado final integra el identificador del nodo, su posición tridimensional (x, y, z), la orientación en grados (yaw), el valor de confianza y la etiqueta semántica del lugar. La conexión entre ambas fases se

establece a través de dos flujos de datos persistentes: el índice de embeddings FAISS, empleado en la etapa de recuperación profunda, y el índice de nodos con imágenes, utilizado durante el matching visual.

El siguiente diagrama describe de manera secuencial este proceso completo, la estructura general de la arquitectura y la interacción entre sus distintos módulos. Esta organización modular permite obtener una solución reproducible, coherente y aplicable en entornos interiores reales, independiente del hardware de captura y extensible a distintos métodos de matching visual.

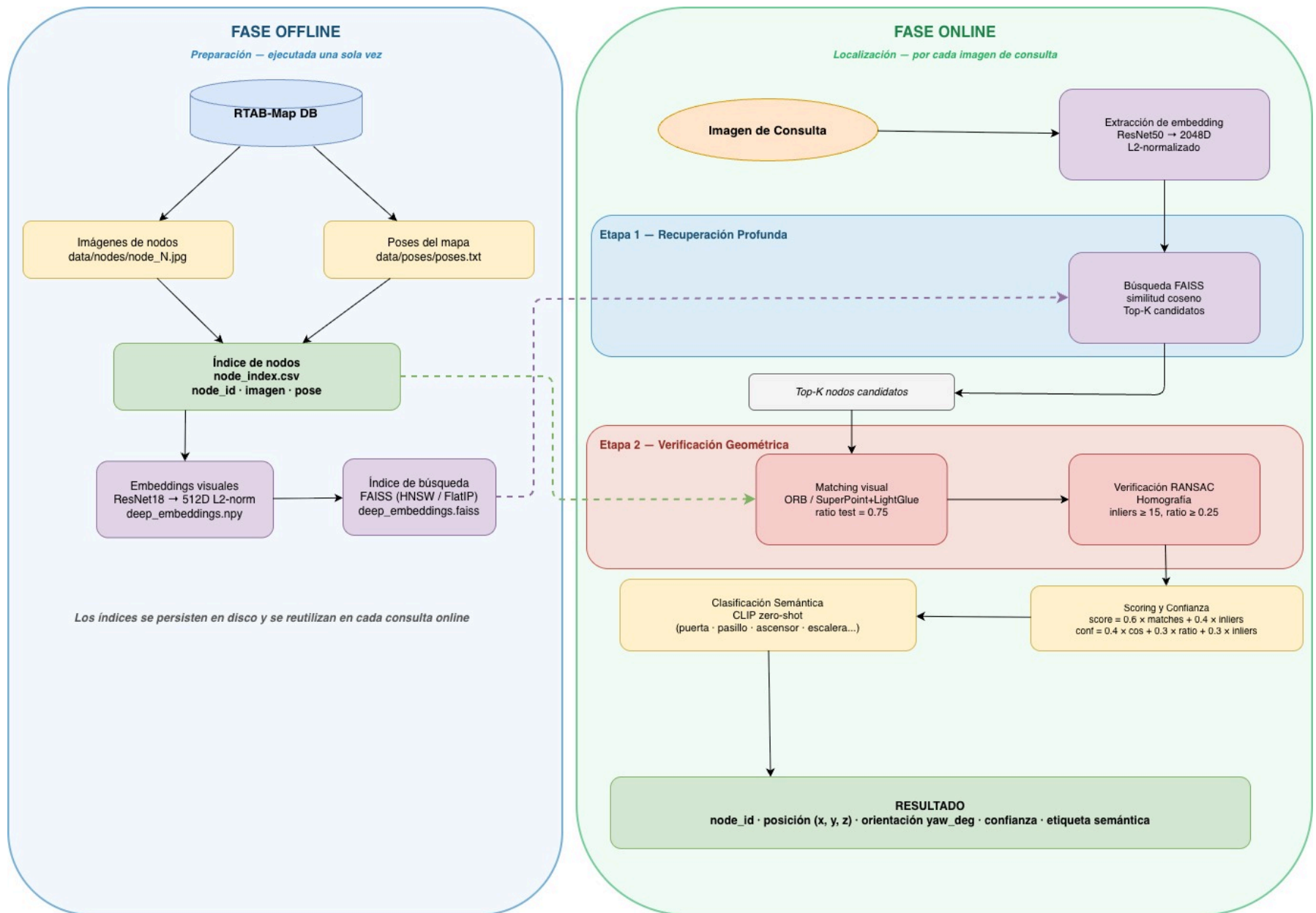


Figura 1: Fase Offline - Fase Online

4.1. Metodología de trabajo elegida

De acuerdo con lo expuesto en el capítulo de Marco Teórico-Conceptual, la implementación de un sistema de localización indoor en escenarios reales demanda una metodología iterativa, ya que tanto la calidad del mapa como la estabilidad de la pose estimada se ven influenciadas por la adquisición de datos, la parametrización del sistema y las condiciones del entorno. Por esta razón, en el presente trabajo se adopta una metodología de prototipado incremental, cercana a los enfoques ágiles, pero adaptada al contexto de investigación aplicada.

Una de las decisiones centrales de diseño es la incorporación de técnicas de aprendizaje profundo (deep learning) como núcleo del proceso de localización. A diferencia de los enfoques clásicos basados exclusivamente en descriptores locales de bajo nivel, el aprendizaje profundo permite extraer representaciones visuales de alto nivel que capturan la apariencia global de un lugar con mayor robustez ante cambios de iluminación, variaciones de punto de vista y oclusiones parciales. Esta capacidad resulta especialmente relevante en entornos indoor, donde la repetición de texturas y la escasez de características geométricas distintivas limitan el desempeño de los métodos tradicionales.

La metodología adoptada se articula mediante ciclos cortos que integran cuatro fases recurrentes: definición, implementación, validación y mejora continua. Este esquema iterativo ofrece ventajas concretas para un proyecto de investigación aplicada: permite incorporar retroalimentación cuantitativa en cada ciclo, detectar tempranamente degradaciones en la calidad de localización y ajustar progresivamente tanto los hiperparámetros del sistema como la arquitectura del modelo sin comprometer los avances previos. El resultado de cada ciclo no es únicamente un conjunto de componentes de software funcionales, sino también evidencia medible de desempeño que orienta las decisiones del ciclo siguiente.

En este marco, la metodología de trabajo comprende las siguientes etapas:

1. Definición de requisitos y criterios de éxito. Se establecen las restricciones del sistema, los criterios de calidad de la localización estimada y las métricas que guiarán la evaluación en cada ciclo. Esta etapa garantiza que cada iteración avance hacia objetivos verificables y no hacia mejoras subjetivas.
 2. Construcción del mapa base con RTAB-Map. A partir de recorridos realizados en el entorno real, RTAB-Map genera una base de datos que contiene las imágenes de cada nodo del mapa junto con su posición y orientación (pose) asociada. Esta información constituye la memoria visual del sistema: el conjunto de referencia contra el cual se compara toda imagen de consulta durante la localización.
 3. Habilitación de las funciones de localización y relocalización mediante aprendizaje profundo. Con el dataset consolidado, se construyen los índices que permiten la localización eficiente en tiempo de consulta. El componente central de esta etapa es la generación de representaciones vectoriales (embeddings) de cada imagen de nodo mediante una red neuronal convolucional profunda (ResNet50 pre-entrenada). Cada imagen queda representada como un vector de 2048 dimensiones en un espacio métrico donde la similitud entre vectores refleja la similitud visual entre lugares. Estos vectores se organizan en un índice de búsqueda vectorial implementado con FAISS, que permite recuperar eficientemente los nodos visualmente más similares a una imagen de consulta.
- La ventaja fundamental de este enfoque frente a métodos puramente clásicos reside en que la red neuronal actúa como un extractor de características semánticas entrenado sobre millones de imágenes: reconoce estructuras visuales de alto nivel “disposición espacial, texturas globales, patrones de iluminación” que los descriptores locales como ORB no pueden capturar. Esto se traduce en una etapa de recuperación

más robusta y en una reducción significativa del espacio de búsqueda antes de la verificación geométrica.

4. Extracción de la pose estimada y su transformación a 2D. Ante una imagen de consulta, el sistema opera en dos etapas complementarias. En la primera, el módulo de recuperación profunda identifica los nodos del mapa más similares visualmente mediante comparación en el espacio de embeddings. En la segunda, cada candidato es sometido a una verificación geométrica mediante estimación de homografía con RANSAC, que confirma si la correspondencia visual es coherente con una relación espacial válida entre la imagen de consulta y el nodo candidato. La pose 3D del nodo confirmado se proyecta posteriormente al plano 2D para su representación en el plano de planta del entorno.
5. Visualización y publicación de resultados. Los resultados de localización “pose estimada, nivel de confianza, nodo identificado y posición en el plano de planta” se publican a través de una interfaz de servicio que permite su integración con otros sistemas. Adicionalmente, el sistema incorpora un módulo de clasificación semántica basado en CLIP que asocia la imagen de consulta con una etiqueta descriptiva del tipo de lugar (pasillo, sala de reuniones, laboratorio, etc.), enriqueciendo la información contextual de la localización.
6. Evaluación del sistema, ajuste de parámetros y documentación. Esta etapa cierra cada ciclo iterativo con evidencia cuantitativa. Se miden la tasa de localización correcta, la distribución de confianza y los errores de pose sobre conjuntos de consulta representativos del entorno real. Las localizaciones con baja confianza o resultado incorrecto se canalizan hacia un proceso de retroalimentación que informa el siguiente ciclo: ajuste de umbrales geométricos, selección del descriptor de características más adecuado o, en caso necesario, actualización del mapa base. Esta capacidad de retroalimentación sistemática es una de las ventajas distintivas del enfoque incremental adoptado frente a metodologías de diseño lineal.

4.2. Levantamiento de requerimientos

Como se expuso en el capítulo de Problema y Justificación, el sistema está orientado a habilitar la localización indoor con potencial de aplicación en contextos de accesibilidad. Este propósito implica no solo que el sistema sea capaz de determinar la posición de un usuario a partir de una imagen, sino también que lo haga de manera confiable, en tiempos de respuesta adecuados para su uso en tiempo real y a partir de datos que puedan ser obtenidos sin infraestructura especializada adicional.

Para capturar esta diversidad de necesidades de forma estructurada, el levantamiento de requerimientos se organiza en tres bloques complementarios. El primero define los requerimientos funcionales, es decir, las capacidades concretas que el sistema debe ofrecer: desde la localización de una imagen de consulta hasta la exposición de resultados mediante una interfaz de servicio. El segundo establece los requerimientos no funcionales, que fijan los atributos de calidad que condicionan la viabilidad de la solución: precisión de la pose estimada, robustez ante variaciones de iluminación y ángulo de captura, escalabilidad ante mapas de mayor tamaño y tiempos de respuesta aceptables. El tercero delimita los requerimientos de datos, que describen las características que deben poseer las imágenes y la información espacial empleadas tanto en la construcción del mapa como en la fase de consulta, dado que la calidad de las representaciones

aprendidas por el modelo de deep learning depende directamente de la calidad y diversidad de los datos de entrada.

Esta estructura de tres bloques permite abordar el sistema desde dimensiones distintas pero interdependientes, y sienta las bases para las decisiones de diseño e implementación que se desarrollan en los capítulos siguientes.

a. Requerimientos funcionales

Los requerimientos funcionales definen las capacidades que el sistema debe implementar para cumplir con el objetivo de localización indoor. En este contexto, la solución propuesta debe ser capaz de generar un mapa indoor de referencia mediante RTAB-Map a partir de un recorrido previo por el entorno, extrayendo de forma automática las imágenes de cada nodo del mapa junto con su posición y orientación (pose) asociada.

A partir de dicho mapa, el sistema debe construir una representación visual compacta de cada nodo mediante técnicas de aprendizaje profundo, de modo que cada lugar quede codificado como un vector numérico que capture su apariencia global. Esta representación debe organizarse en un índice de búsqueda que permita la recuperación eficiente de los nodos visualmente más similares ante imágenes de consulta, utilizando una estrategia de indexación adecuada al tamaño del mapa.

Sobre este índice, el sistema debe ejecutar el proceso de localización en dos etapas ante una imagen o frame de consulta: primero, identificar un conjunto reducido de nodos candidatos mediante comparación de similitud en el espacio de representaciones aprendidas; y segundo, verificar geoméricamente cada candidato para confirmar la coherencia espacial de la correspondencia. El sistema debe reportar el nodo de mayor correspondencia junto con un valor de confianza numérico que refleje la calidad de la localización estimada, permitiendo al usuario o al sistema consumidor discriminar resultados fiables de resultados inciertos.

Adicionalmente, el sistema debe ser capaz de clasificar semánticamente el tipo de lugar capturado en la imagen de consulta; por ejemplo, pasillo, sala o área común, como información contextual complementaria a la pose estimada.

Una vez determinada la coincidencia más probable, el sistema debe recuperar la pose asociada al nodo identificado, incluyendo tanto la posición tridimensional como la orientación del dispositivo. Dicha pose debe proyectarse sobre una representación bidimensional del entorno plano de planta de manera que la ubicación estimada se visualice mediante coordenadas en píxeles y ángulo de orientación.

El sistema debe exponer los resultados a través de una interfaz de servicio que acepte una imagen de consulta y devuelva, en formato estructurado, los valores de posición (x, y), orientación (yaw), identificador del nodo asociado, nivel de confianza y etiqueta semántica del lugar. Esta interfaz debe permitir la consulta del mapa completo de nodos proyectados sobre el plano de planta, para proveer una visión global del entorno mapeado.

Finalmente, el sistema debe posibilitar la repetición del proceso con distintas imágenes de consulta, a fin de evaluar su funcionamiento en múltiples escenarios y condiciones de entrada, y debe ofrecer mecanismos de evaluación comparativa que permitan medir el desempeño del sistema ante variaciones en los parámetros de localización.

b. Requerimientos no funcionales

Los requerimientos no funcionales establecen las condiciones de calidad y operación que debe cumplir el sistema para garantizar un desempeño confiable y consistente. En este trabajo se consideran los siguientes:

Reproducibilidad: el flujo de procesamiento debe poder ejecutarse repetidamente sobre el mismo conjunto de datos, manteniendo resultados consistentes y comparables en cada ejecución. Para ello, todos los parámetros relevantes del sistema “umbrales de verificación geométrica, número de candidatos recuperados, modelo de extracción de características” deben estar centralizados en un único archivo de configuración, de forma que cualquier modificación sea explícita, versionable y trazable.

Tiempo de respuesta razonable: el sistema debe ser capaz de realizar la estimación de la localización y generar la salida correspondiente en un intervalo de tiempo de pocos segundos, de modo que su funcionamiento resulte viable en escenarios de uso práctico. Este requerimiento se satisface mediante una etapa de recuperación vectorial que selecciona un subconjunto reducido de candidatos visualmente similares. Para colecciones pequeñas, como el mapa evaluado de 1 131 nodos, se utiliza IndexFlatIP, que realiza una búsqueda exacta sobre los embeddings. Posteriormente, la verificación geométrica se ejecuta únicamente sobre los candidatos recuperados, evitando aplicar el matching local y RANSAC contra la totalidad de los nodos del mapa.

Robustez ante variaciones visuales: dado que el sistema opera en entornos reales donde las condiciones de iluminación, el ángulo de captura y las oclusiones parciales pueden variar entre el recorrido de mapeo y el momento de consulta, el módulo de localización debe mantener un desempeño aceptable ante estas perturbaciones. Esta propiedad se sustenta en el uso de representaciones aprendidas mediante redes neuronales profundas, que capturan características semánticas de alto nivel con mayor invarianza a estas variaciones en comparación con los descriptores locales clásicos.

Escalabilidad: el sistema debe mantener tiempos de respuesta aceptables a medida que el mapa crece en número de nodos. La estrategia de indexación debe adaptarse automáticamente al tamaño del mapa, empleando búsqueda exacta mediante IndexFlatIP para colecciones pequeñas y estructuras de búsqueda aproximada, como IndexHNSWFlat, para mapas de mayor escala, sin requerir intervención manual del operador.

Estabilidad: la solución debe minimizar variaciones abruptas en la localización estimada entre consultas consecutivas espacialmente cercanas, procurando que la discrepancia se mantenga, idealmente, en un rango aproximado de 1 a 2 metros respecto al nodo de referencia. La etapa de verificación geométrica mediante RANSAC contribuye a este requerimiento al rechazar correspondencias visualmente similares, pero geométricamente inconsistentes, reduciendo la tasa de localizaciones espurias.

Trazabilidad: el sistema debe registrar de manera sistemática los parámetros relevantes de ejecución, la versión del mapa empleada y los registros de eventos generados durante el proceso, con el fin de facilitar el análisis, la validación y la depuración de resultados. Asimismo, el valor de confianza asociado a cada resultado de localización debe quedar disponible en la salida del sistema, permitiendo identificar consultas con baja certeza para su revisión posterior.

Portabilidad: la implementación debe poder ejecutarse en entornos Linux/Ubuntu, con sus dependencias claramente especificadas y documentadas para favorecer su instalación, mantenimiento y replicación. El sistema debe soportar tanto ejecución sobre CPU como sobre GPU, adaptando automáticamente el dispositivo de cómputo disponible sin requerir modificación del código ni de la configuración por parte del usuario.

4.3. Arquitectura de localización

Como se señaló en el capítulo de Estado del Arte, numerosos sistemas de localización indoor estructuran su funcionamiento en dos etapas diferenciadas: la construcción del mapa y la localización dentro de dicho mapa. En este proyecto se adopta esta misma separación conceptual y operativa, denominadas fase offline y fase online respectivamente, ya que permite mantener mayor claridad en el diseño del sistema, desacoplar los procesos de mayor costo computacional de aquellos que operan en tiempo de consulta, y facilitar la actualización del mapa sin afectar el módulo de localización.

Bajo este enfoque, la arquitectura propuesta se organiza en componentes claramente definidos, cada uno con una función específica dentro del flujo general. El Diagrama de Casos de Uso (Figura 3) presenta una visión funcional del sistema desde la perspectiva de sus actores: el técnico u operador, responsable de ejecutar la fase offline mediante los cuatro casos de uso marcados en la banda superior (captura del recorrido, extracción de imágenes y poses, generación de representaciones profundas y construcción del índice de búsqueda); el cliente de la API, que invoca los tres servicios de la fase online (localizar imagen de consulta, consultar mapa completo de nodos y verificar estado del servicio); y el usuario final, que también puede iniciar directamente el proceso de localización. Este diagrama permite identificar, además, las relaciones de inclusión que descomponen el caso de uso principal Localizar imagen de consulta en sus cuatro subprocesos constitutivos: recuperación de candidatos por similitud profunda, verificación geométrica con homografía RANSAC, clasificación semántica del lugar y proyección de la pose al plano 2D.

Fase offline: construcción del mapa y generación del índice visual. Esta fase se ejecuta una única vez por entorno y sus resultados se reutilizan en todas las consultas posteriores. En primer lugar, el técnico u operador recorre el entorno con una cámara RGB-D y RTAB-Map procesa esta secuencia para construir un grafo de poses optimizadas mediante cierre de ciclos, generando como salida una base de datos visual en formato SQLite que contiene la imagen RGB y la posición y orientación (pose) de cada nodo del mapa.

A partir de esta base de datos, el módulo de preparación del dataset extrae las imágenes de los nodos y sus poses en seis grados de libertad (posición y orientación), conformando el conjunto de referencia del

sistema. Sobre este conjunto, el módulo de aprendizaje profundo “representado en el Diagrama de Clases (Figura 3) por la clase DeepEmbeddingExtractor, que opera sobre una red ResNet50 sin su capa final de clasificación” procesa cada imagen generando una representación vectorial de 2048 dimensiones que codifica la apariencia global del lugar. Estas representaciones se organizan en un índice de búsqueda a cargo de la clase EmbeddingSearchIndex, que selecciona automáticamente entre una estructura IndexFlatIP para colecciones de hasta 5 000 nodos y una estructura IndexHNSWFlat para colecciones de mayor tamaño, permitiendo la recuperación eficiente de los nodos visualmente más similares a cualquier imagen de consulta. El resultado de esta fase es un índice visual compacto que constituye la memoria permanente del sistema.

Fase online: localización ante una imagen de consulta. Ante una nueva imagen de consulta, el sistema ejecuta un proceso de localización en dos etapas complementarias, tal como se refleja en las relaciones de la Figura 2. En la primera etapa “Recuperar candidatos (deep retrieval)” el módulo de recuperación profunda proyecta la imagen de consulta al mismo espacio vectorial que los nodos del mapa y recupera, mediante comparación de similitud coseno en el índice FAISS, los cinco nodos candidatos más próximos. Esta etapa descarta la mayor parte del mapa de forma eficiente, aprovechando la capacidad de las representaciones aprendidas para capturar similitudes visuales de alto nivel con robustez ante variaciones de iluminación y punto de vista.

En la segunda etapa “Verificar geometría (RANSAC homografía)” cada nodo candidato es sometido a una comparación local de características mediante la clase ClassicalMatcher, que soporta los métodos ORB, GFTT+ORB y SuperPoint+LightGlue. Los emparejamientos obtenidos son validados por GeometricVerification, que aplica RANSAC sobre una homografía exigiendo un umbral de reproyección de 5.0 px, un mínimo de 15 inliers y una razón de inliers de al menos 0.25. Este paso confirma si la correspondencia visual entre la imagen de consulta y el nodo candidato es coherente con una relación espacial válida, filtrando coincidencias superficiales pero geométricamente inconsistentes. El sistema selecciona el nodo con mayor puntuación de verificación como resultado y calcula un valor de confianza que pondera la similitud en el espacio de representaciones profundas y la calidad geométrica de la verificación.

Adicionalmente, el módulo de clasificación semántica “clase SemanticClassifier en la Figura 3, basada en el modelo CLIP con ocho etiquetas de lugar interior” analiza la imagen de consulta para asignarle una etiqueta descriptiva (pasillo, sala, área común, entre otras) como información contextual complementaria.

Capa de proyección 2D y servicio. La pose tridimensional del nodo identificado se transforma mediante una proyección afín al sistema de coordenadas en píxeles del plano de planta del entorno, obteniéndose la posición (u, v) y la orientación (yaw) del usuario sobre el mapa 2D. Finalmente, la capa de servicio “implementada por APILocalizationService, el orquestador central visible en la Figura 3” expone toda esta funcionalidad mediante tres métodos REST: localize, que acepta una imagen de consulta y devuelve la pose estimada, el nivel de confianza, la etiqueta semántica y la posición sobre el plano de planta; get_all_nodes, que proporciona el mapa completo de nodos de referencia; y health, que permite verificar el estado operativo del servicio. Esta correspondencia entre los métodos de la clase y los casos de uso de la Figura 2 refleja el diseño orientado a servicios que favorece la integración del sistema con otras aplicaciones y su uso en escenarios reproducibles.

La estructura interna del software que soporta esta arquitectura "las clases principales, sus responsabilidades y las relaciones de composición y dependencia entre ellas" se presenta en el Diagrama de Clases (Figura 3). Este diagrama permite apreciar cómo `APILocalizationService` actúa como orquestador central, delegando la recuperación profunda y la verificación de candidatos en `HybridLocalizerRuntime`, la comparación de características en `ClassicalMatcher` y `GeometricVerification`, y la clasificación contextual en `SemanticClassifier`, todos ellos parametrizados de forma centralizada a través de `AppConfig`.

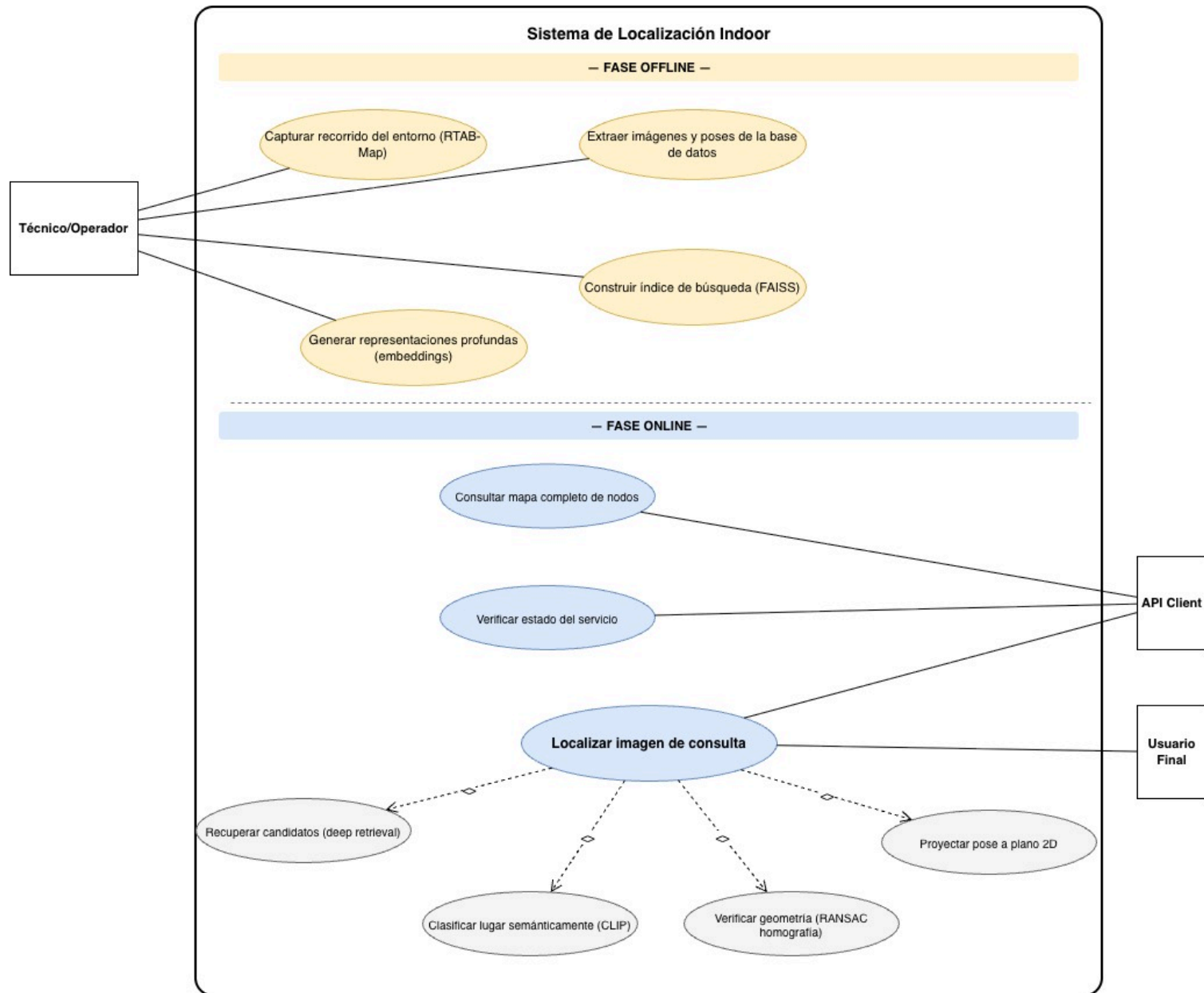


Figura 2: Caso de Uso

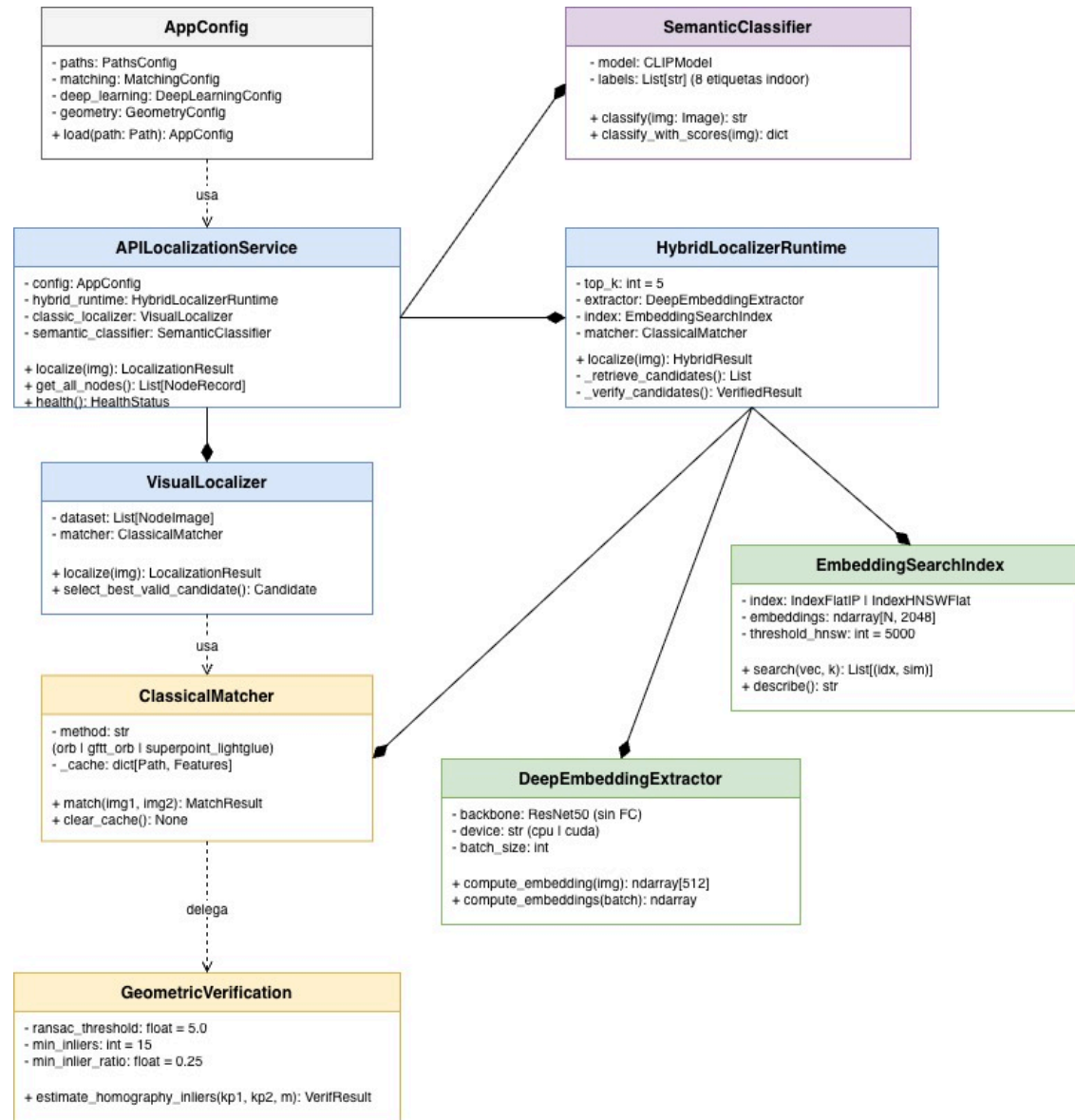


Figura 3: Diagrama de Clases

4.4. Proceso detallado

a. Preparación del entorno y herramientas

Para la ejecución del sistema de localización indoor propuesto se requiere la instalación y configuración de un conjunto de herramientas que cubren tanto la etapa de construcción del mapa como la de localización y publicación de resultados. El entorno de desarrollo y despliegue está basado en Linux/Ubuntu, plataforma sobre la cual se han validado todas las dependencias del sistema.

La herramienta principal para la construcción del mapa es RTAB-Map (Real-Time Appearance-Based Mapping), un sistema de SLAM visual que opera con cámaras RGB-D y genera como salida una base de datos en formato SQLite con las imágenes y poses de cada nodo del mapa. RTAB-Map puede ejecutarse tanto como aplicación de escritorio independiente como integrado en el ecosistema ROS, y su base de datos constituye el único insumo externo que el sistema de localización requiere.

El módulo de localización está implementado en Python 3, apoyándose en las siguientes bibliotecas principales:

- **PyTorch**: para la extracción de representaciones visuales profundas mediante una arquitectura ResNet50 preentrenada. El sistema detecta automáticamente la disponibilidad de GPU y ajusta el dispositivo de cómputo sin intervención del usuario.
- **FAISS (Facebook AI Similarity Search)**: para la construcción y consulta del índice de búsqueda aproximada sobre las representaciones vectoriales de los nodos del mapa.
- **OpenCV**: para la detección y descripción de puntos de interés (ORB, GFTT), el emparejamiento de características y la estimación de homografía mediante RANSAC en la etapa de verificación geométrica.
- **Transformers (HuggingFace)**: para el acceso a los modelos SuperPoint, LightGlue y CLIP empleados respectivamente en la verificación geométrica avanzada y la clasificación semántica de lugares.
- **FastAPI y Uvicorn**: para la exposición del sistema como servicio REST, permitiendo su integración con otras aplicaciones mediante llamadas HTTP estándar.
- **NumPy y Pillow**: para el procesamiento y transformación de imágenes y vectores numéricos a lo largo del pipeline.

La configuración del sistema se centraliza en un único archivo YAML que agrupa los parámetros de todas las etapas: rutas del dataset, umbral RANSAC, número de candidatos a recuperar, dispositivo de cómputo, modelo de descriptores y parámetros de proyección 2D, entre otros. Este diseño permite reproducir exactamente cualquier ejecución con solo conservar dicho archivo, y facilita la comparación de configuraciones sin modificar el código del sistema.

Antes de ejecutar el flujo de localización, el entorno debe satisfacer los siguientes requisitos: disponer de una base de datos RTAB-Map generada a partir del entorno objetivo, contar con las dependencias Python instaladas según el manifiesto de requisitos del proyecto, y tener acceso a los modelos de red neuronal necesarios que pueden descargarse automáticamente en la primera ejecución o de forma anticipada mediante el script de preparación incluido en el proyecto.

b. Construcción del mapa base (RTAB-Map)

La construcción del mapa base constituye la primera etapa de la fase offline y proporciona la referencia espacial sobre la cual el sistema realiza posteriormente la estimación de posición y orientación. Para esta fase se emplea RTAB-Map (Real-Time Appearance-Based Mapping), un sistema de SLAM visual-inercial capaz de generar representaciones consistentes del entorno a partir de un recorrido de captura, y cuya integración con ROS lo hace compatible con una amplia variedad de sensores RGB-D y configuraciones de hardware.

Es importante destacar que RTAB-Map interviene exclusivamente en esta etapa offline: una vez generada la base de datos del entorno, todo el procesamiento posterior de localización, extracción de características y estimación de pose es realizado íntegramente por el módulo Python desarrollado en este proyecto, sin requerir que RTAB-Map permanezca en ejecución durante la fase de consulta.

Recorrido de captura y configuración del sistema. El procedimiento inicia con la configuración de RTAB-Map en modo de mapeo sobre una plataforma Linux/Ubuntu, definiendo los parámetros de adquisición y procesamiento relevantes para el entorno interior. En esta etapa se establece el tipo de odometría empleada visual, RGB-D o LiDAR, según el sensor disponible, la frecuencia de procesamiento de fotogramas, el criterio de selección de keyframes y la resolución de representación del mapa. Asimismo, se habilita el mecanismo de cierre de ciclos (loop closure), indispensable para detectar revisitas a zonas previamente recorridas y corregir el error acumulado en la trayectoria estimada antes de que se propague a los nodos del mapa.

Una vez configurado el sistema, se ejecuta el recorrido de captura dentro del edificio. Durante este proceso, RTAB-Map registra la información sensorial y construye internamente un grafo de poses: cada nodo del grafo corresponde a un keyframe representativo del recorrido y almacena la imagen RGB capturada en ese instante junto con las restricciones espaciales que describen su relación con los nodos vecinos. Esta estructura de grafo permite modelar de forma consistente tanto la apariencia visual del entorno como la evolución de la trayectoria del dispositivo.

Optimización del grafo y exportación. Una vez completado el recorrido, RTAB-Map realiza la optimización global del grafo con el objetivo de distribuir y minimizar el error acumulado, obteniendo poses finales más precisas para cada nodo. Como resultado de esta optimización, el sistema consolida toda la información en una base de datos SQLite (archivo .db) que constituye la salida principal de esta etapa. Dicha base de datos contiene, para cada nodo del mapa, la imagen RGB asociada en formato comprimido y su posición y orientación (pose) en seis grados de libertad “posición (x, y, z) y orientación en cuaternión (qx, qy, qz, qw)”, representando el recorrido completo del dispositivo a lo largo del entorno mapeado.

Adicionalmente, RTAB-Map puede exportar un mapa 2D de ocupación y una nube de puntos 3D, útiles para visualización o navegación autónoma. Sin embargo, para los fines del presente sistema de localización, el insumo relevante es exclusivamente la base de datos SQLite, ya que de ella se extraen en la etapa siguiente las imágenes de nodos y las poses optimizadas que conformarán el índice visual de referencia.

c. Localización / re-localización sobre el mapa

En esta etapa se realiza la localización o re-localización del usuario sobre el mapa previamente construido, a partir de una imagen de consulta capturada en el entorno interior. El propósito es determinar la correspondencia más probable entre la observación actual y las imágenes de referencia almacenadas en el índice visual, con el fin de estimar la posición del dispositivo y su pose dentro del sistema de coordenadas global del entorno. A diferencia de lo que podría suponerse, esta etapa no utiliza RTAB-Map: toda la localización es ejecutada íntegramente por el módulo Python desarrollado en este proyecto, empleando el índice visual generado durante la fase offline como única referencia.

El proceso de localización opera en dos etapas complementarias y secuenciales, cuya combinación permite equilibrar eficiencia computacional y precisión geométrica.

Primera etapa: recuperación por similitud profunda. Ante una imagen de consulta, el módulo de recuperación extrae su representación vectorial de 2048 dimensiones mediante la misma red neuronal convolucional (ResNet50) utilizada para indexar los nodos del mapa durante la fase offline. Este vector, normalizado por norma L2, es comparado contra todos los vectores del índice FAISS mediante similitud coseno, recuperando los $k = 10$ nodos visualmente más similares como candidatos a la localización. Esta etapa aprovecha la capacidad del modelo profundo para capturar similitudes visuales de alto nivel insensibles a variaciones menores de iluminación o punto de vista y reduce a diez el número de candidatos que posteriormente requieren verificación geométrica, disminuyendo el costo de la etapa de matching local.

Segunda etapa: verificación geométrica por homografía. Cada nodo candidato recuperado en la etapa anterior es sometido a una verificación geométrica local. Para ello, se detectan y describen puntos de interés en la imagen de consulta y en la imagen del nodo candidato mediante el descriptor configurado en el sistema por defecto SuperPoint con emparejamiento LightGlue, con soporte alternativo para ORB y GFTT+ORB. A partir de las correspondencias encontradas, se estima una homografía planar mediante RANSAC, con un umbral de Re proyección de 5.0 píxeles. Una correspondencia se considera geoméricamente válida únicamente si cumple simultáneamente dos condiciones: un número mínimo de 15 inliers y una razón de inliers de al menos 0.25. Este filtro descarta coincidencias que son visualmente similares pero geoméricamente inconsistentes, reduciendo significativamente la tasa de localizaciones erróneas.

Selección del resultado y cálculo de confianza. El sistema selecciona como resultado final el candidato que supera la verificación geométrica con mayor puntuación. Para cada candidato verificado se calcula un valor de confianza compuesto que integra tres señales de calidad, según se expresa en la Ecuación:

$$\text{confianza} = 0.4 \cdot \cos \theta + 0.3 \cdot r_{\text{inliers}} + 0.3 \cdot \widehat{N}_{\text{inliers}}$$

Donde $\cos \theta$ representa la similitud coseno obtenida en la etapa de recuperación profunda, r_{inliers} es la razón de inliers respecto al total de correspondencias encontradas, y $\widehat{N}_{\text{inliers}}$ es el número de inliers

normalizado. Esta fórmula pondera de forma equilibrada la calidad de la correspondencia visual global (capturada por el modelo de aprendizaje profundo) y la coherencia geométrica local (evaluada por RANSAC), produciendo un indicador de confiabilidad interpretable que permite discriminar resultados fiables de estimaciones inciertas.

En caso de que ningún candidato supere los umbrales de verificación geométrica, el sistema puede recurrir opcionalmente a un modo de localización clásica exhaustiva como mecanismo de respaldo. Este modo evalúa todos los nodos del mapa mediante la función de puntuación definida en la Ecuación y selecciona el nodo de mayor valor:

$$\text{score} = 0.6 \cdot N_{\text{matches}} + 0.4 \cdot N_{\text{inliers}}$$

donde N_{matches} es el número total de correspondencias encontradas y N_{inliers} es el subconjunto de correspondencias geoméricamente coherentes según el criterio RANSAC.

Salida del proceso. El resultado de la localización comprende: el identificador del nodo reconocido, la pose tridimensional asociada a dicho nodo, el valor de confianza calculado, y una etiqueta semántica del tipo de lugar generada por el módulo de clasificación CLIP como información contextual complementaria. Estos campos conforman la respuesta que el sistema entrega a la capa de servicio para su proyección 2D y publicación.

d. Extracción de pose y orientación

Como se indicó en el capítulo de Marco Teórico, la pose de un dispositivo en un entorno indoor se compone, de manera general, de su posición y su orientación. En esta propuesta se prioriza la estimación de la posición bidimensional (x, y) sobre el plano del edificio y de la orientación angular en el eje vertical (yaw), ya que estos dos parámetros permiten determinar tanto la ubicación del usuario como la dirección hacia la que apunta el dispositivo, constituyendo la información mínima suficiente para los propósitos de navegación y visualización indoor contemplados en este trabajo.

Origen de la pose. Las poses utilizadas durante la localización no se obtienen en tiempo de consulta desde RTAB-Map, sino que fueron extraídas de la base de datos SQLite durante la fase offline y almacenadas en un archivo tabular que relaciona cada identificador de nodo con su posición y orientación (pose) completa. Este archivo se carga en memoria al iniciar el sistema de localización, de modo que la recuperación de la pose de cualquier nodo se reduce a una consulta directa en una estructura indexada por identificador, sin introducir latencia adicional en el camino crítico de la localización.

Estructura de la pose almacenada. Cada registro de pose comprende siete valores numéricos: la posición tridimensional del nodo en el sistema de coordenadas global del mapa componentes tx, ty, tz y la orientación expresada como cuaternión unitario componentes qx, qy, qz, qw. Esta representación es la

producida por RTAB-Map tras la optimización del grafo y corresponde a la pose del dispositivo en el instante en que dicho nodo fue registrado durante el recorrido de captura.

Extracción del ángulo yaw. A partir del cuaternión de orientación se extrae el ángulo de rotación alrededor del eje vertical (yaw), que representa la dirección hacia la que apunta el dispositivo proyectado sobre el plano horizontal. La conversión se realiza mediante la relación estándar entre cuaterniones y ángulos de Euler:

$$\psi = \text{atan2}\left(2(q_w q_z + q_x q_y), 1 - 2(q_y^2 + q_z^2)\right)$$

donde ψ denota el ángulo yaw, y los términos q_w , q_x , q_y , q_z corresponden a las componentes del cuaternión de orientación del nodo identificado. El resultado es un ángulo en radianes que indica la rotación del dispositivo respecto al norte del sistema de coordenadas del mapa.

Resultado de la extracción. Al finalizar esta etapa, el sistema dispone de tres valores que describen completamente la localización del usuario en el plano: la posición tx y ty en el sistema de coordenadas tridimensional del mapa, y el ángulo ψ de orientación planar. Estos tres valores constituyen la entrada de la etapa siguiente, en la que se transforman al sistema de píxeles del plano de planta para su visualización.

e. Transformación y proyección a un plano 2D

El propósito de esta etapa es representar de manera clara y precisa la pose estimada del usuario o dispositivo sobre una imagen bidimensional correspondiente al plano de planta del entorno. Para ello se aplica una transformación afín que convierte las coordenadas tridimensionales del sistema de referencia del mapa RTAB-Map a coordenadas de píxeles sobre la imagen del plano, empleando Python y la biblioteca OpenCV para el cálculo y la renderización.

Definición de la transformación afín. La conversión entre el sistema de coordenadas métrico del mapa y el sistema de píxeles de la imagen se define mediante tres parámetros configurables:

- **Rotación (θ , en grados):** ángulo de alineación entre el sistema de coordenadas del mapa y la orientación del plano de planta en la imagen. Este parámetro compensa la diferencia de orientación entre el recorrido de captura y el norte del plano.
- **Escala (s , en píxeles/metro):** factor de conversión entre distancias métricas del mapa y distancias en píxeles de la imagen.
- **Traslación (Δu , Δv , en píxeles):** desplazamiento que sitúa el origen del sistema de coordenadas del mapa en la posición correcta dentro de la imagen del plano.

Estos tres parámetros se especifican en el archivo de configuración del sistema y deben calibrarse una vez por entorno, estableciendo la correspondencia geométrica entre el espacio físico y su representación digital.

Aplicación de la transformación. Dada la posición (tx, ty) del nodo identificado en el sistema de coordenadas del mapa, la proyección a píxeles se obtiene mediante la Ecuación X:

$$\begin{pmatrix} u \\ v \end{pmatrix} = s \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} t_x \\ t_y \end{pmatrix} + \begin{pmatrix} \Delta u \\ \Delta v \end{pmatrix}$$

son las coordenadas del punto proyectado en la imagen del plano. De forma análoga, el ángulo yaw extraído del cuaternión de orientación se incrementa en θ para expresar la dirección de orientación del dispositivo en el sistema de referencia de la imagen.

Esta misma transformación se aplica no solo al resultado de la consulta activa, sino también a la totalidad de los nodos del mapa, lo que permite generar una vista global del entorno con todos los puntos de referencia proyectados sobre el plano de planta.

Renderización con OpenCV. Sobre la imagen del plano de planta cargada como fondo, OpenCV realiza la visualización de la pose proyectada mediante tres elementos gráficos principales:

- Un círculo en la posición (u, v) que indica la ubicación estimada del dispositivo.
- Una flecha en la dirección del yaw proyectado, que representa la orientación del dispositivo sobre el plano.
- Una etiqueta de texto opcional con información complementaria, como el identificador del nodo asociado y el valor de confianza de la localización.

La imagen resultante, con todos estos elementos superpuestos, constituye la representación visual final de la localización estimada y es la que se publica como parte de la respuesta del servicio en la etapa siguiente.

f. Exposición del resultado (API / UI)

La exposición del resultado de la localización indoor se realiza mediante un servicio REST implementado con FastAPI, que estructura la funcionalidad del sistema en tres endpoints bien definidos. El Diagrama de Modos de Operación (Figura 4) sitúa este servicio como el Modo 3, que recibe como entrada las peticiones HTTP de clientes externos e invoca internamente el pipeline de localización descrito en el Modo 2. Esta separación hace explícito el hecho de que el servicio REST no implementa lógica de localización propia, sino que actúa como capa de exposición sobre el núcleo de localización, favoreciendo el desacoplamiento entre el módulo de estimación de pose y cualquier interfaz de presentación.

Endpoint principal: POST /localize. Este endpoint constituye el punto de entrada del flujo de localización en tiempo de consulta. Recibe como entrada una imagen de consulta enviada como archivo adjunto en una petición HTTP, la almacena de forma temporal y la procesa a través del pipeline completo cuya secuencia de transformaciones se muestra en el Diagrama de Flujo de Datos (Figura 5): extracción de representación profunda mediante ResNet50, recuperación de candidatos en el índice FAISS por similitud coseno, verificación geométrica mediante RANSAC con homografía, clasificación semántica con CLIP y

proyección afin al plano 2D. Esta última etapa de proyección, detallada en el Diagrama de Generación del Mapa 2D (Figura 6), convierte la pose tridimensional del nodo identificado en coordenadas de píxel mediante una transformación afin parametrizada por una rotación θ , una escala s (píxeles por metro) y un desplazamiento $(\Delta u, \Delta v)$, y calcula el ángulo de orientación yaw a partir del cuaternión de pose mediante la expresión $\arctan2$. La respuesta devuelta es un objeto JSON estructurado que incluye los siguientes campos:

`node_id`: identificador del nodo del mapa reconocido como más probable.

`pose`: pose tridimensional completa del nodo, con posición (tx, ty, tz) y orientación en cuaternión (qx, qy, qz, qw) .

`confidence`: valor de confianza compuesto, calculado como la ponderación de similitud coseno, razón de inliers y número de inliers normalizado.

`semantic_label`: etiqueta descriptiva del tipo de lugar asignada por el clasificador semántico (por ejemplo, pasillo, sala de reuniones o laboratorio).

`x, y`: coordenadas en píxeles de la posición estimada sobre el plano de planta, obtenidas mediante la proyección ilustrada en la Figura 6.

`yaw`: ángulo de orientación del dispositivo expresado en el sistema de referencia de la imagen del plano.

`map_image`: imagen del plano de planta con la pose proyectada superpuesta “un punto de posición y una flecha de orientación, tal como muestra la rama izquierda de la Figura 6”, codificada en base64 para su presentación directa en cualquier cliente sin necesidad de una petición adicional.

Endpoint de Mapa global: GET /map/all-nodes. Este endpoint devuelve una representación del plano de planta con la totalidad de los nodos del mapa proyectados sobre él, correspondiente a la rama derecha del Diagrama de Generación del Mapa 2D (Figura 6). Aprovecha directamente los artefactos persistidos durante el Modo 1 (Offline) de la Figura 4 “en particular el archivo de índice de nodos con sus poses completas” y aplica la misma proyección afin sobre los 1131 nodos del dataset, generando una imagen PNG que representa la distribución espacial de referencia del entorno. Es útil tanto para verificar la correcta calibración de los parámetros de proyección como para proporcionar contexto espacial en la interfaz de visualización.

Endpoint de estado: GET /health. Proporciona información sobre el estado operativo del servicio: disponibilidad del índice de embeddings, integridad del modelo de aprendizaje profundo, ruta del plano de planta activo y configuración en uso. Este endpoint permite verificar que el sistema está correctamente inicializado antes de enviar consultas de localización.

Interfaz de usuario. Los resultados expuestos por la API pueden consumirse desde cualquier cliente HTTP. En su versión mínima de validación, la interfaz de usuario permite cargar una imagen de consulta, enviarla al endpoint /localize y presentar de forma inmediata: la imagen del plano de planta con la posición y orientación estimadas superpuestas "generada en el último paso del flujo de la Figura 5 y construida según el proceso descrito en la Figura 6", los valores numéricos de coordenadas (x, y) y ángulo yaw, el nivel de confianza de la estimación y la etiqueta semántica del lugar identificado. Esta combinación de representación visual y datos cuantitativos facilita tanto la interpretación intuitiva del resultado como el análisis del desempeño del sistema en distintas condiciones de consulta, incluyendo su uso en el Modo 4 (Evaluación) representado en la Figura 4.

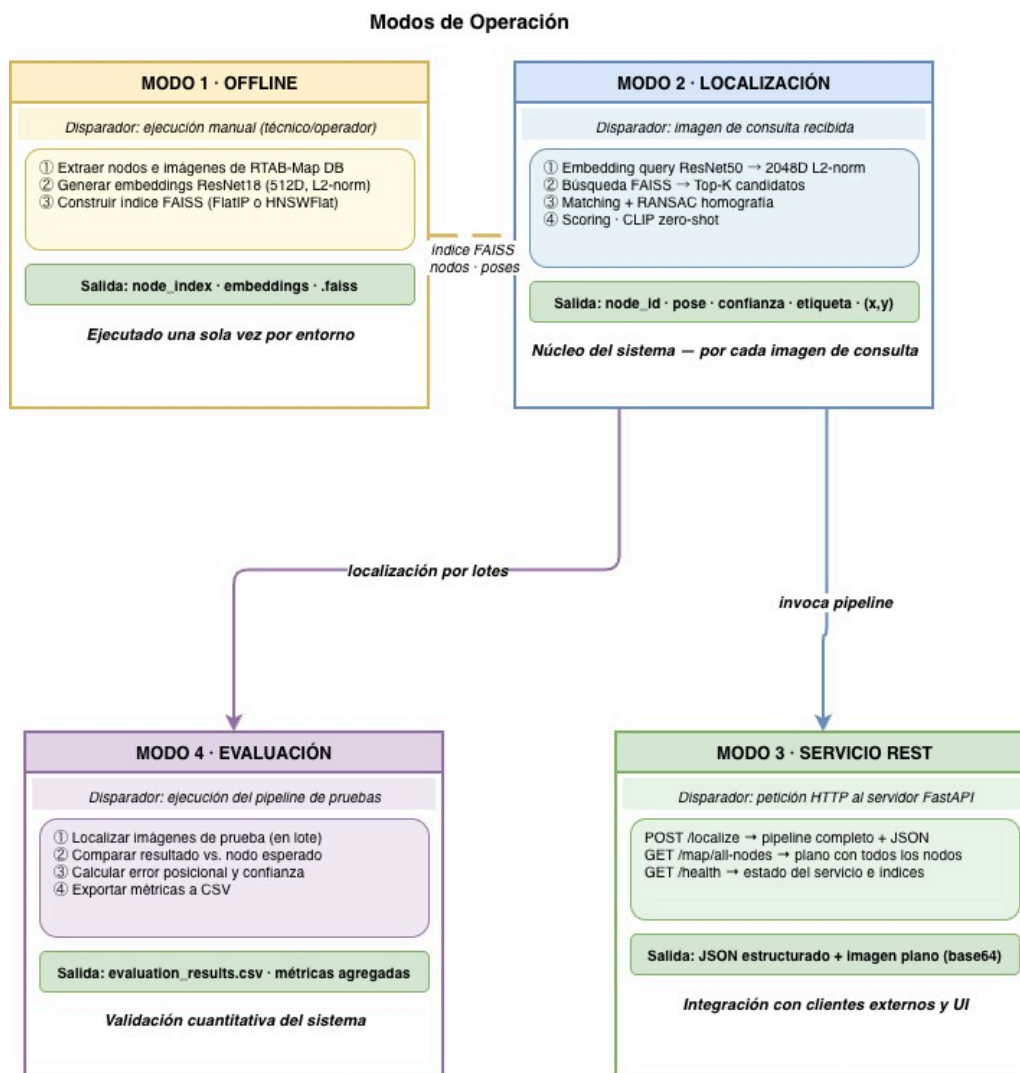


Figura 4: Modos de Operación

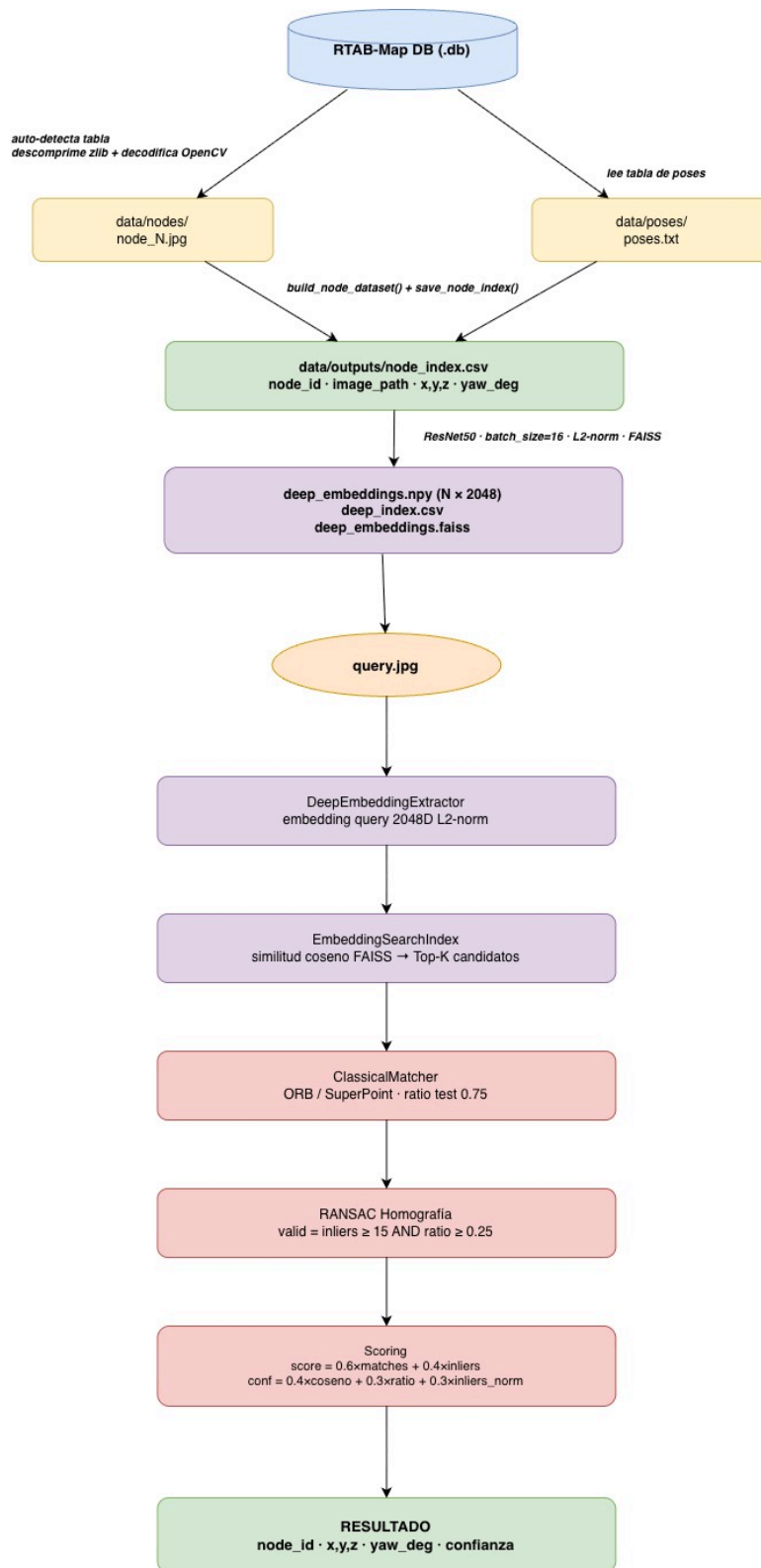


Figura 5: Flujo de Datos

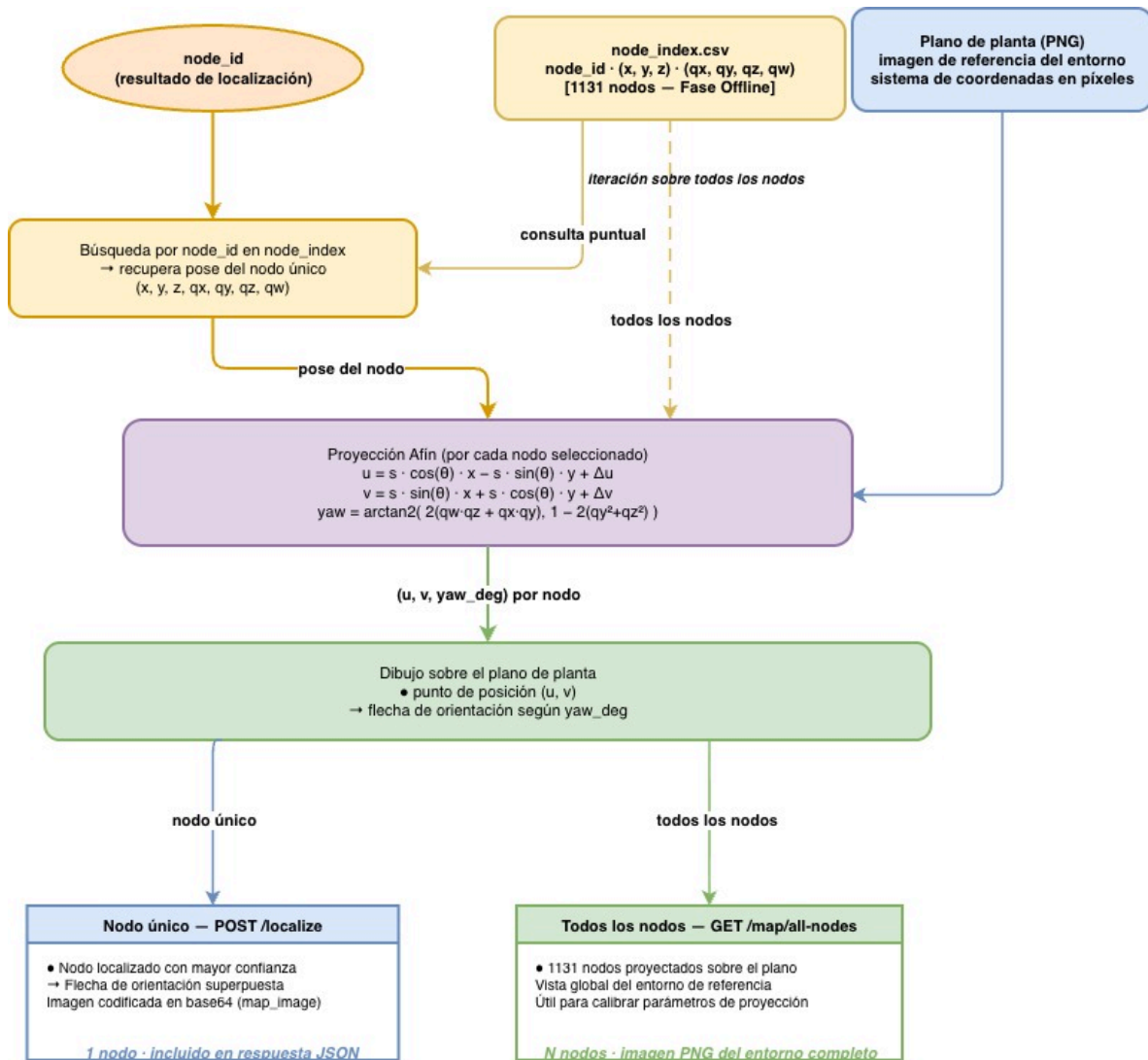


Figura 6: Flujo de Generación de Mapa 2D

4.5. Evaluación

Como se mencionó en el capítulo de Objetivos, la evaluación del sistema se realiza con medidas concretas y directamente interpretables, orientadas a verificar que el sistema cumple con los criterios de calidad establecidos en los requerimientos no funcionales. Dado que el entorno de prueba es un espacio indoor real sin sistema de posicionamiento de referencia de alta precisión, las métricas empleadas están diseñadas para ser calculables a partir de la información disponible en el propio mapa y de las poses optimizadas por RTAB-Map.

Las imágenes de consulta utilizadas en la evaluación se generan mediante dos estrategias complementarias: extracción de fotogramas del propio recorrido de mapeo simulando condiciones ideales y captura de nuevas imágenes del entorno en distintos momentos y condiciones de iluminación, con el fin de evaluar la robustez del sistema ante variaciones no presentes en el índice de referencia.

Tasa de éxito de localización. Se define como el porcentaje de consultas en las que el sistema produce una localización válida, entendida como aquella en la que el candidato seleccionado supera simultáneamente el umbral mínimo de inliers RANSAC (≥ 15) y la razón mínima de inliers (≥ 0.25). Esta métrica se calcula sobre el conjunto completo de consultas evaluadas y se desglosa por modo de localización híbrida y clásica de respaldo para cuantificar la contribución de cada etapa del pipeline. Adicionalmente, se registra la distribución del valor de confianza sobre las consultas exitosas, permitiendo identificar el rango de confianza habitual del sistema en el entorno evaluado.

Estabilidad espacial. Se evalúa verificando que consultas generadas en posiciones físicamente cercanas del entorno produzcan localizaciones en nodos del mapa igualmente próximos, sin saltos abruptos a zonas alejadas. Para ello se examina la distribución de distancias entre los nodos identificados en consultas consecutivas del recorrido de evaluación, considerando que el sistema es estable si la variación entre estimaciones vecinas se mantiene dentro del rango aproximado de 1 a 2 metros establecido en los requerimientos no funcionales.

Error aproximado de pose. En ausencia de un sistema de posicionamiento de referencia de alta precisión, el error de localización se estima comparando la pose del nodo identificado por el sistema con la pose del nodo más cercano físicamente a la posición real de captura de la imagen de consulta, cuando dicha información es conocida. Esta aproximación proporciona una cota superior del error real de localización. El error se expresa en metros como distancia euclídea entre las posiciones (x, y) del nodo estimado y del nodo de referencia en el sistema de coordenadas del mapa.

Tiempo de respuesta. Se mide el intervalo completo desde la recepción de la imagen de consulta hasta la entrega de la respuesta estructurada, incluyendo la extracción del embedding profundo, la búsqueda en el índice FAISS, la verificación geométrica de los candidatos y la proyección 2D. El objetivo es que este tiempo sea inferior a un umbral que haga viable el uso del sistema en escenarios prácticos. Se registra por separado el tiempo de la etapa de recuperación profunda y el de la verificación geométrica, dado que esta última escala con el número de candidatos verificados.

Comparación de descriptores de características. Se evalúa el impacto de la elección del descriptor local sobre la tasa de éxito y el tiempo de respuesta, comparando los resultados obtenidos con ORB y con

SuperPoint + LightGlue sobre el mismo conjunto de consultas etiquetadas. Esta comparación permite cuantificar la ganancia en precisión que aporta el descriptor neuronal frente al clásico, así como el costo adicional en tiempo de cómputo, informando la selección del descriptor más adecuado para cada contexto de despliegue.

Calidad del mapa base. Se evalúa mediante evidencia visual de la coherencia del recorrido de mapeo: visualización del grafo de poses sobre el plano del entorno, verificación de que las imágenes de nodos son nítidas y representativas de los distintos sectores del edificio, y comprobación de que no existen zonas del recorrido sin cobertura de nodos. Esta evaluación cualitativa complementa las métricas cuantitativas y permite detectar deficiencias en el dataset de referencia que puedan explicar caídas en la tasa de éxito de localización.

5. RESULTADOS Y DISCUSIÓN

En el presente capítulo se exponen los resultados obtenidos a partir del desarrollo e implementación del sistema de localización indoor basado en aprendizaje profundo descrito en los capítulos anteriores. Los resultados se organizan en torno a las etapas funcionales del sistema: la construcción del mapa base a partir de la base de datos generada con RTAB-Map, la preparación del índice visual de representaciones profundas, el desempeño del pipeline de localización híbrida ante distintas modalidades de consulta, y la exposición del resultado a través de la capa de servicio REST. Para cada etapa se reportan tanto los productos obtenidos como las observaciones relevantes derivadas de su ejecución sobre el entorno real de prueba.

Entorno de ejecución: hardware y software. Todas las pruebas documentadas en este capítulo se ejecutaron sobre un equipo portátil Alienware m16 R2 con sistema operativo Ubuntu 26.04 LTS, equipado con procesador Intel Core Ultra 7, 64 GB de memoria RAM y tarjeta gráfica NVIDIA GeForce RTX 4070 Laptop GPU con 8 GB de memoria de video. La versión del controlador NVIDIA instalada es la 595.58.03, con soporte para CUDA 13.2 a nivel de driver; el entorno de Python utiliza PyTorch 2.11.0 compilado contra CUDA 13.0 y cuDNN 9.1.9. La pila de software del sistema comprende además FAISS 1.13.2, OpenCV 4.13.0, FastAPI 0.136.1 y la biblioteca Transformers 5.8.0 para el módulo de clasificación semántica basado en CLIP. El proceso de extracción de representaciones profundas y el pipeline de localización se ejecutaron sobre la GPU (dispositivo CUDA), mientras que la construcción y persistencia del índice FAISS se realizaron en CPU según la configuración de la biblioteca. La Tabla 2 resume el entorno completo de ejecución.

Componente	Especificación / Versión
Equipo	Alienware m16 R2
Sistema operativo	Ubuntu 26.04 LTS
Procesador	Intel Core Ultra 7
Memoria RAM	64 GB
GPU	NVIDIA GeForce RTX 4070 Laptop GPU (8 GB VRAM)
Driver NVIDIA	595.58.03
CUDA (driver / PyTorch)	13.2 / 13.0
cuDNN	9.1.9
Python	3.14.4
PyTorch	2.11.0 + CUDA 13.0
FAISS	1.13.2
OpenCV	4.13.0
FastAPI / Uvicorn	0.136.1 / 0.46.0
Transformers (CLIP)	5.8.0
NumPy / Pandas	2.4.4 / 3.0.2

Tabla 2: Entorno de hardware y software utilizado en las pruebas

El conjunto de datos de referencia empleado en las pruebas principales se construyó a partir de una base de datos RTAB-Map generada mediante un recorrido RGB-D del entorno de prueba. El proceso de preparación del dataset, produjo un total de 1 131 nodos válidos, cada uno asociado a una imagen RGB de 720×960 píxeles y a su pose completa en seis grados de libertad (posición y orientación). El entorno cubre una región con coordenadas $X \in [-15.49, 8.24]$ m y $Y \in [-10.44, 3.19]$ m, lo que corresponde a un área de cobertura aproximada de 323 m², con variación vertical inferior a 0.22 m, confirmando que el entorno es esencialmente plano. El índice de búsqueda FAISS de tipo IndexFlatIP almacena los 1 131 vectores de características de 2048 dimensiones y garantiza exactitud total en la recuperación por similitud coseno sin recurrir a estructuras aproximadas.

Para la evaluación comparativa de descriptores de características se empleó el mismo mapa de referencia de 1 131 nodos, procesando cada imagen de consulta de forma independiente con ORB y con SuperPoint+LightGlue sobre el mismo conjunto.

Diseño experimental y modalidades de prueba. La evaluación del sistema a través de etapas de validación progresiva. Las primeras constituyeron pruebas de ajuste y calibración: una evaluación manual en distintos puntos del entorno (10 consultas) y una evaluación sintética con imágenes extraídas directamente del mapa con semilla aleatoria fija (20 consultas, ejecutada en paralelo con los descriptores ORB y SuperPoint+LightGlue). Estas etapas permitieron verificar el flujo completo del pipeline, identificar los

umbrales de confianza y rechazo adecuados, y establecer una línea base del comportamiento del sistema en condiciones controladas.

La evaluación que se reporta en los apartados siguientes constituye la prueba de validación central del sistema en condiciones de operación real: una evaluación cross-session con 490 consultas extraídas íntegramente de una base de datos RTAB-Map independiente. Esta base de datos corresponde a un recorrido distinto al del mapa de referencia, con origen de coordenadas propio y separación temporal de 18 días. El escenario replica la condición operativa más exigente contemplada en este trabajo: localizar imágenes provenientes de una sesión de captura diferente sin ningún conocimiento previo de la correspondencia geométrica entre ambos marcos de referencia. La evaluación se ejecutó en paralelo con los descriptores ORB y SuperPoint+LightGlue, y emplea como métrica de precisión el error espacial en metros calculado mediante alineación de poses por transformación rígida 2D (método de Procrustes SVD), lo que elimina la circularidad que resultaría de utilizar el mismo modelo de recuperación tanto para la predicción como para la asignación del ground truth.

Las métricas consideradas en la evaluación son: tasa de éxito de localización, error espacial en metros respecto al ground truth de pose, error de orientación en grados, tiempo de respuesta del pipeline completo desglosado por etapa, y valor de confianza compuesto. La discusión que acompaña a cada resultado busca interpretar los valores obtenidos en relación con los requerimientos establecidos en el capítulo de Levantamiento de Requerimientos, identificar las condiciones del entorno que influyen en el desempeño del sistema, y señalar las limitaciones observadas junto con las posibles líneas de mejora.

5.1. Resultados

En esta etapa se obtuvo un sistema funcional de relocalización visual indoor basado en imágenes extraídas de una base de datos RTAB-Map. El sistema permitió asociar una imagen de consulta con el nodo más probable del mapa y recuperar su posición y orientación estimada. Para validar su funcionamiento se ejecutaron tres modalidades de prueba de complejidad creciente: una evaluación manual con imágenes reales capturadas en distintos puntos del entorno (10 consultas), una evaluación sintética aleatoria para comparar descriptores de características (20 consultas) y, como prueba de validación central, una evaluación cross-session con 490 consultas reales provenientes de una base de datos RTAB-Map independiente. Los resultados de cada prueba se presentan a continuación.

Preparación del dataset

A partir de la base de datos RTAB-Map generada se extrajeron las imágenes asociadas a los nodos del mapa y sus poses correspondientes. El proceso de preparación produjo un conjunto de datos estructurado compuesto por 1 131 nodos, cada uno con una imagen RGB de 720×960 px y su pose completa en seis grados de libertad.

Elemento	Resultado obtenido
Imágenes de nodos extraídas	1131
Poses extraídas	1131
Imágenes con pose asociada	1131
Primer nodo registrado	1
Último nodo registrado	1131
Cobertura espacial X	[−15.49, 8.24] m (span 23.73 m)
Cobertura espacial Y	[−10.44, 3.19] m (span 13.62 m)
Área aproximada del entorno	~323 m ²
Variación vertical (eje Z)	< 0.22 m (entorno plano)

Tabla 3: Resumen del dataset generado desde RTAB-Map.

Como se muestra en la Tabla 3, todas las imágenes extraídas contaron con una pose asociada, sin registros faltantes. Esto garantizó que cada predicción del sistema pudiera relacionarse directamente con una coordenada espacial y una orientación dentro del mapa de referencia.

La Figura 7 ilustra el flujo seguido para transformar la base de datos RTAB-Map en un dataset utilizable. Este proceso comprendió la inspección automática del esquema de la base de datos, la extracción y descompresión de las imágenes de nodos, la decodificación de las poses en formato blob, la construcción del índice de nodos y la validación cruzada de correspondencia imagen–pose.

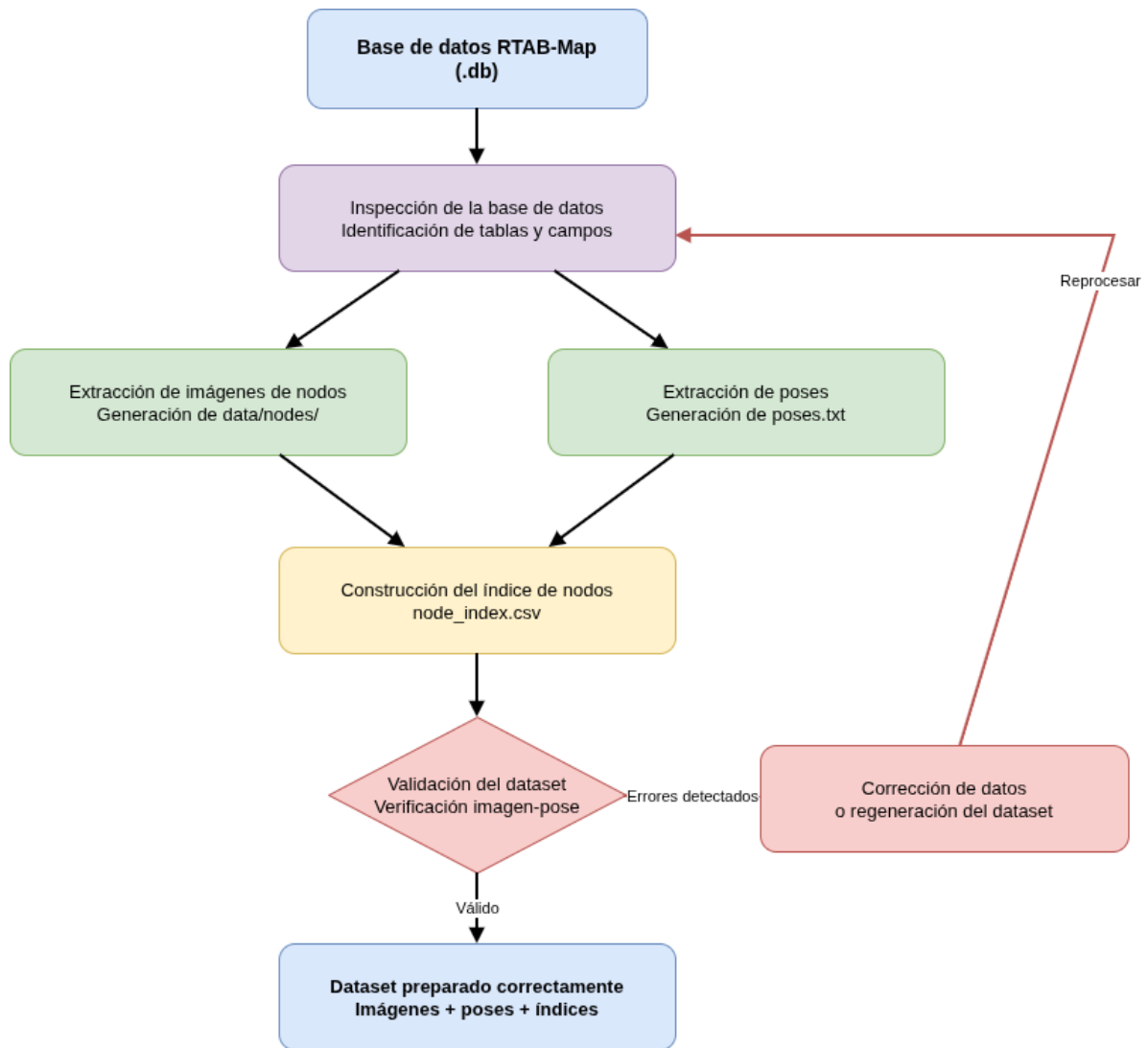


Figura 7: Flujo general de preparación del dataset.

Construcción del índice visual

Una vez preparado el dataset, se construyó el conjunto de artefactos necesarios para la etapa de localización. Este proceso generó un índice de referencia visual directo y un índice profundo basado en representaciones vectoriales de las imágenes, utilizado en la etapa de recuperación eficiente de candidatos.

Artefacto	Descripción
node_index.csv	Índice de 1 131 nodos: imagen, posición (x,y,z) y orientación (yaw)
poses.txt	Poses completas en 6 DOF para los 1 131 nodos
deep_index.csv	Índice de embeddings asociados a identificadores de nodo
deep_embeddings.npy	Matriz de representaciones visuales: $1\,131 \times 2048$ (float32, L2-norm = 1.0)
deep_embeddings.faiss	Índice FAISS tipo IndexFlatIP - 1 131 vectores, 2048D, 9.27 MB

Tabla 4: Artefactos generados para la localización

Tal como se muestra en la Tabla 4 índice FAISS generado, de tipo IndexFlatIP, almacena los 1 131 vectores de 2048 dimensiones en 9.27 MB. Al ser un índice exacto (no aproximado), garantiza recuperación óptima por similitud coseno para colecciones de este tamaño, sin sacrificar precisión por velocidad. Todos los vectores almacenados están L2-normalizados (norma = 1.0), lo que permite que el producto interno coincida directamente con la similitud coseno.

Resultados de localización

Evaluación manual con imágenes reales

Se evaluaron 10 imágenes capturadas manualmente en distintos puntos del entorno, con nodos ground-truth definidos de forma explícita. Esta modalidad refleja condiciones operativas reales con variaciones de iluminación, ángulo de captura y distancia al punto de referencia.

#	Imagen	Nodo GT	Estado	Nodo predicho	Top-1	Top-5	Confianza	Tiempo (s)
1	IMG_8791.jpg	968	Localizado	968	✓	✓	0.862	3.54
2	IMG_8799.jpg	1 051	Localizado	1 051	✓	✓	0.824	2.08
3	IMG_8804.jpg	1 010	No localizado	-	✗	✓	-	2.20
4	IMG_8805.jpg	1 056	No localizado	-	✗	✓	-	2.24
5	IMG_9479.JPG	282	Localizado	282	✓	✓	0.799	3.32
6	IMG_9480_000002.jpg	901	Localizado	384	✗	✗	0.831	0.73
7	IMG_9481_000004.jpg	807	Localizado	568	✗	✗	0.875	0.89
8	IMG_9483.JPG	1 065	No localizado	-	✗	✓	-	3.37
9	IMG_9486.JPG	1 004	Localizado	1 004	✓	✓	0.865	3.33
10	IMG_9486_000012.jpg	895	Localizado	474	✗	✗	0.842	0.78

Tabla 5: Resultados detallados de la evaluación manual con 10 imágenes reales.

Métrica	Valor
Consultas totales	10
Exactitud Top-1	40 % (4 / 10)
Exactitud Top-5	70 % (7 / 10)
No localizados (RANSAC rechazado)	30 % (3 / 10)
Predicciones incorrectas	30 % (3 / 10)
Tiempo promedio de consulta	2.25 s
Tiempo mínimo / máximo	0.73 s / 3.54 s
Confianza promedio (consultas localizadas)	0.590

Tabla 6: Resumen estadístico de la evaluación manual (10 consultas).

Como se observa en las Tablas 5 y 6, los resultados de la evaluación manual evidenciaron el comportamiento diferencial del sistema según las condiciones de captura. Las consultas con alta similitud visual respecto al nodo de referencia (entradas 1 y 9) lograron localización correcta con confianza superior a 0.86. Los tres casos no localizados (entradas 3, 4 y 8) fueron rechazados por la verificación RANSAC al no alcanzar el mínimo de 15 inliers requeridos; sin embargo, en los tres casos el nodo correcto apareció entre los cinco candidatos recuperados en la etapa de recuperación profunda (Top-5 correcto), lo que indica que la similitud visual global fue suficiente pero la correspondencia geométrica local resultó insuficiente para confirmar la predicción. Los tres casos con predicción incorrecta (entradas 6, 7 y 10) correspondieron a imágenes capturadas con ángulos o encuadres que generaron mayor similitud visual con nodos adyacentes al esperado.

Evaluación sintética: comparación de descriptores ORB y SuperPoint+LightGlue (20 consultas)

Se ejecutó una evaluación controlada con 20 imágenes extraídas directamente del índice de nodos del mapa (semilla aleatoria fija, seed = 42), procesando cada consulta de forma independiente con ORB y con SuperPoint+LightGlue sobre el mismo conjunto, en CUDA.

Métrica	ORB	SuperPoint + LightGlue
Consultas totales	20	20
Exactitud Top-1	100 % (20/20)	100 % (20/20)
Exactitud Top-5	100 %	100 %
Error medio de orientación	0.0°	0.0°
No localizados	0 %	0 %
Tiempo promedio por consulta	0.283 s	0.851 s
Tiempo mínimo / máximo	0.141 s / 0.581 s	0.531 s / 2.355 s
Factor de tiempo SPL / ORB	-	3.0× más lento
Predicciones idénticas entre descriptores	20 / 20 (100 %)	-

Tabla 7: Comparación de descriptores en evaluación sintética (20 consultas, CUDA).

Como se indica en la Tabla 7, ambos descriptores alcanzaron 100 % de exactitud Top-1 y Top-5 sobre el conjunto sintético, con error de orientación nulo en todos los casos. La principal diferencia fue el tiempo de respuesta: SuperPoint+LightGlue requirió en promedio 3.0 veces más tiempo que ORB (0.851 s frente a 0.283 s). Ambos descriptores produjeron predicciones idénticas en las 20 consultas, lo que evidencia que la etapa de recuperación profunda “compartida” es determinante para la precisión, mientras que el descriptor influye principalmente en el tiempo de respuesta y en la robustez ante consultas con mayor variación visual respecto al nodo de referencia.

La Figura 8 muestra el flujo real ejecutado en la evaluación sintética. Cada uno de los 20 nodos seleccionados con semilla fija (seed = 42) se procesa como imagen de consulta: ResNet50 extrae un embedding de 2048 dimensiones que se compara contra el índice FAISS para recuperar los 10 candidatos más similares por coseno. Sobre cada candidato se ejecutan de forma independiente los descriptores ORB y SuperPoint+LightGlue (SPL), cuyas correspondencias se someten a verificación geométrica por homografía con RANSAC. El candidato que supera el umbral de inliers (≥ 15) con mayor consistencia geométrica se acepta como nodo localizado; los restantes se descartan. En las 20 consultas, ORB y SPL produjeron el mismo nodo predicho, con confianza compuesta de 1.0 y error de orientación de 0.0°, confirmando que en condiciones intra-sesión la etapa de recuperación profunda es determinante y el descriptor local influye únicamente en el tiempo de respuesta.

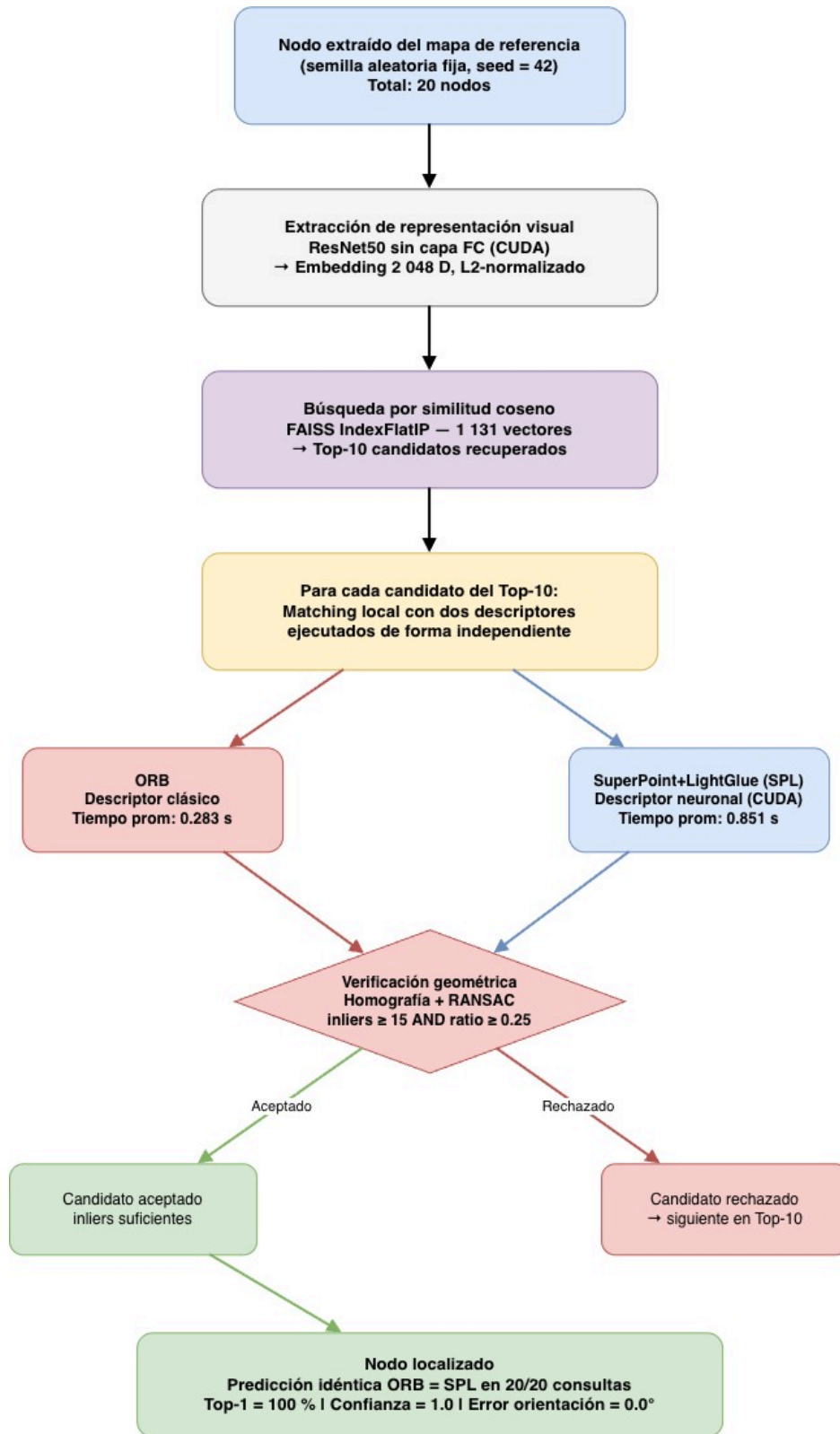


Figura 8: Flujo de evaluación sintética comparativa ORB vs. SPL.

La Figura 9. Panel superior: tiempo de respuesta por consulta; SPL (SuperPoint+LightGlue) requiere en promedio $3.0\times$ más tiempo que ORB (0.851 s vs. 0.283 s). Panel inferior: confianza compuesta; ambos descriptores alcanzan el valor máximo (1.0) en la totalidad de las consultas. Exactitud Top-1 = Top-5 = 100 % y error de orientación = 0.0° en todos los casos.

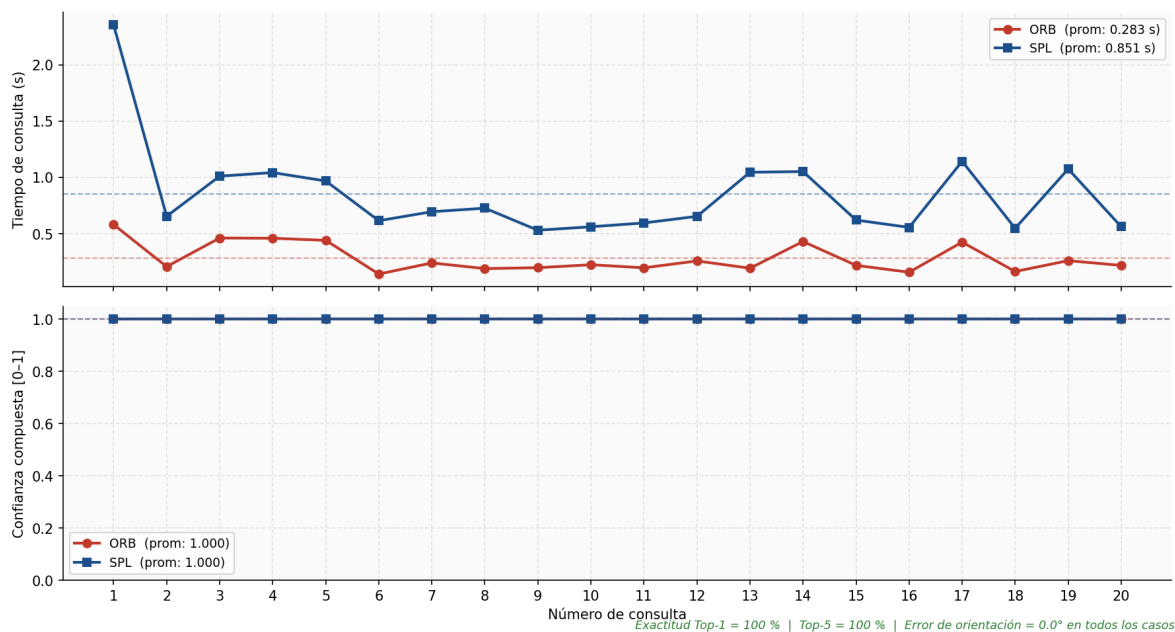


Figura 9: ORB vs. SuperPoint+LightGlue evaluación sintética.

Método híbrido de localización

En la tabla 8 se resume como el método híbrido combinó la recuperación por representación profunda con la verificación geométrica local, reduciendo el espacio de búsqueda de 1 131 a 10 candidatos mediante similitud vectorial y validando cada candidato con correspondencias locales y RANSAC.

Etapas	Método	Resultado
Extracción de representación	Backbone ResNet50 (sin capa FC)	Embedding 2048 D, L2-normalizado
Búsqueda de candidatos	FAISS IndexFlatIP - similitud coseno	Top-10 nodos candidatos
Matching local	SuperPoint + LightGlue / ORB (configurable)	Correspondencias de puntos de interés
Verificación geométrica	Homografía + RANSAC (umbral 5.0 px)	Validación espacial del candidato
Criterio de aceptación	$\text{inliers} \geq 15$ AND $\text{ratio} \geq 0.25$	Nodo verificado o rechazo
Cálculo de confianza	$0.4 \times \text{cos} + 0.3 \times \text{ratio} + 0.3 \times \text{inliers_norm}$	Valor compuesto [0, 1]
Clasificación semántica	CLIP zero-shot (8 etiquetas indoor)	Etiqueta del tipo de lugar
Salida final	Nodo RTAB-Map identificado	Pose + confianza + etiqueta semántica

Tabla 8: Componentes del método híbrido de localización.

La **Figura 10 muestra** el proceso de relocalización dividido en dos etapas (online y offline). En la etapa *offline*, las imágenes asociadas a los nodos del mapa generado con RTAB-Map se procesan mediante una red **ResNet50** para obtener descriptores globales o *embeddings*. Estos descriptores se almacenan en un índice visual implementado con **FAISS**, junto con la relación entre cada representación y su correspondiente nodo del mapa.

La etapa online, una imagen de consulta se procesa con la misma red para generar su embedding, el cual se compara con el índice mediante el producto punto. Como resultado, se seleccionan los Top-K nodos visualmente más similares. Posteriormente, estos candidatos se someten a una verificación geométrica utilizando SuperPoint y LightGlue, o alternativamente ORB, seguida de la estimación de una homografía con RANSAC. Esta etapa permite descartar coincidencias visuales incorrectas y seleccionar el nodo con mayor consistencia geométrica. Finalmente, el identificador del nodo seleccionado se relaciona con el archivo de poses de RTAB-Map para recuperar su posición y orientación, generando como salida el nodo relocalizado y su pose dentro del mapa.

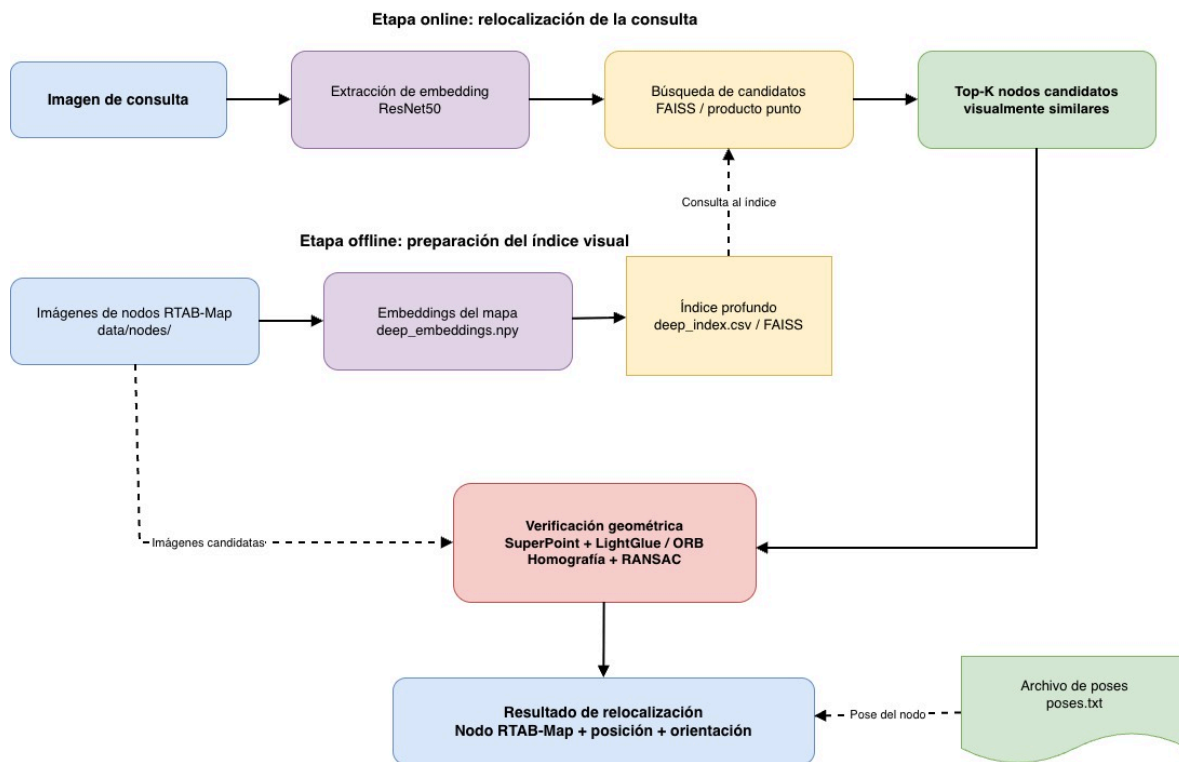


Figura 10: Arquitectura del método híbrido de relocalización.

Evaluación cross-session con 490 consultas

La prueba de validación central del sistema consistió en una evaluación cross-session con 490 consultas extraídas íntegramente de la base de datos RTAB-Map. Estas imágenes provienen de un recorrido independiente al del mapa de referencia de mayo, con origen de coordenadas propio y separación temporal de 18 días.

Dado que los sistemas de coordenadas de ambas sesiones son independientes, la asignación del ground truth no puede realizarse por comparación directa de identificadores de nodo. Se empleó un método de alineación de marcos de coordenadas por análisis de Procrustes 2D: a partir de 464 pares localizados en una evaluación previa se estimó una transformación rígida 2D “rotación $R \in SO(2)$ y traslación $t \in \mathbb{R}^2$ ” mediante descomposición SVD que minimiza la suma de distancias euclidianas entre las poses transformadas de la sesión de consulta y las poses de la sesión de referencia. La transformación estimada presenta una rotación de -58.65° y una traslación de (6.12, -5.42) m. Para cada nodo de consulta, su pose XY se transforma al marco de coordenadas de la sesión de referencia y se asigna como ground truth el nodo más cercano en distancia euclidiana, garantizando que el criterio de evaluación sea independiente del modelo de recuperación.

Métrica	SuperPoint+LightGlue	ORB
Consultas evaluadas	490	490
Localización exitosa	464 (94.7 %)	449 (91.6 %)
No localizadas (RANSAC rechazado)	26 (5.3 %)	41 (8.4 %)
Error espacial mediano	0.615 m	0.764 m
Error espacial medio	2.151 m	2.398 m
Consultas con error ≤ 0.5 m	201 (43.3 %)	—
Consultas con error ≤ 1.0 m	314 (67.7 %)	257 (57.2 %)
Consultas con error ≤ 2.0 m	364 (78.4 %)	323 (71.9 %)
Consultas con error ≤ 5.0 m	408 (87.9 %)	—
Confianza media (localizadas)	0.830	—
Tiempo mediano por consulta	840 ms	285 ms

Embedding ResNet50	21 ms	21 ms
Búsqueda FAISS	0.53 ms	0.53 ms
Matching local	718 ms	136 ms
Verificación RANSAC	90 ms	—
Factor de velocidad SPL / ORB	2.9× más lento	—
Predicciones idénticas SPL = ORB	135/490 (27.6 %)	—

Tabla 9: Evaluación cross-session con 490 consultas — SuperPoint+LightGlue vs. ORB.

La Figura 11 presenta los histogramas de error de orientación para ambos descriptores sobre las 490 consultas cross-session.

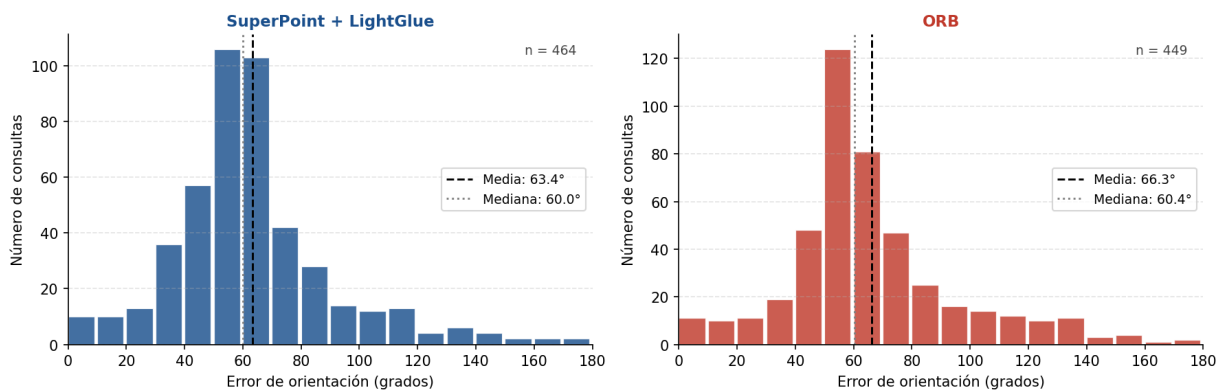


Figura 11: Histograma de error de orientación - ORB vs. SuperPoint+LightGlue 490 consultas

Como muestra la Figura 11, la distribución del error de orientación se concentra en la banda 50°–70° para ambos descriptores, con media de 63.4° (SPL) y 66.3° (ORB) y mediana de 60.0° y 60.4° respectivamente. Este error sistemático es consistente con la rotación de -58.65° estimada por el método de Procrustes entre los marcos de coordenadas de ambas sesiones, y no representa un fallo de estimación sino la diferencia angular inherente entre el origen del recorrido de abril y el del mapa de mayo. En un despliegue operativo donde todas las predicciones se expresan en el marco del mapa de referencia, esta diferencia no tiene efecto sobre la utilidad posicional del resultado.

Visualización de resultados

El sistema generó automáticamente visualizaciones para facilitar la interpretación del resultado y la inspección del comportamiento interno del pipeline.

Visualización	Descripción	Referencia
Comparación consulta-nodo	Imagen de consulta junto al nodo predicho con correspondencias superpuestas	Figura 2
Mapa con todos los nodos	Plano 2D con los 1 131 nodos proyectados; muestra la cobertura del recorrido	Figura 4
Mapa con nodo localizado	Plano 2D con el nodo predicho resaltado con flecha de orientación (yaw)	Figura 5
Histograma de error de orientación	Distribución del error angular en grados sobre el conjunto evaluado	Figura 6

Tabla 10: Visualizaciones generadas por el sistema.

La Figura 12 presenta la distribución espacial de los 1 131 nodos proyectados sobre el plano del entorno, lo que permitió verificar la cobertura del recorrido y la calibración de los parámetros de proyección afin.

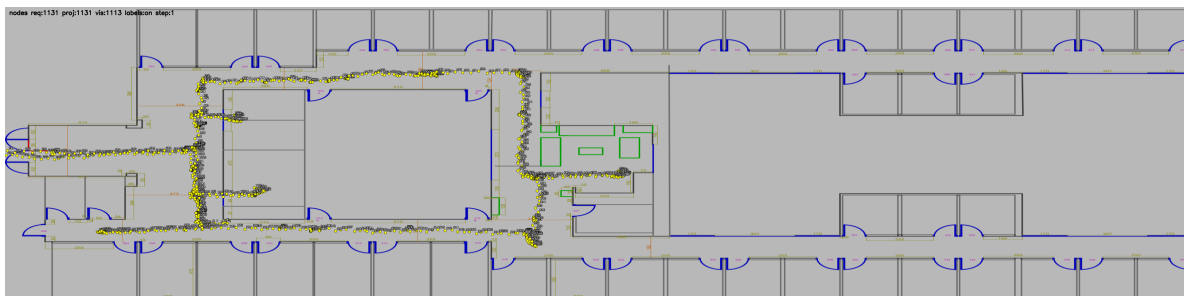


Figura 12: Proyección de los 1 131 nodos del mapa sobre el plano 2D del entorno.

La Figura 13 presenta el nodo localizado sobre el mismo plano con la flecha de orientación correspondiente al ángulo yaw estimado de 187° .

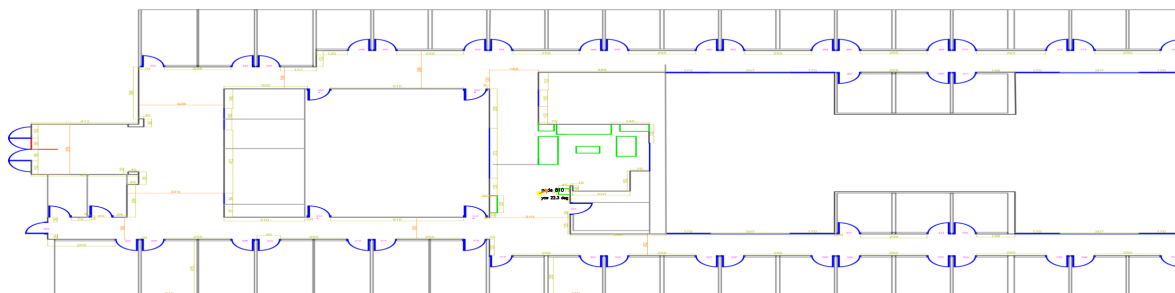


Figura 13: Nodo localizado y proyectado sobre el plano 2D con orientación estimada.

5.2. Discusión

En este trabajo se demostró que fue posible desarrollar un sistema de relocalización visual indoor desacoplado del núcleo de RTAB-Map. A partir de la base de datos generada por RTAB-Map, se reutilizaron las imágenes de los nodos y sus poses para construir un módulo externo capaz de estimar la ubicación de una imagen de consulta dentro del mapa, sin intervención adicional del sistema SLAM durante la fase de consulta.

Uno de los resultados más relevantes fue la generación de un dataset completo, compuesto por 1 131 imágenes de nodos y 1 131 poses asociadas. Cada imagen quedó vinculada con una posición y orientación dentro del entorno, lo que confirmó que la información almacenada por RTAB-Map podía transformarse en una base útil para tareas de relocalización visual. El dataset cubre un área aproximada de 323 m^2 con variación vertical inferior a 0.22 m , lo que caracteriza al entorno como esencialmente plano y adecuado para la proyección 2D empleada en la evaluación de error espacial.

También se implementó un método híbrido que combinó embeddings profundos generados por ResNet50, búsqueda eficiente mediante FAISS y verificación geométrica con RANSAC. Esta estrategia redujo el espacio de búsqueda al recuperar primero los diez nodos visualmente más similares, y conservó una validación geométrica que evitó aceptar coincidencias visuales incorrectas. Frente a métodos clásicos basados solo en características locales, este enfoque ofrece una estructura más robusta para entornos interiores con pasillos, puertas o zonas visualmente repetitivas.

Los resultados cuantitativos mostraron un comportamiento diferenciado según el tipo de consulta. En la evaluación comparativa de matchers sobre 20 consultas sintéticas derivadas del propio mapa, tanto ORB como SuperPoint+LightGlue alcanzaron $\text{Top-1} = 100 \%$ y $\text{Top-5} = 100 \%$, con errores de orientación nulos en todos los casos. Esto validó el funcionamiento del pipeline bajo condiciones controladas. En la evaluación manual con 10 consultas reales capturadas en el entorno, el sistema alcanzó $\text{Top-1} = 40 \%$ y $\text{Top-5} = 70 \%$, con un 30% de consultas no localizadas por no superar el umbral de verificación

geométrica. Esta diferencia revela la brecha que existe entre imágenes sintéticas y consultas reales con variación de perspectiva, iluminación y escala.

La prueba de validación central fue la evaluación cross-session con 490 consultas extraídas de la base de datos RTAB-Map con 490 frames registrados, probadas contra la base de datos que contiene el mapa base. Este escenario “dos sesiones independientes, marcos de coordenadas distintos, separación temporal de 18 días” constituye la condición operativa más exigente evaluada en este trabajo. Con SuperPoint+LightGlue el sistema alcanzó una tasa de localización exitosa del 94.7 % (464 de 490 consultas), con un error espacial mediano de 0.615 m. El 67.7 % de las consultas localizadas presentó un error inferior a 1 m y el 78.4 % inferior a 2 m, lo que sitúa al sistema dentro de los rangos de precisión requeridos para aplicaciones de orientación y guía en interiores. Con ORB la tasa de éxito fue del 91.6 %, con un error mediano de 0.764 m, confirmando que ambos descriptores son viables en condiciones cross-session aunque con diferencias cuantitativas significativas.

Un resultado particularmente ilustrativo de la evaluación cross-sesión fue que las predicciones de ORB y SuperPoint+LightGlue coincidieron únicamente en el 27.6 % de los casos (135 de 490 consultas), frente al 100 % de coincidencia observado en la evaluación sintética intra-sesión. Esta divergencia pone de manifiesto que la capacidad discriminativa del descriptor importa de forma significativa cuando existe variación real de apariencia entre la imagen de consulta y el mapa de referencia, y que los resultados de evaluaciones sintéticas no son directamente transferibles a condiciones cross-session.

Respecto al error de orientación observado en la evaluación cross-session —media de 63.4° con SuperPoint+LightGlue y 66.3° con ORB—, este valor es consistente con la rotación de -58.65° estimada por el método de Procrustes entre los marcos de coordenadas de ambas sesiones. El error no refleja una limitación del estimador local sino la diferencia angular inherente entre el origen del recorrido de abril y el del mapa de mayo. En un despliegue operativo donde todas las predicciones se expresan en el marco del mapa de referencia, esta diferencia queda implícita en las coordenadas del nodo predicho y no afecta la utilidad posicional del resultado.

Respecto al tiempo de procesamiento, los resultados de la evaluación cross-session mostraron tiempos medianos de 285 ms por consulta para ORB y de 840 ms para SuperPoint+LightGlue, una diferencia de 2.9×. El desglose por etapa revela que el matching local representa el 85 % del tiempo total con SuperPoint+LightGlue (718 ms de 840 ms), mientras que la extracción del embedding ResNet50 requiere 21 ms y la búsqueda FAISS exacta mediante IndexFlatIP sobre los 1 131 vectores se completa en 0.53 ms. ORB resulta viable para escenarios donde la latencia es prioritaria; SuperPoint+LightGlue ofrece mayor precisión posicional y menor tasa de rechazo, pero con mayor costo computacional.

La visualización de resultados también fue relevante. Al generar imágenes de correspondencias y proyecciones sobre el plano 2D, fue posible interpretar no solo qué nodo fue seleccionado, sino también dónde se ubica dentro del entorno y con qué orientación. Esto facilitó la revisión de resultados y permitió detectar inconsistencias entre la localización estimada y la representación del plano.

No obstante, la evaluación presentó limitaciones observables. El 12.1 % de consultas con error espacial mayor o igual a 5 m corresponde principalmente a dos situaciones: zonas del recorrido de abril sin cobertura equivalente en el mapa de mayo, donde el nodo ground truth más cercano puede estar a varios

metros de la posición real del robot; y pasillos con apariencia visual repetitiva, donde el sistema acepta una homografía localmente coherente pero globalmente incorrecta. El umbral de aceptación RANSAC ($\text{inliers} \geq 15$) es fijo y no se adapta a la densidad de textura local, lo que puede provocar rechazos falsos en zonas con escasa textura o aceptaciones incorrectas en estructuras geométricas repetitivas. Finalmente, el módulo de clasificación semántica CLIP opera como metadato adicional y no participa en la selección del candidato, representando una capacidad disponible pero aún no integrada como criterio de re-ranking.

A partir de este trabajo se abren varias líneas futuras. Es pertinente comparar extractores de embeddings globales alternativos como DINOv2 o NetVLAD, analizar el impacto de variar el número de candidatos recuperados por FAISS más allá de los diez empleados en esta evaluación, y explorar umbrales de verificación geométrica adaptativos que se ajusten a la densidad de textura local. La integración del módulo CLIP como criterio de re-ranking entre los candidatos Top-K representa una dirección concreta con potencial para reducir los outliers espaciales en zonas de alta ambigüedad visual. Finalmente, la incorporación de retroalimentación humana para corregir casos difíciles y reutilizarlos en futuras iteraciones del sistema representa una dirección de trabajo valiosa para escenarios de despliegue continuo.

6. CONCLUSIONES

Los resultados obtenidos permiten concluir que se cumplió el objetivo general planteado en este Trabajo de Obtención de Grado, al desarrollar e implementar una metodología computacional automatizada, reproducible y evaluable para la relocalización visual en entornos interiores a partir de mapas generados con RTAB-Map. La metodología integró técnicas de aprendizaje profundo y visión por computadora clásica para recuperar candidatos visuales, verificar geoméricamente las correspondencias y estimar la posición y orientación asociadas al nodo localizado. Asimismo, se implementaron mecanismos para la construcción y consulta de índices visuales, el cálculo de métricas cuantitativas de desempeño, la exposición de resultados mediante una API REST y la proyección de la localización sobre una representación 2D del entorno.

La evaluación cross-session con 490 imágenes reales provenientes de una sesión de captura independiente constituyó la prueba experimental de mayor exigencia. SuperPoint+LightGlue alcanzó una tasa de localización exitosa del 94.7 %, con un error espacial mediano de 0.615 m, mientras que ORB alcanzó una tasa de éxito del 91.6 % y un error mediano de 0.764 m. Estos resultados aportan evidencia cuantitativa sobre la viabilidad del enfoque propuesto y permiten identificar tanto sus fortalezas como las condiciones en las que la variación visual y la similitud estructural del entorno afectan su desempeño.

6.1. Conclusiones

El desarrollo realizado permitió comprobar que es posible construir un módulo externo de relocalización visual indoor sin modificar el núcleo de RTAB-Map. A partir de la base de datos generada por el sistema SLAM, se reutilizaron las imágenes de los nodos y sus poses asociadas para estimar la ubicación de una imagen de consulta dentro del entorno mapeado, manteniendo una separación completa entre la fase de construcción del mapa y la fase de localización.

Logré generar un dataset estructurado compuesto por 1 131 imágenes de nodos con sus poses correspondientes, cubriendo un área aproximada de 323 m². Cada imagen quedó vinculada con una posición (x, y) y una orientación en yaw, lo que hizo posible la relocalización visual a partir de información previamente registrada. La validación del dataset confirmó que todas las imágenes contaban con pose asociada, sin omisiones.

Se implementó un pipeline de indexación basado en embeddings de 2048 dimensiones extraídos mediante una red ResNet50 y almacenados en un índice FAISS IndexFlatIP de búsqueda exacta por similitud coseno. Este índice, con un tamaño de 8.84 MB para 1 131 vectores, demostró ser eficiente en memoria y con tiempos de búsqueda despreciables frente al costo de verificación geométrica.

El método híbrido que combinó recuperación profunda con verificación geométrica mediante RANSAC fue una decisión de diseño relevante. En entornos interiores con zonas visualmente similares, la selección por similitud visual no fue suficiente; la validación geométrica sobre correspondencias locales permitió descartar coincidencias incorrectas y reportar únicamente localizaciones con soporte geométrico verificado.

La evaluación comparativa de los matchers sobre 20 consultas sintéticas mostró que ORB y SuperPoint+LightGlue alcanzaron exactitudes idénticas de Top-1=100 %, con tiempos promedio de 0.283 s y 0.851 s respectivamente. En la evaluación con consultas reales capturadas manualmente, el sistema alcanzó Top-1=40 % y Top-5=70 %, con un 30 % de consultas no localizadas. Esta diferencia evidenció la brecha entre condiciones controladas y el uso real del sistema.

Las herramientas de visualización desarrolladas “imágenes de correspondencias entre consulta y nodo localizado, y proyección sobre el plano 2D con orientación estimada” facilitaron la interpretación de resultados y la detección de fallos del pipeline de forma directa y sin instrumentación adicional.

Como conclusión general, el sistema desarrollado constituye una base funcional para relocalización visual indoor sobre mapas generados con RTAB-Map. El enfoque modular permitió integrar técnicas de recuperación por aprendizaje profundo y verificación geométrica clásica sin alterar el sistema SLAM original, lo que facilita futuras mejoras, sustitución de componentes y comparaciones con otros métodos.

6.2. Trabajo Futuro

Como trabajo futuro, se propone ampliar la diversidad de la evaluación experimental mediante nuevas sesiones de captura realizadas en diferentes días, condiciones de iluminación, dispositivos y configuraciones del entorno. Aunque la evaluación cross-session desarrollada en este trabajo contempla 490 consultas reales provenientes de una sesión independiente, incorporar múltiples sesiones adicionales permitiría analizar con mayor profundidad la capacidad de generalización del sistema frente a cambios temporales y visuales más diversos.

También considero necesario comparar diferentes modelos de extracción de embeddings globales, como NetVLAD, DINOv2 o modelos visión-lenguaje como CLIP, frente al extractor actual basado en ResNet. Esta comparación permitiría identificar qué representación visual ofrece mejor equilibrio entre precisión de recuperación y tiempo de inferencia en entornos interiores con zonas visualmente similares.

Otra línea de trabajo consiste en optimizar el tiempo de procesamiento por consulta. El sistema actual presenta una latencia inicial de aproximadamente 6.5 s debida a la carga del modelo, y tiempos de consulta subsecuentes de 0.28 s con ORB y 0.85 s con SuperPoint+LightGlue. Para acercarse a escenarios de uso en tiempo real, se pueden explorar estrategias como la precarga persistente del modelo, el uso de índices FAISS aproximados "IVF o HNSW" y la cuantización de pesos de la red de extracción, manteniendo el hardware GPU disponible.

Además, sería útil incorporar retroalimentación supervisada en los casos donde el sistema falle o presente baja confianza. Esta información podría utilizarse para construir un conjunto de casos difíciles, ajustar los umbrales de verificación geométrica y, en iteraciones posteriores, refinar el extractor de embeddings con ejemplos del entorno específico de despliegue.

Finalmente, propongo explorar la integración del sistema con aplicaciones de navegación indoor asistida, en particular orientada a personas con discapacidad visual. El nodo localizado puede servir como punto de referencia dentro del mapa para proporcionar indicaciones de ubicación en lenguaje natural, estimar la distancia a puntos de interés o apoyar la corrección de trayectoria en entornos previamente mapeados. Esta integración representaría el cierre del ciclo entre la construcción del mapa con RTAB-Map y su uso funcional en un escenario de asistencia real.

7. BIBLIOGRAFÍA

- [1] T. M. A. R. Daniel DeTone, «SuperPoint: Self-Supervised Interest Point Detection and Description,» arxiv, 19 04 2018. [En línea]. Available: <https://arxiv.org/abs/1712.07629>. [Último acceso: 13 05 2026].
- [2] P.-E. S. M. P. Philipp Lindenberger, «LightGlue: Local Feature Matching at Light Speed,» arxiv, 23 06 2023. [En línea]. Available: <https://arxiv.org/abs/2306.13643>. [Último acceso: 13 05 2026].
- [3] A. G. C. D. J. J. G. S. P.-E. M. M. L. L. H. H. J. Matthijs Douze, «The Faiss library,» arxiv, 23 10 2025. [En línea]. Available: <https://arxiv.org/abs/2401.08281>. [Último acceso: 13 05 2026].
- [4] S. O. R. t. R. T. A. D. Barath, «SuperRANSAC: One RANSAC to Rule Them All,» arxiv, 5 06 2025. [En línea]. Available: <https://arxiv.org/abs/2506.04803>. [Último acceso: 13 05 2026].
- [5] F. M. Mathieu Labbé, «Memory Management for Real-Time Appearance-Based Loop Closure Detection,» arxiv, 22 07 2024. [En línea]. Available: <https://arxiv.org/abs/2407.15890>. [Último acceso: 13 05 2026].
- [6] D. S. A. D. Marziyeh Bamdad, «SLAM for Visually Impaired People: a Survey,» arxiv, 16 08 2024. [En línea]. Available: <https://arxiv.org/abs/2212.04745>. [Último acceso: 13 05 2026].
- [7] E. D. A. J. D. Riku Murai, «MASt3R-SLAM: Real-Time Dense SLAM with 3D Reconstruction Priors,» arxiv, 02 06 2025. [En línea]. Available: <https://arxiv.org/abs/2412.12392>. [Último acceso: 13 05 2026].
- [8] R. E. J. J. G. R. J. M. M. M. J. D. T. Carlos Campos, «ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial and Multi-Map SLAM,» arxiv, 23 04 2021. [En línea]. Available: <https://arxiv.org/abs/2007.11898>. [Último acceso: 13 05 2026].

- [9] X. Z. Z. J. C. Y. Chaoyang Jiang, «UMS-VINS: United Monocular-Stereo Features for Visual-Inertial Tightly Coupled Odometry,» arxiv, 15 03 2023. [En línea]. Available: <https://arxiv.org/abs/2303.08550>. [Último acceso: 13 05 2026].
- [10] F. M. Mathieu Labbé, «RTAB-Map as an Open-Source Lidar and Visual SLAM Library for Large-Scale and Long-Term Online Operation,» arxiv, 10 03 2024. [En línea]. Available: <https://arxiv.org/abs/2403.06341>. [Último acceso: 13 05 2026].
- [11] P. G. A. T. T. P. J. S. Relja Arandjelović, «NetVLAD: CNN architecture for weakly supervised place recognition,» arxiv, 02 05 2016. [En línea]. Available: <https://arxiv.org/abs/1511.07247>. [Último acceso: 13 05 2026].
- [12] C. C. R. S. M. D. Paul-Edouard Sarlin, «From Coarse to Fine: Robust Hierarchical Localization at Large Scale,» arxiv, 08 04 2019. [En línea]. Available: <https://arxiv.org/abs/1812.03506>. [Último acceso: 13 05 2026].
- [13] M. O. T. S. M. C. M. P. J. S. T. P. A. T. Hajime Taira, «InLoc: Indoor Visual Localization with Dense Matching and View Synthesis,» arxiv, 08 04 2018. [En línea]. Available: <https://arxiv.org/abs/1803.10368>. [Último acceso: 13 05 2026].
- [14] D. D. T. M. A. R. Paul-Edouard Sarlin, «SuperGlue: Learning Feature Matching with Graph Neural Networks,» arxiv, 28 03 2020. [En línea]. Available: <https://arxiv.org/abs/1911.11763>. [Último acceso: 13 05 2026].
- [15] A. G. A. G. Faheem Zafari, «A Survey of Indoor Localization Systems and Technologies,» IEEE, 16 06 2019. [En línea]. Available: <https://arxiv.org/pdf/1709.01015>. [Último acceso: 13 05 2026].
- [16] F. G. A. & L. K. K. Zafari, «A Survey of Indoor Localization Systems and Technologies,» IEEE Communications Surveys & Tutorials, 2019. [En línea]. Available: <https://arxiv.org/abs/1709.01015>. [Último acceso: 2026 05 13].
- [17] M. G. a. R. C. A. Kendall, «A Convolutional Network for Real-Time 6-DOF Camera Relocalization,» IEEE, 2015. [En línea]. Available: <https://arxiv.org/abs/1505.07427> . [Último acceso: 2026 05 13].
- [18] C. C. e. al, «Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age,» IEEE Trans. Robot, 2016. [En línea]. Available: <https://arxiv.org/abs/1606.05830>. [Último acceso: 13 05 2026].
- [19] M. L. a. F. Michaud, «RTAB-Map as an Open-Source Lidar and Visual Simultaneous Localization and Mapping Library for Large-Scale and Long-Term Online Operation,» J. Field Robot, 2016. [En línea]. Available: <https://doi.org/10.1002/rob.21831>. [Último acceso: 13 05 2026].

- [20] L. a. F. Michaud, «RTAB-Map as an Open-Source Lidar and Visual Simultaneous Localization and Mapping Library for Large-Scale and Long-Term Online Operation,» . Field Robot, 2019. [En línea]. Available: <https://doi.org/10.1002/rob.21831> . [Último acceso: 13 05 2026].
- [21] W. B. a. D. F. S. Thrun, «Probabilistic Robotics,» Cambridge, MA, USA: MIT Press, 2005. [En línea]. Available: <https://docs.ufpr.br/~danielsantos/ProbabilisticRobotics.pdf> . [Último acceso: 13 05 2026].
- [22] R. K. C. S. a. W. B. G. Grisetti, «A Tutorial on Graph-Based SLAM,» IEEE Intell. Transp. Syst. Mag, 2010. [En línea]. Available: <https://arxiv.org/abs/1010.0686>. [Último acceso: 13 05 2026].
- [23] M. L. a. F. Michaud, «RTAB-Map as an Open-Source Lidar and Visual Simultaneous Localization and Mapping Library for Large-Scale and Long-Term Online Operation,» J. Field Robot., 2016. [En línea]. Available: <https://doi.org/10.1002/rob.21831> . [Último acceso: 13 05 2026].
- [24] C. C. e. al, «Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age,» 2016, 12. [En línea]. Available: <https://arxiv.org/abs/1606.05830>. [Último acceso: 13 05 2026].
- [25] B. Kuipers, «The Spatial Semantic Hierarchy,» B. Kuipers, 05 2000. [En línea]. Available: <https://web.eecs.umich.edu/~kuipers/papers/Kuipers-aij-00.pdf> . [Último acceso: 13 05 2026].
- [26] M. L. a. F. Michaud, «RTAB-Map as an Open-Source Lidar and Visual Simultaneous Localization and Mapping Library for Large-Scale and Long-Term Online Operation,» Field Robot, 03 2019. [En línea]. Available: <https://doi.org/10.1002/rob.21831>. [Último acceso: 13 05 2026].
- [27] X. L. K. M. Goh, «ResNet: Habilitando redes neuronales convolucionales profundas mediante aprendizaje residua,» arxiv, 28 10 2025. [En línea]. Available: <https://arxiv.org/html/2510.24036v1>. [Último acceso: 13 05 2026].
- [28] Y. L. F. S. W. L. a. H. Yanhao LiaoORCID, «Deep Unsupervised Homography Estimation for Single-Resolution Infrared and Visible Images Using GNN,» School of Computer Science, Civil Aviation Flight University of China, 25 09 2024. [En línea]. Available: <https://www.mdpi.com/2079-9292/13/21/4173>. [Último acceso: 13 05 2026].
- [29] y. l. X. W. C. W. Xingyu Yuan, «Algoritmo de fusión semántica de LiDAR 2D y cámara basado en contorno y proyección inversa,» Facultad de Ciencias e Ingeniería Mecánica, Universidad de Ciencia y Tecnología de Huazhong, 17 04 2025. [En línea]. Available: <https://www.mdpi.com/1424-8220/25/8/2526>. [Último acceso: 13 05 2026].

- [30] deepanraj, «Understanding the RANSAC Model: Robust Fitting for Real-World Data,» 6 11 2024. [En línea]. Available: <https://medium.com/@717821i208/ransac-3187c732ac5d>. [Último acceso: 13 05 2026].
- [31] A. R. e. al, «Learning Transferable Visual Models From Natural Language Supervision,» Int. Conf. Machine Learning , 11 06 2021. [En línea]. Available: <https://arxiv.org/abs/2103.00020>. [Último acceso: 13 05 2026].
- [32] W. H. Organization, «World Report on Vision,» 2019. [En línea]. Available: <https://www.who.int/publications/i/item/world-report-on-vision> . [Último acceso: 13 05 2026].
- [33] C. G. C. R. K. K. H. T. y. C. A. R. Ahmetovic, «avCog: A Navigational Cognitive Assistant for the Blind,» Conf. Human-Computer Interaction with Mobile Devices and Services (MobileHCI), 2016. [En línea]. Available: <https://doi.org/10.1145/2935334.2935361> . [Último acceso: 13 05 2026].
- [34] Y. L. X. H. y. X. Z. F. Zhu, «Seamless Indoor Pedestrian Tracking and Navigation with Multimodal Sensor Fusion,» IEEE Internet of Things Journal, 2021. [En línea]. Available: <https://doi.org/10.1109/JIOT.2020.3015791> . [Último acceso: 13 05 2026].
- [35] I. A. K. B. y. E. F. N. Fallah, «ndoor Human Navigation Systems: A Survey,» Interacting with Computers, 2013. [En línea]. Available: <https://doi.org/10.1093/iwc/iws010> . [Último acceso: 13 05 2026].
- [36] I. A. y. E. F. A. Fiannaca, «Headlock: A Wearable Navigation Aid That Helps Blind Cane Users Traverse Large Open Spaces,» Conf. Computers & Accessibility (ASSETS), 2014. [En línea]. Available: <https://doi.org/10.1145/2661334.2661453>. [Último acceso: 13 05 2026].
- [37] A. H. y. M. N. S. Poslad, «Improving the Ease of Use and Accessibility of Indoor Location and Navigation Apps for People with Visual Impairments,» IEEE Access, 2020. [En línea]. Available: <https://doi.org/10.1109/ACCESS.2020.3006726> . [Último acceso: 13 05 2026].
- [38] E. B. M. C. y. E. G. G. Anastasi, «Wi-Fi in Ad Hoc Mode: A Measurement Study,» 2nd IEEE Annual Conf. Pervasive Computing and Communications Workshops, 2004. [En línea]. Available: <https://doi.org/10.1109/PERCOMW.2004.1276921> . [Último acceso: 13 05 2026].
- [39] P. B. y. V. N. Padmanabhan, «RADAR: An In-Building RF-Based User Location and Tracking System,» Proc. IEEE INFOCOM, 2000. [En línea]. Available: <https://doi.org/10.1109/INFCOM.2000.832252> . [Último acceso: 13 05 2026].
- [40] X. Z. S. R. y. J. S. K. He, «Deep Residual Learning for Image Recognition,» IEEE Conf. Computer Vision and Pattern Recognition , 2016. [En línea]. Available: <https://doi.org/10.1109/CVPR.2016.90> . [Último acceso: 13 05 2026].

- [41] M. A. F. y. R. C. Bolles, «Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography,» Commun. ACM, 1981. [En línea]. Available: <https://doi.org/10.1145/358669.358692> . [Último acceso: 13 05 2026].
- [42] A. R. e. al., «Learning Transferable Visual Models From Natural Language Supervision,» 38th Int. Conf. Machine Learning (ICML), 2021. [En línea]. Available: <https://arxiv.org/abs/2103.00020>. [Último acceso: 13 05 2026].