UNIVERSIDAD MIGUEL HERNÁNDEZ DE ELCHE

ESCUELA POLITÉCNICA SUPERIOR DE ELCHE

GRADO EN INGENIERÍA ELECTRÓNICA YAUTOMÁTICA INDUSTRIAL

ESTUDIO DE REDES NEURONALES CONVOLUCIONALES PARA LA REALIZACIÓN DE TAREAS DE *MAPPING* Y LOCALIZACIÓN

TRABAJO FIN DE GRADO

Junio –2022

AUTORA: Ana María Ortiz Legación

DIRECTOR/ES: Mónica Ballesta Galdeano

Orlando José Céspedes Gómez

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

1 INTRODUCCIÓN

La innovación en la tecnología cada vez es más evidente y es un hecho innegable. Tanto en nuestra vida cotidiana como en diversos campos más específicos, podemos observar cómo el avance de la tecnología nos está proporcionando una mejora sustancial facilitándonos en muchos ámbitos nuestra manera de convivir en sociedad.

Cuando hablamos de las numerosas aplicaciones que nos ha otorgado el avance tecnológico, la robótica móvil ha logrado captar gran parte de nuestra atención cuando se han aplicado en accidentes nucleares, localización de naufragios, exploración de volcanes o incluso viajes espaciales. La exposición del ser humano en este tipo de situaciones resulta muy inconveniente debido al alto riesgo al que se enfrentan. Es por ello por lo que se han desarrollado estos robots móviles que permiten reemplazar al hombre con el fin de realizar las tareas con una mayor seguridad.

1.1 Robótica móvil

Dentro de la robótica móvil debemos destacar tres grupos generales según el tipo de funcionamiento: robótica teleoperada, robótica automática y robótica autónoma. Los robots teleoperados son los robots controlados a distancia por un usuario que se encuentra en una estación remota. En cuanto a los robots automáticos, estos incluyen en su configuración una previa programación, de manera que los robots automáticos realizan la misma tarea ordenada por el usuario en ambientes controlados que no pueden alterar el funcionamiento del robot automático, como puede ser el caso de un brazo robótico que se encuentra en una fábrica y que debe realizar la misma tarea una y otra vez. Por último, la robótica autónoma, cuyo funcionamiento se basa también en una previa programación, pero en este caso los robots autónomos son capaces de tomar decisiones propias operando en entornos naturales que no son modificados, es decir, el robot autónomo es capaz de aprender y ejecutar simultáneamente en función de un algoritmo previamente desarrollado por el usuario [1]. A modo de representación, en la Figura 1.1 se presentan diferentes ejemplos de los tipos de robots móviles según su funcionamiento.

(a) (b) (c)

Figura 1.1: Ejemplos de los diferentes tipos de robótica móvil según su tipo de funcionamiento: (a) robot teleoperado, (b) robot automático y (c) robot autónomo.

Durante este trabajo, haremos hincapié en este último grupo referido a robots autónomos y en cómo su funcionamiento les permite navegar de forma autónoma a través de ambientes y entornos naturales que no han sido previamente modificados y adaptados a los robots.

1.2 SLAM: Tarea de *mapping* y localización

Para que el funcionamiento de la robótica móvil se lleve a cabo, se deben desempeñar diversas tareas. En concreto, el robot móvil debe establecer el modelo del entorno por donde va a realizar la navegación o, dicho de otra forma, el robot debe crear un mapa del lugar en el cual se encuentra. Posteriormente, el robot móvil deberá realizar la tarea de localización, la cual consiste en estimar su posición dentro del modelo del entorno previamente creado en la tarea de *mapping*, de manera que el robot móvil define sus coordenadas y su orientación respecto a dicho mapa.

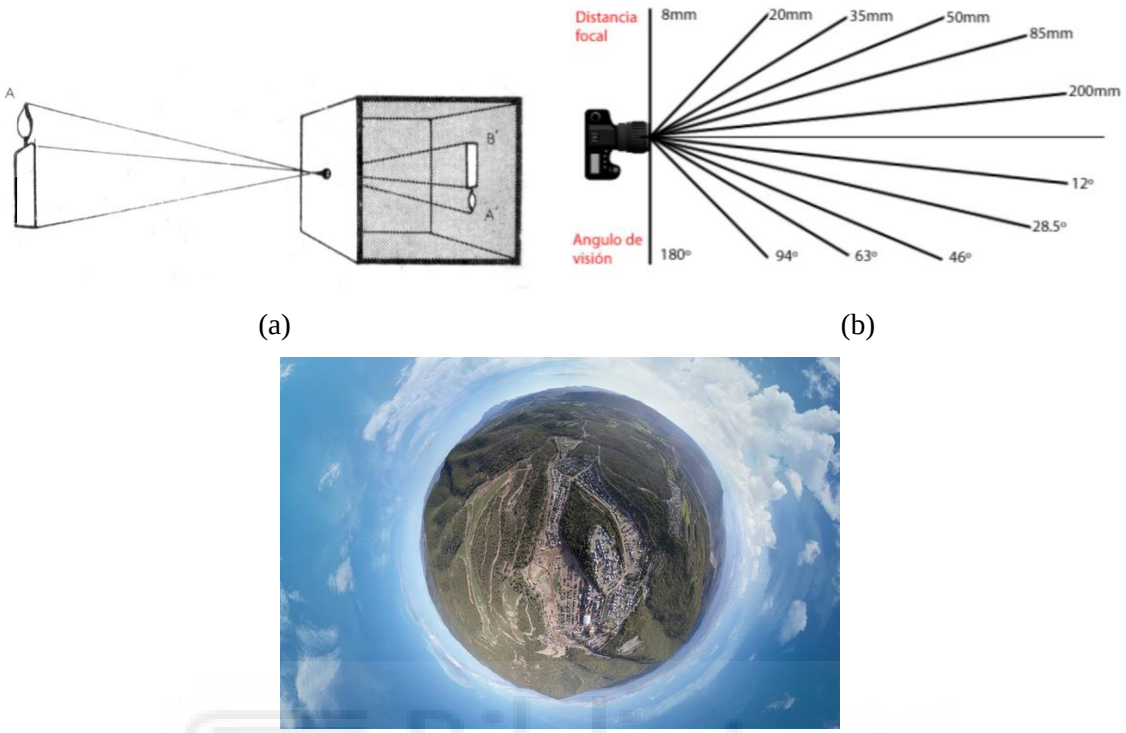
La importancia de la robótica autónoma reside en su versatilidad para la navegación a través de mapas donde no se obtiene una información previa para crearlas, es decir, el robot móvil autónomo es capaz de crear mapas de entornos desconocidos. En estos casos, el robot autónomo debe establecer y crear el mapa mediante la información visual que está recibiendo de su propio equipo sensorial al mismo tiempo que debe estimar su posición respecto a este mapa con objeto de planificar las trayectorias y evadir

obstáculos. A este proceso de tareas simultáneo se le conoce comúnmente como método de localización y mapeo simultáneo, donde sus siglas en inglés determinan el algoritmo de SLAM (*Simultaneous Localization and Mapping*). Este proceso de SLAM se ha convertido en una de las áreas más desafiantes dentro de la robótica autónoma, siendo este proceso de gran importancia para la auto-localización y navegación autónoma.

1.3 Sensores visuales

El problema de SLAM resulta muy complejo de resolver debido a que el ruido de la estimación en la localización del robot induce ruido en la estimación del mapa y viceversa. Según el equipo sensorial que se incorpore en el robot, existen diversas soluciones que permiten resolver la tarea de SLAM. El uso de los sensores láser en dos y tres dimensiones ha sido utilizado por autores como Grisetti *et al*. [2], Hähnel *et al*. [3], Biber *et al*. [4], Eustice *et al*. [5], Triebel and Burgard [6]. Sin embargo, en este trabajo vamos a hacer uso de métodos basados en la visión mediante cámaras como sensores, ya que estos métodos nos otorgan mayor información de una manera más económica y con resultados eficientes, además de ser esta información visual un medio natural que nos permite a los humanos desenvolvernos más fácilmente.

La obtención de información visual se realiza mediante diferentes tipos de sensores los cuales se incorporan en el robot móvil. Según el tipo de lente que lleva incorporado el sensor visual se distinguen tres tipos: cámaras estenopeicas o colineales, las cámaras gran angular y las cámaras omnidireccionales. Por un lado, la cámara estenopeica o colineal se refiere a una cámara fotográfica la cual no presenta ninguna lente, donde su configuración consiste en una caja estanca a la luz integrada por un material sensible que presenta un pequeño orificio por donde la luz se introduce al interior de la caja. Por otro lado, la cámara gran angular presenta una lente cuya distancia focal permite un ángulo de visión que oscila entre 60 y 180 grados. Finalmente, la cámara omnidireccional presenta una lente que ofrece un ángulo de visión de 360 grados, siendo este tipo de lente el más adecuado para las aplicaciones robóticas ya que ofrecen información visual en 360 grados alrededor del robot que lleva integrada dicha cámara omnidireccional. En la Figura 1.2 se muestran unas imágenes que describen el tipo de lente para cada una de las cámaras descritas previamente.



(a) (b)

(c)

Figura 1.2: Sensores visuales según el tipo de lente que llevan incorporado: (a) cámara estenopeica, (b) ángulo de visión de la cámara gran angular y (c) fotografía tomada por una cámara omnidireccional.

1.4 Descriptores visuales

Tras obtener la información visual encontramos un inconveniente ya que, a pesar de brindarnos la ventaja de ofrecer una gran cantidad de información del entorno, debemos adquirir un algoritmo que sea capaz de extraer de una imagen la información visual útil de aquella que no nos aporta nada con el fin de solventar las tareas de mapeo y localización de una forma mucho más efectiva. Así, respecto a esta necesidad de describir y extraer el contenido de la información visual de forma objetiva y automatizada, surgen como respuesta los descriptores.

Concretamente, los descriptores visuales son los encargados de establecer el punto de unión entre los píxeles que comprenden una imagen y las propiedades abstractas que

percibimos los humanos y que establecemos como características determinantes en una imagen concreta.

Para definir con una mejor claridad los descriptores visuales, haremos uso de las propiedades que idealmente deben poseer para realizar las funciones de extracción de información útil de una imagen, donde las propiedades son las siguientes [7]:

- Simplicidad. Esta propiedad consiste en que el descriptor debe realizar su función con la mayor sencillez y candor posibles, de forma que la interpretación de su contenido resulte clara.

- Repetibilidad. Es de gran importancia que el descriptor que se genera a partir de una imagen sea independiente del momento en el que se ha generado, pudiendo ser utilizado en diferentes momentos.

- Diferenciabilidad. El descriptor debe contener suficiente información para relacionar la imagen que está procesando con las imágenes que son similares, de manera que el descriptor debe, simultáneamente, diferenciar la imagen a procesar respecto del resto de imágenes.

- Invariancia. En el caso en el que dos imágenes presenten alteraciones debido a diferentes efectos visuales, el descriptor debe ser capaz de relacionarlos a pesar de que las imágenes estén afectadas por distintas deformaciones.

- Eficiencia: Los descriptores deben tener la capacidad de lograr los resultados deseados en el mínimo posible de recursos con el fin de poder adaptar estos descriptores a aplicaciones que requieran de un tiempo y/o espacio limitado.

Por otro lado, existen diferentes tipos de descriptores según el nivel de abstracción que presenten las características extraídas de la información visual. Dicho de otro modo, encontramos descriptores que no profundizan en la representación del contenido extraído, sino que se encuentran en un nivel más superficial describiendo características esenciales como pueden ser la forma, el color o la textura entre otros. Mientras que existen descriptores con un alto grado de profundidad en cuanto a la abstracción de las

características, llegando a describir información relacionada con los sentimientos o sensaciones, que pueden llegar a ser triviales para un humano, pero no para un descriptor.

Así, según el nivel de abstracción de representación del contenido, podemos clasificar los descriptores en dos grandes grupos:

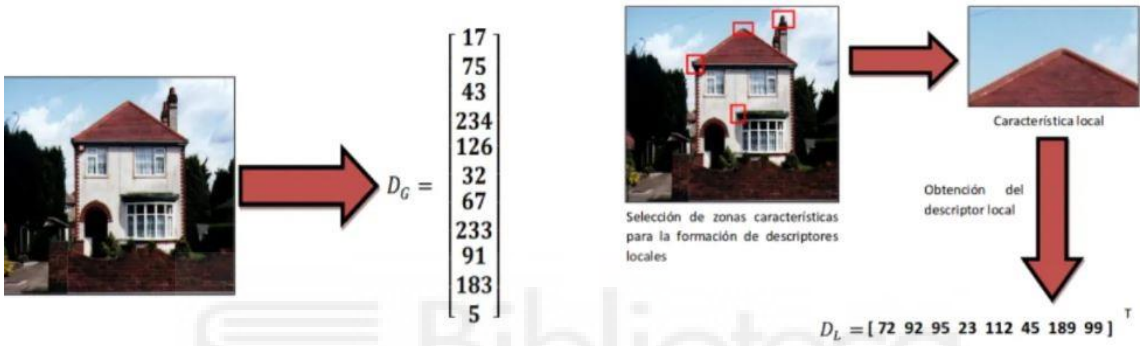
- Descriptores de información general. Este grupo está integrado por los descriptores de bajo nivel que no proporcionan información más allá del color, formas, regiones, texturas y movimientos localizados en la imagen a describir, sin llegar a profundizar en otras características más abstractas.

- Descriptores de información de dominio específico o descriptores semánticos. En este grupo se encuentran los descriptores que presentan un mayor nivel de profundidad, de manera que son capaces de extraer información sobre los objetos y eventos que integran una escena. El funcionamiento de los descriptores de información de dominio específico consiste en utilizar los descriptores de información general para resolver el "gap" que existe entre las características visuales más superficiales y las diferentes categorías semánticas [8].

No obstante, dentro del grupo de descriptores de información general, los descriptores realizan unas determinadas operaciones sobre la imagen con el fin de conseguir el contenido que compone dicho descriptor, pero no todos los descriptores realizan las operaciones en las mismas regiones de la imagen [9]. De este modo, según el nivel de aplicación sobre el que actúan en la imagen, obtenemos la siguiente clasificación junto con su respectiva comparación representada en la Figura 1.3:

- Descriptores basados en características locales. Estos descriptores actúan sobre regiones determinadas y concretas, llamadas puntos de interés o *keypoints,* que son previamente seleccionadas, de manera que el descriptor construye un vector de características de esa misma región con información sobre ese punto de interés y sobre zonas vecinas o adyacentes. Por tanto, el descriptor resultante se compone de todos los vectores de características que han sido calculados en cada una de las regiones de la imagen a describir.

- Descriptores basados en apariencia global o descriptores holísticos. En este caso, los descriptores obtienen un único vector o matriz para la imagen a describir, siendo estos descriptores de gran utilidad ya que son capaces de resumir en una pequeña suma de datos una gran cantidad de información. Las ventajas de obtener un solo vector de características se traducen en un bajo coste computacional con unos resultados más que eficientes, además de presentar ventajas en entornos dinámicos o mal estructurados [10].



(a) (b)

Figura 1.3: Comparativa entre descriptores basados en apariencia global (a) y descriptores basados en características locales (b).

Centraremos este trabajo en este último grupo de descriptores basados en apariencia global, también llamados descriptores holísticos, cuya explicación y clasificación vendrán descritas más detalladamente en el capítulo 2.

1.5 Inteligencia artificial

Como adelanto al capítulo 2, cabe recalcar que todo descriptor debe basar su procesamiento y cálculo de los resultados deseados mediante un método específico. En concreto, este trabajo se va a centrar en los descriptores holísticos basados en métodos de *machine learning*.

El *machine learning* se trata de una disciplina y ámbito científico que se fundamenta en un conjunto de técnicas el cual permite el aprendizaje automático a través

de un previo entrenamiento con grandes volúmenes de datos. Precisamente, el *machine learning* se trata de una rama procedente de la inteligencia artificial, cuya definición viene dada por ser un campo de la ciencia multidisciplinaria que, a través de ciencias como la ciencia de la computación, la lógica, la filosofía y/o las matemáticas, trata de diseñar sistemas capaces de resolver problemas por sí mismos, utilizando como paradigma la inteligencia humana [11].

Es por ello que centramos el objeto de estudio del presente trabajo en este método basado en *machine learning*, haciendo uso de algoritmos de aprendizaje profundo, más concretamente de redes neuronales convolucionales (CNN del término en inglés *Convolutional Neural Networks*) con el fin de obtener descriptores globales de imágenes.

Como introducción a los próximos capítulos, aclarar que las redes neuronales convolucionales permiten llevar a cabo una tarea de clasificación o predicción mediante un exhaustivo entrenamiento previo que se basa en el reconocimiento de patrones permitiéndoles "aprender" a identificar y solucionar la tarea a resolver sin requerir de una programación previa. La idea de estas redes se inspira en las neuronas biológicas reales, las cuales están interconectadas siendo capaces de transmitir información de una capa a otra.

Por otro lado, el *deep learning* se basa en el funcionamiento del *machine learning*, pero con la diferencia de que el *deep learning* implementa redes neuronales convolucionales que comprenden diversos niveles jerárquicos dotándole de una mayor precisión. Los descriptores basados en *deep learning* nos brindan un procesamiento de imágenes más precisa, siendo de gran importancia las ventajas que nos ofrecen como el anticipamiento a los problemas que se deben resolver durante la trayectoria del robot con un buen aprovechamiento de los recursos disponibles y en un menor tiempo de actuación.

1.6 Objetivos

Así, la presente investigación se centra en las redes neuronales convolucionales para la realización de tareas de *mapping* y localización de un robot móvil. En los capítulos posteriores se realiza un estudio acerca de la utilización de descriptores holísticos obtenidos a partir de *deep learning* para la resolución de dichas tareas en entornos de

interior que son cambiantes, comprobando cuán de eficientes y efectivos resultan ser dichos descriptores.

En definitiva, el objeto de estudio del presente trabajo reside en el análisis de descriptores holísticos obtenidos a través de *deep learning*, concretamente de redes neuronales convolucionales, para su aplicación en tareas de *mapping* y localización en la robótica móvil. Adicionalmente, el presente trabajo es una continuación de una investigación anterior [12], donde se realizó un estudio sobre unos determinados descriptores holísticos estudiando su comportamiento en las tareas de *mapping* y localización y obteniendo unos resultados que se presentan detallados en capítulos posteriores.

Por último, la estructura a seguir durante el resto del trabajo queda resumido de la siguiente manera:

- El capítulo 2 se centra en explicar la importancia de las cámaras omnidireccionales para realizar la tarea de *mapping* y localización en la robótica móvil. Adicionalmente, se detallan los diferentes tipos de descriptores que permiten extraer las características esenciales de una imagen.

- Durante el capítulo 3 se detalla de forma más exhaustiva el material y la base de datos que se han utilizado para llevar a cabo la presente investigación, de manera que se describen las imágenes usadas para obtener los descriptores y realizar las tareas de *mapping* y localización. Asimismo, se exponen las diferentes redes neuronales que permiten obtener los descriptores con el fin de conocer su estructura previamente a los experimentos explicados en capítulos posteriores.

- El objeto del capítulo 4 es explicar con detalle el procedimiento que se ha llevado a cabo para los cinco experimentos que han permitido dar a conocer qué descriptor de los estudiados es el que nos ofrece mejores resultados respecto a la tarea de localización.

- Por último, en el capítulo 5 se relatan las conclusiones finales a partir de los resultados obtenidos a lo largo de esta investigación. Además, se concretan cuáles pueden ser algunas de las líneas de investigación futuras

con el fin de obtener un mayor conocimiento en el estado de la técnica al que se refiere el presente trabajo de fin de grado.

2 VISIÓN OMNIDIRECCIONAL

Tal y como se adelantaba en el capítulo anterior, las cámaras omnidireccionales son los sensores visuales más adecuados y eficientes para obtener la información visual en las aplicaciones robóticas, gracias a que son capaces de cubrir un campo de visión de 360 grados alrededor de su eje, ofreciéndonos a su vez unas ventajas adicionales respecto a las cámaras convencionales [10].

Asimismo, las imágenes capturadas por las cámaras omnidireccionales son más estables que las capturadas por otras cámaras ya que permanecen más tiempo en el campo de visión a medida que el robot se desplaza. También ofrecen información suficiente para estimar la posición del robot independientemente de su orientación, gracias a su campo visual de 360 grados. Otra de las ventajas que ofrecen las cámaras omnidireccionales reside en su coste económico, resultando ser rentables económicamente por los resultados que nos ofrecen. Por otro lado, es de vital importancia que el sensor visual que presente el robot móvil sea capaz de soportar largos períodos de tiempo con un consumo no excesivo, siendo las cámaras omnidireccionales una buena solución para ello. Todas estas ventajas hacen que los sistemas de visión omnidireccionales se hayan convertido en la solución más eficiente y popular para resolver las tareas de *mapping* y localización en las aplicaciones robóticas.

2.1 Cámaras omnidireccionales

Como ya se definía en el capítulo correspondiente a la introducción, las cámaras omnidireccionales (donde el significado de *omni-* se define como *todo*) son cámaras capaces de ofrecer un campo de visión de 360 grados en el plano horizontal del entorno. Para abarcar este campo de visión, se emplean diversos tipos de sistemas de visión [13].

Por un lado, los sistemas catadióptricos consiguen alcanzar un campo de visión de 360 grados con una elevación superior a 100 grados. Estos sistemas de visión se obtienen a partir de la combinación de una cámara convencional y de un espejo convexo que puede abarcar diferentes formas: cónica, esférica, parabólica o incluso hiperbólica. En la Figura 2.1 se muestra una representación de un sistema de visión catadióptrico haciendo uso de un espejo hiperbólico. Asimismo, los sistemas catadióptricos también

pueden obtenerse a partir de la utilización de lentes como el de ojo de pez tal y como divulgan J. Kumler y M. Bauer [14].

Por otro lado, los sistemas de visión polidióptricos están constituidos por numerosas cámaras que superponen su campo de visión con el fin de alcanzar un campo visual total del entorno. En efecto, este sistema es el único que cumple con el campo de visión omnidireccional real, siendo totalmente esférico y abarcando cualquier punto de visión, tal y como se observa en la Figura 2.1.

(a) (b)

Figura 2.1: Cámara catadióptrica (a) y cámara polidióptrica (b).

Los sistemas de visión catadióptricos deben presentar un único punto de vista [15] [16], lo que quiere decir que en las imágenes omnidireccionales existe un único centro de proyección. Este centro único permite que todos los píxeles de las imágenes omnidireccionales capturadas por los sistemas catadióptricos sean capaces de medir la irradiancia de luz que pasa a través de dicho punto céntrico para cualquier dirección particular. Adicionalmente, utilizando el punto céntrico de las imágenes

omnidireccionales se pueden implementar diversas transformaciones que dan como resultado otras perspectivas o proyecciones según la tarea que se desee desempeñar como pueden ser: omnidireccional, panorámica o vista en planta [17] [18]. No obstante, durante este trabajo haremos uso de las representaciones omnidireccional y panorámica, ya que nos aportan la suficiente información para resolver las tareas de *mapping* y localización.

Para llevar a cabo la presente investigación, el robot móvil llevará incorporado como sensor visual un sistema de visión catadióptrico, permitiéndole capturar las imágenes omnidireccionales de su entorno, cuya configuración queda explicada más detalladamente en el siguiente capítulo.

2.2 Descriptores de apariencia global

La información visual obtenida mediante los sensores correspondientes contiene gran cantidad de información que debemos seleccionar y extraer de la información que no es necesaria para la tarea que deseamos resolver. Para ello, hacemos uso de los descriptores, cuya clasificación se divide en descriptores basados en características locales y descriptores basados en apariencia global, también llamados descriptores holísticos.

Los descriptores holísticos presentan diversas ventajas respecto a los descriptores basados en características locales, ya que ofrecen un único vector de características que reúne toda la información en una pequeña suma de datos. El hecho de utilizar un solo vector permite que el coste computacional se reduzca frente al utilizado por los descriptores basados en características locales. Adicionalmente, los descriptores holísticos son más estables en entornos dinámicos o mal estructurados [10]. Sin embargo, dentro de este grupo de descriptores cabe destacar que según el método que utilizan para ser calculados existen dos subgrupos: los descriptores basados en métodos analíticos y los descriptores basados en *deep learning*.

2.2.1 Descriptores holísticos basados en métodos analíticos

Originalmente, los métodos analíticos constituyen la herramienta más habitual para la obtención de descriptores holísticos en la tarea de localización y navegación dentro de la robótica móvil. Para llevar a cabo estos métodos, se deben calcular gradientes y orientaciones de cada uno de los píxeles que componen la imagen a describir, de manera que los métodos analíticos a través de una imagen implementan transformaciones matemáticas obteniendo así un único vector ($\vec{d} \in \mathbb{R}^{lx1}$) de características. Entre los diferentes descriptores analíticos existentes, debemos destacar HOG y GIST, cuyos descriptores fueron analizados en la investigación previa a este trabajo [12].

Por un lado, el descriptor HOG (*Histograms of Oriented Gradients*) es un tipo de descriptor de características que se utiliza tanto en visión por computador como en procesamiento de imágenes con el fin de solventar la detección de objetos, cuyo descriptor fue presentado en [19] en el año 2005.

Por otro lado, GIST se refiere a un descriptor que permite caracterizar una imagen mediante un descriptor de dimensiones reducidas conteniendo a su vez, suficiente información para identificar la escena de la imagen. Dicho descriptor fue introducido en en [20] en el año 2001.

A pesar de que estos descriptores basados en métodos analíticos han llegado a soluciones más que eficientes, durante los últimos años se han empezado a introducir en el mundo de la robótica móvil los descriptores basados en *deep learning* para resolver tareas de visión por computador.

2.2.2 Descriptores holísticos basados en *deep learning*

A medida que se llevan a cabo nuevas investigaciones, se hace más frecuente el uso de las técnicas de *deep learning* para resolver tareas de navegación en robótica móvil. Esto se debe a que su utilización permite la ventaja de poner el foco en un tipo de imagen específica, dotando a los descriptores de una mayor eficiencia. No obstante, los descriptores basados en *deep learning* requieren de un gran procesamiento de datos y un elevado tiempo de cómputo debido a que deben pasar por una etapa previa de entrenamiento.

Dentro de las técnicas de *deep learning* empleadas para la construcción de descriptores holísticos encontramos los *autoencoders*, los cuales son un tipo de red neuronal artificial que permite el aprendizaje de codificaciones de datos eficientes de manera no supervisada. No obstante, la investigación del presente trabajo se centra en los descriptores holísticos que utilizan como herramienta para su cálculo las redes neuronales convolucionales (CNNs). En este sentido, las CNNs permiten resolver la tarea de clasificación y predicción a través de un exhaustivo entrenamiento previo con el fin de extraer las características principales de una imagen.

Para entender mejor la configuración de las CNNs y su funcionamiento es vital comprender cómo están estructuradas y la función que desempeñan las principales capas que las componen. En primer lugar, una red neuronal convolucional debe contener capas convolucionales a las que se le atribuyen la mayor parte de las operaciones matemáticas que lleva a cabo la CNN, constituyendo así, el bloque principal de la estructura. Concretamente, estas capas realizan la operación matemática llamada convolución que se presenta en la Figura 2.2, la cual consiste en una fórmula matemática que integra diferentes multiplicaciones y sumas según el filtro que se desea aplicar.

$$\text{out}(N_i, C_{\text{out}_j}) = \text{bias}(C_{\text{out}_j}) + \sum_{k=0}^{C_{\text{in}}-1} \text{weight}(C_{\text{out}_j}, k) \star \text{input}(N_i, k)$$

Figura 2.2: Expresión matemática de la convolución [21].

A modo representativo, en la Figura 2.3 se muestra la aplicación de un determinado filtro a una entrada de tamaño 6x6x1, obteniéndose como salida una matriz de tamaño 4x4x1.

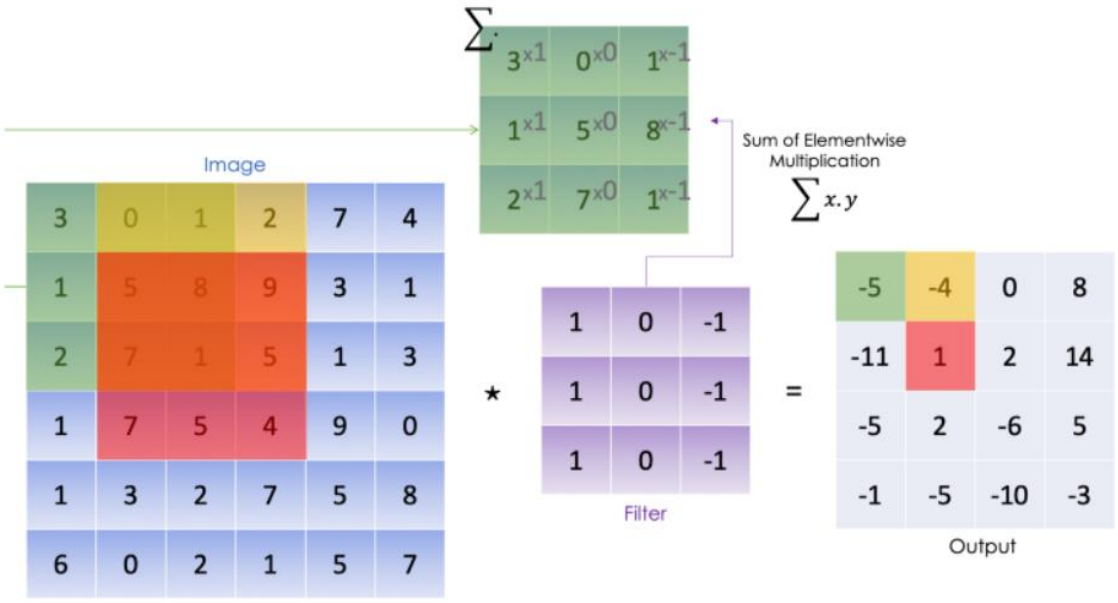


Figura 2.3: Representación gráfica del efecto obtenido por la expresión matemática llamada convolución [22].

Continuando con la estructura de la CNN encontramos las capas de agrupación, las cuales consisten en un submuestreo que reduce las dimensiones de la función conforme se va profundizando en la red. Estas capas de agrupación junto con las capas convolucionales constituyen a grandes rasgos las capas para el aprendizaje de características. Por último, las capas completamente conectadas o *Fully Connected* (FC) se encargan de implementar las conexiones entre cada una de las neuronas, de manera que dichas conexiones se traducen en los parámetros característicos que permiten la clasificación de la imagen.

En definitiva, el funcionamiento de una CNN consiste en introducir como entrada a la red una imagen para la obtención de un vector de características de una de las capas intermedias, siendo este vector el descriptor holístico. Así, durante la presente investigación, se estudiarán las redes ResNet y VGGNet, que vendrán explicadas de una forma más detallada en el capítulo 3. Para el estudio de dichas CNNs, se han llevado a cabo una serie de experimentos que permiten concluir su fiabilidad y robustez ante la resolución de tareas de *mapping* y localización en la robótica móvil.

3 MATERIAL Y MÉTODOS

Durante este apartado, se detallarán tanto la base de datos en la que se ha respaldado la presente investigación como la estructura de cada una de las redes que son el objeto de estudio, además del material utilizado que permite llevar a cabo los experimentos para definir la robustez de las redes VGGNet y ResNet en la tarea de SLAM.

3.1 Base de datos empleada

En primer lugar, durante la presente investigación debemos adquirir la información visual necesaria para el estudio de los descriptores holísticos basados en *deep learning*. Concretamente, se va a hacer uso de la base de datos llamada COLD *database* [23], cuyo acrónimo viene dado por el término en inglés *COsy* (*Cognitive Systems for Cognitive Assistants*) *Localization Database*. Esta base de datos proporciona un entorno de prueba flexible a gran escala que permite evaluar y resolver sistemas de localización basados en la visión, de manera que la base de datos COLD se compone de imágenes omnidireccionales que son captadas en entornos de interior. Asimismo, esta base de datos está integrada a su vez de tres *datasets* que han sido adquiridos en tres entornos de interior diferentes y que están ubicados en tres ciudades europeas: (a) el *Visual Cognitive Systems Laboratory* en la universidad de Ljubljana, Eslovenia; (b) el *Autonomous Intelligent Systems Laboratory* en la universidad de Friburgo, Alemania; (c) y el *Language Technology Laboratory* en el Centro Alemán de Investigación en Inteligencia Artificial de Saarbrücken, Alemania. No obstante, para facilitar la referencia a cada uno de estos *datasets*, haremos uso de la ciudad en la cual tuvieron lugar las adquisiciones de imágenes: COLD – Ljubljiana, COLD – Friburgo y COLD – Saarbrücken.

La base de datos elegida para el presente estudio ha sido COLD – Saarbrücken. Esto se debe a que el presente trabajo de fin de grado sigue la línea de investigación del trabajo previo [12], cuyo estudio se basa en este *dataset*, pudiendo contrastar así los resultados entre los diferentes descriptores obtenidos basados en *deep learning*.

Por otro lado, la adquisición de imágenes se lleva a cabo mediante una plataforma robótica móvil, cuyo desplazamiento a través de las distintas estancias se configura con un joystick, de forma que un equipo sensorial integrado en la plataforma robótica móvil permite captar las imágenes necesarias. Concretamente, para la base de datos COLD – Saarbrücken la plataforma robótica móvil corresponde con el modelo *ActivMedia PeopleBot* representado en la Figura 3.1. El equipo sensorial que lleva incorporado dicho robot se compone de escáneres láser tipo SICK y encoders en las ruedas tipo SICK, los cuales ofrecen una medida de distancia escalar desde el sensor al objeto con una precisión óptima, además de que permiten medir el giro de las ruedas para calcular la posición y velocidad del robot móvil. Gracias a este equipo sensorial, se conoce con exactitud la posición real o *ground truth* de la plataforma robótica móvil en cada instante. Sin embargo, no haremos uso de los datos aportados por estos sensores ya que no son el objeto de estudio de la presente investigación.

(a) (b)

Figura 3.1: Plataforma robótica móvil usada en el *dataset* de COLD – Saarbrücken (a) y detalle de la configuración de cámaras que lleva integrada (b). Imágenes obtenidas de COLD *database* [23].

Las imágenes capturadas se obtienen mediante dos cámaras digitales *Videre Design* MDCS2 tal y como se observa en la Figura 3.1, donde una de ellas adquiere imágenes en perspectiva y la otra, imágenes omnidireccionales. Así, la cámara digital que obtiene las imágenes omnidireccionales forma parte de un sistema catadióptrico que incluye un espejo hiperbólico. De forma más detallada, en la Tabla 3.1 se muestra la

configuración y parámetros que caracterizan al modelo *ActivMedia PeopleBot* y al sistema de cámaras que lleva integrado.

Plataforma robótica	ActivMedia PeopleBot	
Tipo de cámara	Perspectiva	Omnidireccional
Ratio	5 fotogramas por segundo	
Resolución	640 × 480 píxeles, filtro de Bayer	
Exposición	Automático	
Campo de visión	68,9º × 54,4º	---
Altura	140cm	116cm

Tabla 3.1: Configuración y parámetros de las cámaras incorporadas en el *ActivMedia PeopleBot.*

El procedimiento que se lleva a cabo para la adquisición de imágenes es el mismo para todos los *datasets*. Concretamente, el robot móvil se desplaza (controlado por un joystick) a través de las diferentes habitaciones que componen el laboratorio con una velocidad de 0,3 m/s y capturando 5 fotogramas por segundo, es decir, cada 6 cm recorridos el robot móvil captura una imagen del entorno. A continuación, en la Figura 3.2 se muestran algunas de las imágenes capturadas por el equipo visual que se encuentra incorporado en la plataforma robótica móvil, tanto en representación omnidireccional como en vista en planta.

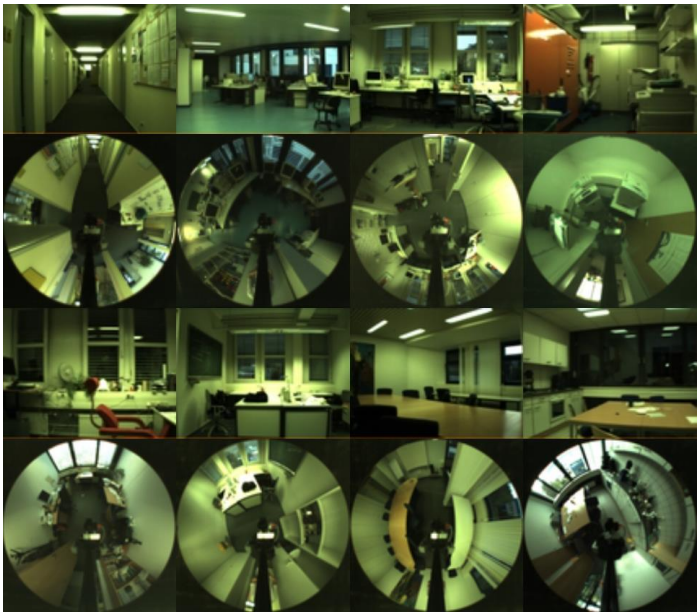
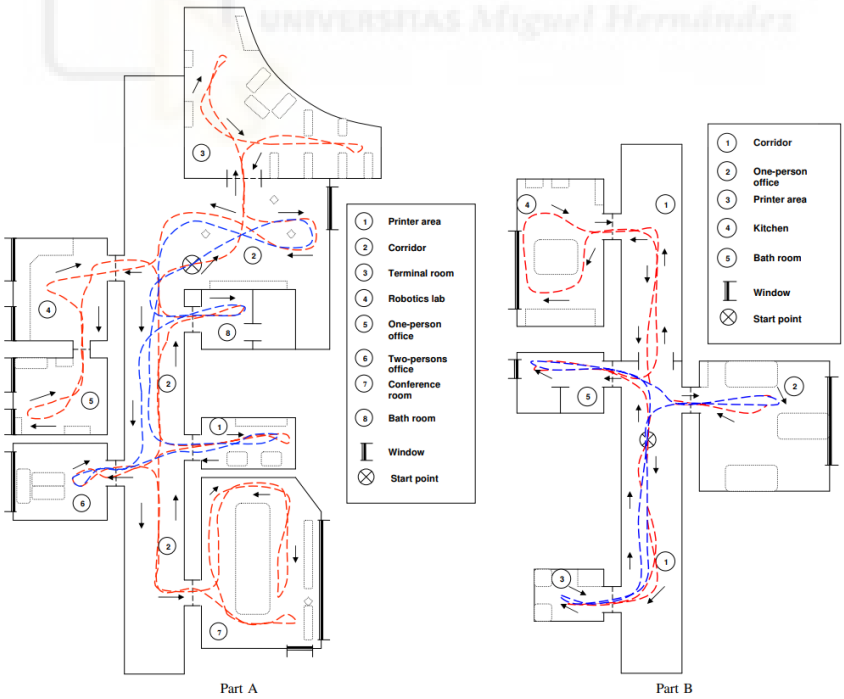


Figura 3.2: Imágenes capturadas en perspectiva y omnidireccionales. Imágenes obtenidas de COLD *database* [23].

Para llevar a cabo la adquisición de imágenes, el laboratorio de Saarbrücken ha sido dividido en dos entornos diferentes denominados ‘Parte A’ y ‘Parte B’ los cuales quedan reflejados en la Figura 3.3. A su vez, el robot móvil diferencia dos tipos de trayectorias en cada uno de los entornos en los que queda dividido el laboratorio. Por un lado, la primera trayectoria representada en color rojo que constituye el camino más corto donde el robot móvil recorre únicamente las habitaciones más comunes que se pueden encontrar en un laboratorio estándar y, por otro lado, la segunda trayectoria representada en color azul que corresponde al camino más extenso donde el robot toma imágenes de todas las estancias que se encuentran en el laboratorio. Asimismo, la obtención de la información visual se realiza bajo diversas condiciones de tiempo e iluminación, dando lugar a tres grupos distintos de imágenes: tiempo soleado, tiempo nublado y noche.

Durante los experimentos se hace uso del entorno correspondiente a la ‘Parte B’ ya que de los entornos a elegir es el único que contiene los tres tipos de iluminación que permite realizar un estudio más exhaustivo. Además, se escoge la trayectoria representada en color rojo ya que nos ofrece una mayor información.



(a) (b)

Figura 3.3: Mapas de los entornos recorridos por el robot móvil a través del laboratorio de Saarbrücken. Imágenes obtenidas de COLD *database* [23].

Una vez tenemos la base de datos seleccionada para el presente estudio, se debe realizar una conversión de formato de cada una de las imágenes contenidas en el *dataset* COLD – Saarbrücken. Concretamente, debemos transformar las imágenes omnidireccionales RGB a imágenes panorámicas en escala de grises ya que, a pesar de que los descriptores holísticos basados en deep learning pueden trabajar perfectamente con imágenes omnidireccionales RGB, los descriptores holísticos basados en *deep learning* obtenidos en [12] se han desarrollado con imágenes panorámicas en escala de grises. Por ello, para obtener una comparación efectiva debemos partir del mismo formato de imágenes: panorámicas en escala de grises.

Dicha conversión se muestra en la Figura 3.4 y se realiza para cada una de las imágenes contenidas en los grupos de datos según la iluminación (nublado, soleado y noche) de la base de datos COLD - Saarbrücken. En primer lugar, se deben eliminar las porciones de las imágenes que no aportan información, es decir, los marcos negros del contorno. Posteriormente, mediante el parámetro del centro de la imagen se obtiene la imagen deseada en formato panorámico y escala de grises.

Figura 3.4: Conversión de imagen omnidireccional RGB a imagen panorámica en escala de grises.

Finalmente, la tarea de SLAM requiere de dos tareas: tarea de *mapping* y tarea de localización. Por ello, debemos obtener un cuarto *dataset* que sería el obtenido en la tarea de *mapping* y con el que compararemos las imágenes test durante la tarea de localización. Para la creación de este *dataset* de entrenamiento plantearemos el mismo procedimiento llevado a cabo en el trabajo previo [12], de manera que se escogen una de cada cinco imágenes del grupo de la base de datos correspondiente al tiempo nublado, originando imágenes cada 30 cm aproximadamente.

3.2 Redes neuronales utilizadas

La obtención de los descriptores basados en apariencia global está fundamentada en las redes neuronales convolucionales que constituyen una de las técnicas con gran potencial y que está ganando popularidad a lo largo de los años. En concreto, haremos uso de la CNN VGGNet y la CNN ResNet, las cuales son redes que ya han sido entrenadas previamente por sus creadores y que no requieren por tanto de un entrenamiento previo durante este estudio.

3.2.1 CNN VGGNet

En primer lugar, el nombre de la CNN VGGNet proviene del término en inglés *Visual Geometry Group* y fue desarrollado por A. Zisserman y K. Simonyan [24] en 2014 con el fin de desarrollar y aumentar el rendimiento de las redes neuronales convolucionales conocidas. Concretamente, la CNN VGGNet puede encontrarse como VGG-16 o VGG-19. Durante la presente investigación se ha tomado la decisión de trabajar con la variante VGG-19 ya que dicha red proporciona una mayor capacidad para adaptarse a funciones más complejas gracias a la estructura que presenta con un mayor número de capas. Esta CNN VGG-19 ha sido previamente entrenada con más de un millón de imágenes pudiendo clasificar hasta 1.000 posibles categorías de objetos, como pueden ser teclados, lápices y numerosos animales entre otros, siendo la entrada a esta red una imagen RGB de $224 \times 224 \times 3$.

La arquitectura de la CNN VGGNet-19 se representa en la Figura 3.5 con el fin de entender con una mayor exactitud su funcionamiento. En este sentido, la red VGG-19 está constituida por 47 capas en total de las cuales 16 corresponden a capas convolucionales que se encuentran agrupadas, de manera que cada agrupación va seguida de una capa *maxpool* que permite reducir su tamaño. La red finaliza con tres capas completamente conectadas o FC y con una capa *softmax* la cual es la encargada te contener las 1.000 categorías posibles de objetos.

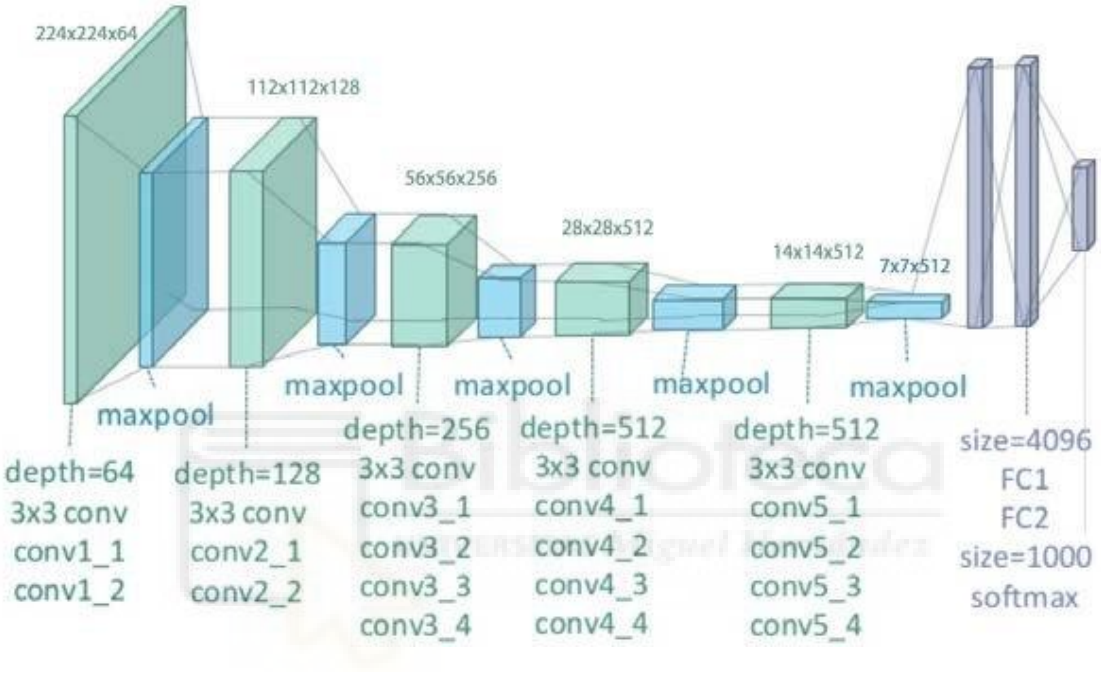


Figura 3.5: Arquitectura de la CNN VGGNet-19.

Las capas convolucionales, a su vez, están integradas por canales que se refieren a las múltiples capas internas que están presentes en cada capa convolucional. Durante el presente estudio trabajaremos con las capas convolucionales que se detallan en la Tabla 3.2, de manera que se han escogido las primeras capas convolucionales de cada una de las agrupaciones.

Capa	Número de canales
conv1_1	64
conv2_1	128
conv3_1	256
conv4_1	512
conv5_1	512

Tabla 3.2: Capas utilizadas de la CNN VGGNet-19.

3.2.2 CNN ResNet

Por otro lado, la CNN ResNet es una red neuronal convolucional que debe su nombre al término en inglés *Residual Network* y que fue introducida por Microsoft, ganando en la competición ILSVRC (ImageNet Large Scale Visual Recognition Challenge) en el año 2015 [25]. A grandes escalas, la arquitectura de esta red neuronal se basa en el aumento del número de capas mediante la introducción de una conexión residual, la cual se inspira en las neuronas biológicas que se conectan con otras neuronas en capas que no son necesariamente contiguas, realizando saltos a capas intermedias. Así, la CNN ResNet también presenta diversas variantes de las cuales las principales son las siguientes: ResNet-18, ResNet-50 y ResNet-101. Sin embargo, se ha escogido la CNN ResNet-50 debido a que es la red más vibrante porque presenta una mayor eficiencia respecto a las demás redes presentando un error de 3,57%. Asimismo, la CNN ResNet-50 ha sido entrenada con más de un millón de imágenes y presenta un total de 177 capas. En este sentido, la arquitectura de la red que se representa en la Figura 3.6, está constituida por cinco etapas, cada una de ellas con un bloque de convolución e identidad, siendo esta última la que implementa las conexiones residuales o de salto (*skip*). Así, de las 177 capas que componen la CNN-ResNet-50, son 53 las que corresponden a las capas convolucionales y 1 capa la que corresponde con una completamente conectada o FC. En la Tabla 3.3 se muestran las capas convolucionales que usaremos en los próximos experimentos junto con el número de canales que componen cada una de ellas, de forma que se han escogido las primeras capas convolucionales de cada una de las 5 etapas.

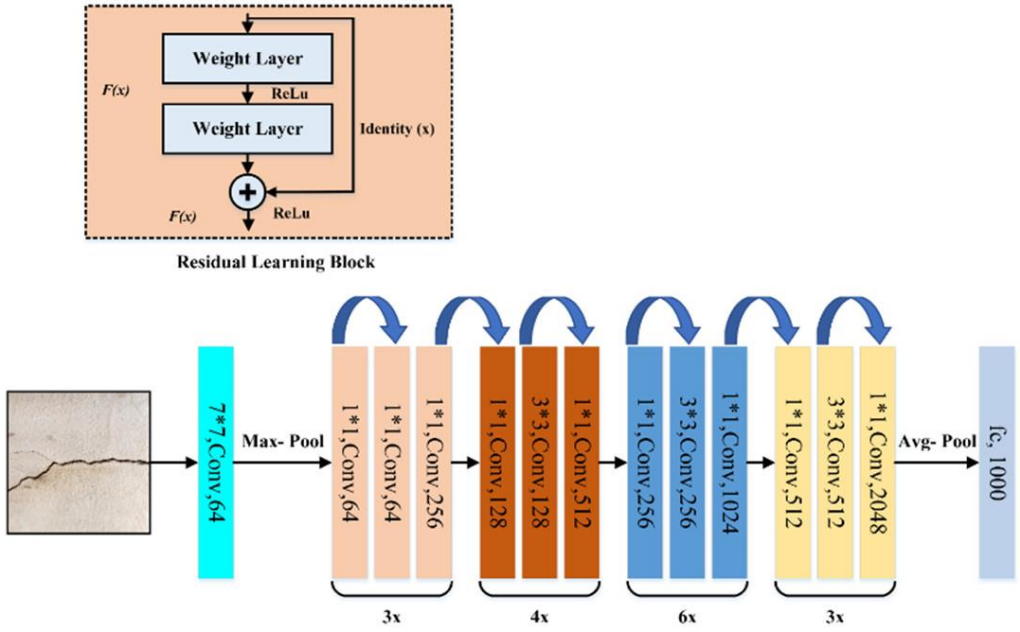


Figura 3.6: Arquitectura de la CNN ResNet-50.

Capa	Número de canales
conv1	64
res2a_branch2a	64
res3a_branch2a	128
res4a_branch2a	256
res5a_branch2a	512

Tabla 3.3: Capas utilizadas de la CNN ResNet-50.

Con todo ello, ya podemos abrir paso a los experimentos donde se investiga la eficiencia del uso de estas redes neuronales convolucionales como técnica para construir un descriptor holístico y comprobar cuán de eficacia nos proporcionan en la resolución de las tareas de *mapping* y localización (SLAM) para la robótica móvil.

4 EXPERIMENTOS

Durante este apartado, nos adentraremos en los experimentos que se van a llevar a cabo con el fin de comprobar y verificar la robustez y eficiencia de cada uno de los descriptores globales obtenidos a partir de las redes neuronales convolucionales VGGNet y ResNet. En este sentido, se han realizado cinco experimentos bajo diferentes efectos que son aplicados en el modelo visual. Se hará uso de cambios de variación en la iluminación, de perturbaciones en la imagen como el *blur*, el ruido o las oclusiones, variaciones de la estructura de la imagen a la entrada de la CNN y posibles rotaciones de la imagen debido a cambios en la orientación con la que el robot móvil capta las imágenes.

4.1 Localización visual

Para llevar a cabo la tarea de *mapping* y localización (SLAM) previamente se ha de realizar la tarea de mapeo obteniéndose un modelo visual y, una vez se resuelve esta tarea, se utiliza el sistema de *image retrieval* para la resolución de la tarea de localización. Un sistema de *image retrieval* es un método que consiste en la exploración, búsqueda y recuperación de imágenes en una base de datos de imágenes digitales. Así, el principal objetivo de la tarea de localización es obtener de la base de datos de entrenamiento la imagen que presente mayor semejanza con respecto a la nueva imagen capturada con el fin de estimar la posición y localización del robot móvil en el modelo visual previamente creado.

Mediante la tarea de *mapping*, se construyen los descriptores de apariencia global basados en *deep learning* que caracterizan a cada una de las imágenes de entrenamiento. En la presente investigación, las imágenes de entrenamiento son las que constituyen el cuarto *dataset* que resulta de una selección del *dataset* correspondiente al tiempo nublado. Posteriormente, el robot móvil debe ser capaz de estimar su posición y orientación mediante la tarea de localización, la cual consta de diversas etapas. En primer lugar, (1) el robot móvil captura una imagen del entorno del cual se ha creado previamente el modelo visual. Estas imágenes serán en nuestro caso los tres *datasets* de test: nublado, soleado y noche. Posteriormente, (2) se realiza una conversión de la imagen de test a analizar, de manera que la imagen pasa de ser omnidireccional RGB a panorámica en

escala de grises, tal y como se ha explicado en capítulos anteriores. Tras esto, (3) se calcula el descriptor correspondiente a dicha imagen test, de forma que se puedan (4) comparar el descriptor de test con cada uno de los descriptores del modelo visual. Para esta comparación, Cebollada et al. [26] propone calcular la distancia coseno entre el descriptor test y cada uno de los descriptores del modelo visual creado previamente obteniéndose un vector de distancias de igual tamaño a las imágenes de entrenamiento. El siguiente paso consiste en (5) obtener el nodo que constituye la distancia mínima entre la imagen test y las imágenes de entrenamiento del modelo visual.

La herramienta a utilizar para resolver la tarea de SLAM del robot móvil corresponde al sistema de cómputo Matlab. Como ya se ha mencionado, en primer lugar se debe crear el modelo visual del entorno y, para ello, debemos construir los descriptores de cada una de las imágenes de entrenamiento. Las imágenes de entrenamiento se encuentran en formato panorámico en escala de grises con un tamaño de $128 \times 512 \times 1$ debido a que previamente han sido sometidas a una conversión. Sin embargo, las redes neuronales convolucionales con las que vamos a trabajar en este estudio solo admiten imágenes en la entrada de un tamaño $224 \times 224 \times 3$. Por ello, además de redimensionar las imágenes se debe triplicar su único canal obteniéndose tres canales superpuestos que simulan la configuración de una imagen RGB. En la Tabla 4.1 se muestra el código que permite redimensionar la imagen y triplicar el único canal con el fin de obtener tres canales. Llegados a este punto, las imágenes presentan el tamaño adecuado para ser introducidas en cada una de las redes neuronales convolucionales.

```
image = imresize(image,[224 224]);
image = image(:,:,[1 1 1]);
```

Tabla 4.1: Línea de código para redimensionar y triplicar el canal de la imagen en escala de grises.

A la hora de introducir una imagen en la red neuronal convolucional, se debe indicar también la capa de la CNN con la que se construye el descriptor holístico que caracteriza dicha imagen. Las siguientes líneas de código de la Tabla 4.2 corresponden al procedimiento que permite cargar en concreto la CNN VGGNet-19 con la imagen a introducir y la capa con la que se construye el descriptor. No obstante, dichos códigos

funcionan para todas las CNNs y capas a implementar, por lo que este código supone un mero ejemplo de aplicación.

```
net = vgg19();
layer = 'conv2_1';
descriptor = activations(net,imagen,layer);
```

Tabla 4.2: Líneas de código para la implementación de la imagen y la capa en la CNN con el fin de obtener el descriptor.

En el proceso de localización, se debe realizar una comparación entre los diferentes descriptores de cada una de las imágenes de entrenamiento y el descriptor de la imagen capturada de test. Para ello, durante los cinco experimentos que se van a llevar a cabo, haremos uso de tres parámetros que permiten medir la eficiencia de la tarea de localización los cuales se detallan a continuación:

- El primer parámetro que se compara se refiere al tiempo de cálculo del descriptor, el cual nos indica el tiempo que se necesita para la construcción del descriptor. Este parámetro corresponde al tiempo transcurrido desde el momento en el que se introduce la imagen y la capa a utilizar para la construcción del descriptor hasta el momento en el que la CNN nos devuelve el descriptor ya construido.

- Por otro lado, el segundo parámetro corresponde al tiempo de cálculo de la estimación de la posición, que es el tiempo transcurrido desde que se obtiene el descriptor hasta adquirir la estimación del robot en el modelo visual. Para llevar a cabo la estimación de la localización del robot móvil, se realiza una comparación entre el descriptor de la imagen test y cada uno de los descriptores de las imágenes de entrenamiento, obteniéndose así un único vector constituido por cada una de las distancias entre las posiciones de las imágenes de entrenamiento y la posición obtenida de la imagen test. Concretamente, esta distancia corresponde con la distancia '*cosine*', cuya fórmula matemática viene representada en la Tabla 4.3. Este parámetro de tiempo pone su fin cuando se obtiene el valor mínimo del vector

constituido por distancias, de forma que dicho valor corresponde con la posición estimada del robot móvil en el modelo visual.

- El tercer parámetro que se compara corresponde con el error de la localización que nos indica la diferencia entre la posición estimada por el método de localización propuesto y la posición real del robot móvil donde se capturó la imagen test. Esta diferencia es posible calcularla debido a que la base de datos COLD ofrece la localización real del robot al llevar incorporado un equipo sensorial de escáneres láser tipo SICK. La comparación de ambas localizaciones se lleva a cabo mediante la distancia '*euclidean*' presentada en la Tabla 4.3.

Distancia	Fórmula matemática
Cosine	$d_{st} = 1 - \frac{x_s x'_t}{\sqrt{(x_s x'_t)(x_t x'_t)}}$
Euclidean	$d_{st}^2 = (x_s - x_t)(x_s - x_t)'$

Tabla 4.3: Fórmulas matemáticas de las distancias utilizadas en la comparación de descriptores.

Finalmente, una vez obtenidos los parámetros de cada uno de los descriptores, se obtendrán sus respectivos valores medios para su representación mediante gráficas que permitirán realizar una comparación más exhaustiva entre los descriptores determinando así, la eficiencia y eficacia de los descriptores holísticos basados en las redes neuronales convolucionales propuestas durante la tarea de *mapping* y localización (SLAM) de un robot móvil.

4.2 Experimento 1: Comparativa entre las diferentes capas de las CNNs

El primer experimento consiste en la investigación sobre la tarea de *mapping* y localización ante diferentes efectos de iluminación en el entorno. Concretamente, para cada una de las CNNs VGGNet-19 y ResNet-50, estudiaremos los descriptores holísticos obtenidos a partir de las capas correspondientes al aprendizaje de características o *feature learning*. Asimismo, haremos uso de las capas de clasificación cuya estructura se compone de un único canal de clasificación.

Respecto a las capas de aprendizaje de características, estas están compuestas por diversos canales que inducen a una salida devolviendo una matriz por cada canal. Es por ello que se debe elegir el canal que da mejores resultados, de manera que en la salida se obtenga una matriz que se pueda reordenar en un solo vector. La elección del mejor canal vendrá dada por el menor error de localización, obteniéndose así unos resultados más eficientes. Por otro lado, se va a trabajar simplemente con el *dataset* correspondiente al efecto de iluminación de nublado, ya que para los demás cambios de iluminación de noche y soleado, usaremos desde un principio el canal que mejores resultados se han obtenido en el *dataset* de nublado.

A continuación, se exponen los resultados tanto para la CNN VGGNet-19 como para la CNN ResNet-50, de manera que para cada una de las CNNs estudiadas se realiza una comparativa entre las diferentes capas de aprendizaje de características y las capas de clasificación, de forma que la comparación se basa en los parámetros previamente explicados: (1) el tiempo de cálculo del descriptor, (2) el tiempo de cálculo de la estimación de la posición y (3) el error de la localización.

4.2.1 Comparativa entre las capas de la CNN VGGNet-19

En primer lugar, se estudia la CNN VGGNet-19, cuya estructura está dividida en agrupaciones tal y como se explica en capítulos anteriores. Para el presente estudio, se hace uso de la primera capa de aprendizaje de características de cada una de las agrupaciones que la componen. En la siguiente Tabla 4.4 se representan los canales que ofrecen un menor error de localización para cada una de las capas convolucionales estudiadas junto al tamaño del descriptor holístico obtenido en forma de vector.

Nombre de la capa	Mejor canal	Tamaño del descriptor
conv1_1	37	50176
conv2_1	124	12544
conv3_1	228	3136
conv4_1	260	784
conv5_1	284	196

Tabla 4.4: Mejor canal obtenido para cada una de las capas convolucionales estudiadas en la CNN VGGNet-19.

Una vez obtenido el canal con los mejores resultados para cada una de las capas estudiadas, se procede a realizar la tarea de localización con las bases de datos de test correspondientes a los diferentes efectos de iluminación: nublado, soleado y noche. Mediante los tres parámetros que se obtienen de dicho análisis, será posible realizar una comparación exhaustiva y obtener unas gráficas que representan de una forma más detallada la eficiencia de cada una de las capas bajo diferentes efectos de iluminación.

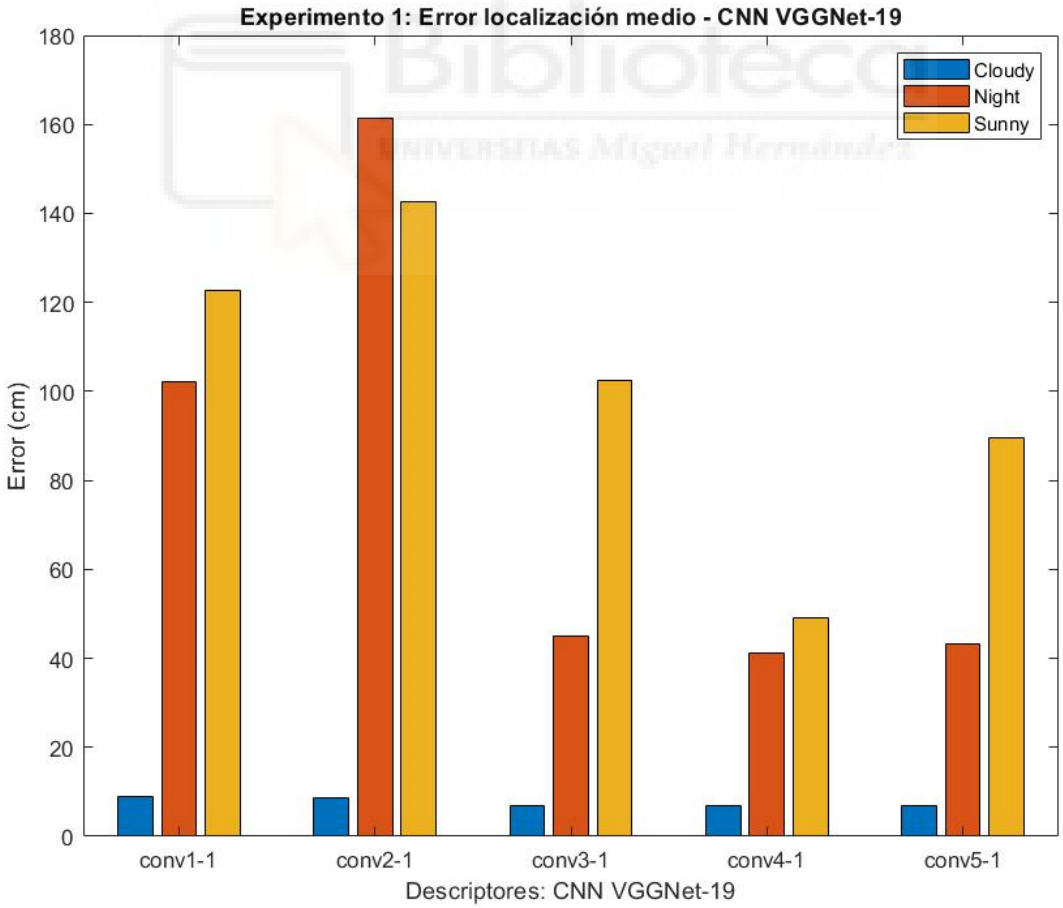


Figura 4.1: Error medio de localización de los descriptores obtenidos de las capas convolucionales de la CNN VGGNet-19.

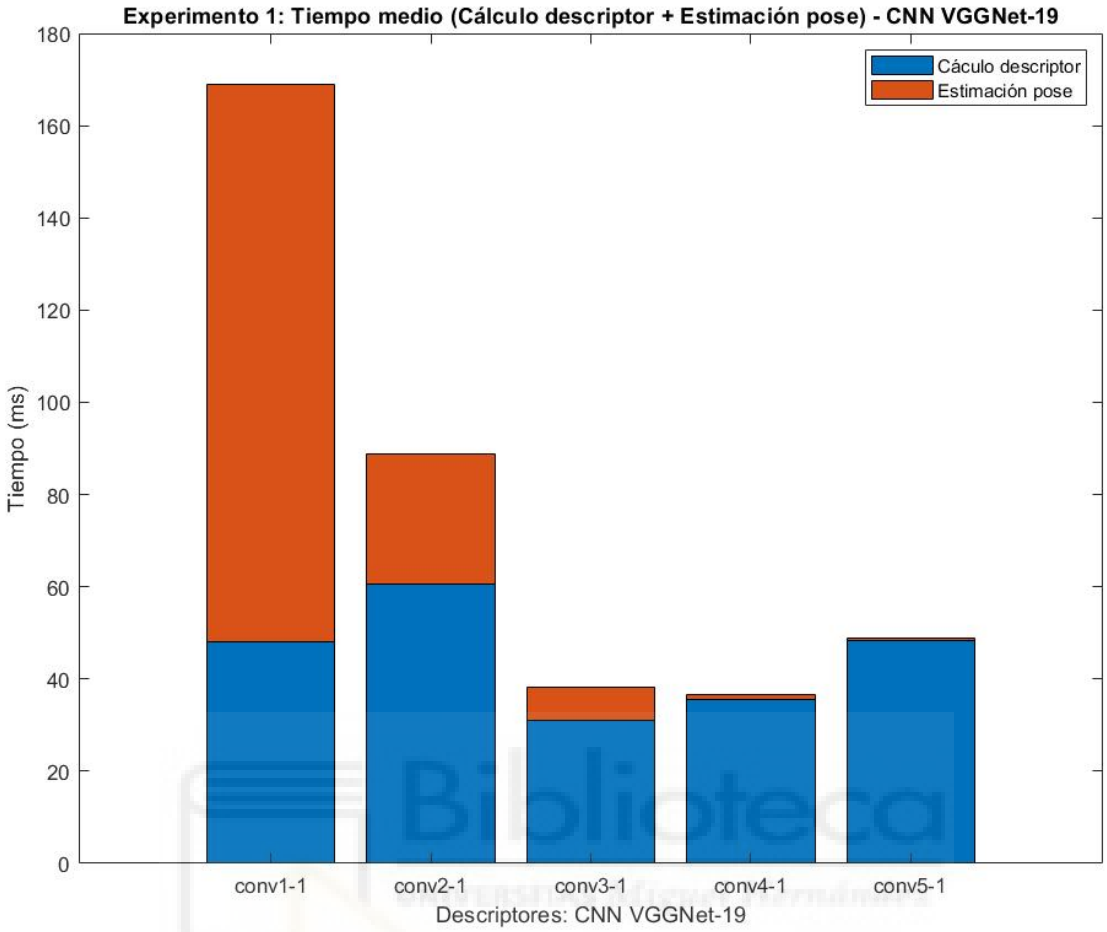


Figura 4.2: Tiempo medio total de cómputo de los descriptores obtenidos de las capas convolucionales de la CNN VGGNet-19.

Por un lado, en la Figura 4.1 se representa una gráfica que contiene el error medio de localización para cada una de las capas estudiadas y, a su vez, para cada uno de los efectos de iluminación estudiados. Por otro lado, en la Figura 4.2 se muestra una gráfica que contiene los datos correspondientes al tiempo. Así, para cada una de las capas se representa el tiempo total medio el cual está compuesto por el tiempo de cálculo del descriptor y el tiempo de cálculo de la estimación de la posición.

En este sentido, respecto al error medio de localización representado en la Figura 4.1, se observa que en el conjunto de imágenes de nublado los errores son similares en las cinco capas estudiadas, sin embargo, a la hora de trabajar con las imágenes de soleado y noche, podemos resaltar diferentes variaciones. Concretamente, en las capas ‘conv3_1’, ‘conv4_1’ y ‘conv5_1’, los errores medios en el conjunto de imágenes de noche

disminuyen considerablemente, siendo similares en estas tres capas. Sin embargo, en relación con el error medio en las imágenes de soleado, la capa ‘conv4_1’ presenta un error medio notablemente menor respecto a las demás capas. Por otro lado, en cuanto al tiempo medio total de cálculo representado en la Figura 4.2, se observa que es mucho mayor en las dos primeras capas de la CNN, disminuyendo en las tres últimas capas estudiadas donde el tiempo medio total de cálculo es similar. El tiempo medio total de cálculo constituye la suma de dos tiempos medios. Por un lado, el tiempo medio de cálculo del descriptor representado en azul, el cual es similar en las cinco capas y, por otro lado, el tiempo medio de cálculo de la estimación de la posición representado en rojo, donde podemos observar que es notablemente mayor en las dos primeras capas, ‘conv1_1’ y ‘conv2_1’, ya que presentan un tamaño mayor de descriptor que las capas posteriores. No obstante, la suma de ambos tiempos medios resulta menor en la’capa4_1’ al igual que el error medio de localización. Por todo ello, la capa convolucional con mejores resultados de la CNN VGGNet-19 corresponde a la capa ‘conv4_1’.

4.2.2 Comparativa entre las capas de la CNN ResNet-50

La CNN ResNet-50 es una red convolucional que destaca por la gran cantidad de capas que presenta y por implementar en ellas la conexión residual ya explicada previamente. Asimismo, estas conexiones residuales permiten realizar saltos entre capas intermedias, de manera que podemos destacar cinco etapas en su estructura. En este sentido, se realiza un estudio de la primera capa convolucional de cada una de estas cinco etapas, por lo que tendremos que realizar un previo estudio para determinar el mejor canal de cada capa convolucional, siguiendo el mismo procedimiento implementado para la CNN VGGNet-19. En la Tabla 4.5 se representa el canal que ofrece un menor error de localización para cada una de las capas convolucionales estudiadas.

Nombre de la capa	Mejor canal	Tamaño del descriptor
conv1	17	12544
res2a_branch2a	8	3136
res3a_branch2a	74	784
res4a_branch2a	20	196
res5a_branch2a	113	49

Tabla 4.5: Mejor canal obtenido para cada una de las capas convolucionales estudiadas en la CNN VGGNet-19.

A continuación, mediante las capas seleccionadas y su mejor canal se realiza la tarea de localización para cada una de las bases de datos con distintos efectos de iluminación, obteniendo los tres parámetros que nos permiten comparar la robustez y eficacia de cada una de las capas para su análisis exhaustivo.

La siguiente Figura 4.3 muestra la gráfica correspondiente al error medio de localización. De esta manera, podemos observar que las capas 'res3a_branch2a' y 'res4a_branch2a' presentan un error medio de posicionamiento bastante similar en todas las imágenes de nublado, noche o soleado, siendo estas dos capas las que mejores resultados proporcionan en comparación con las demás capas estudiadas. Adicionalmente, la gráfica de la Figura 4.4 representa los dos parámetros correspondientes al tiempo de cálculo. El primer tiempo medio de cálculo del descriptor es menor en las primeras dos capas estudiadas, 'res1a_branch2a' y 'res2a_branch2a'. No obstante, en el segundo tiempo medio de cálculo de la estimación de la posición ocurre lo contrario, conforme se avanza a través de las capas de la CNN, este tiempo disminuye debido a que el tamaño de los descriptores es considerablemente menor en las últimas tres capas, 'res1a_branch2a', 'res1a_branch2a' y 'res1a_branch2a'. La suma de estos dos tiempos da como resultado el tiempo medio total de cálculo, donde la capa 'res2a_branch2a' presenta el menor tiempo. Sin embargo, se debe tener en cuenta el error medio de localización, por lo que esta capa no corresponde con la mejor capa de la CNN. Así, entre las capas 'res3a_branch2a' y 'res4a_branch2a', cuyo error medio de posicionamiento es el menor entre todas las capas, la capa que mejores resultados ofrece en la CNNResNet-50 es la capa 'res3a_branch2a' ya que presenta un tiempo medio total de cálculo menor.

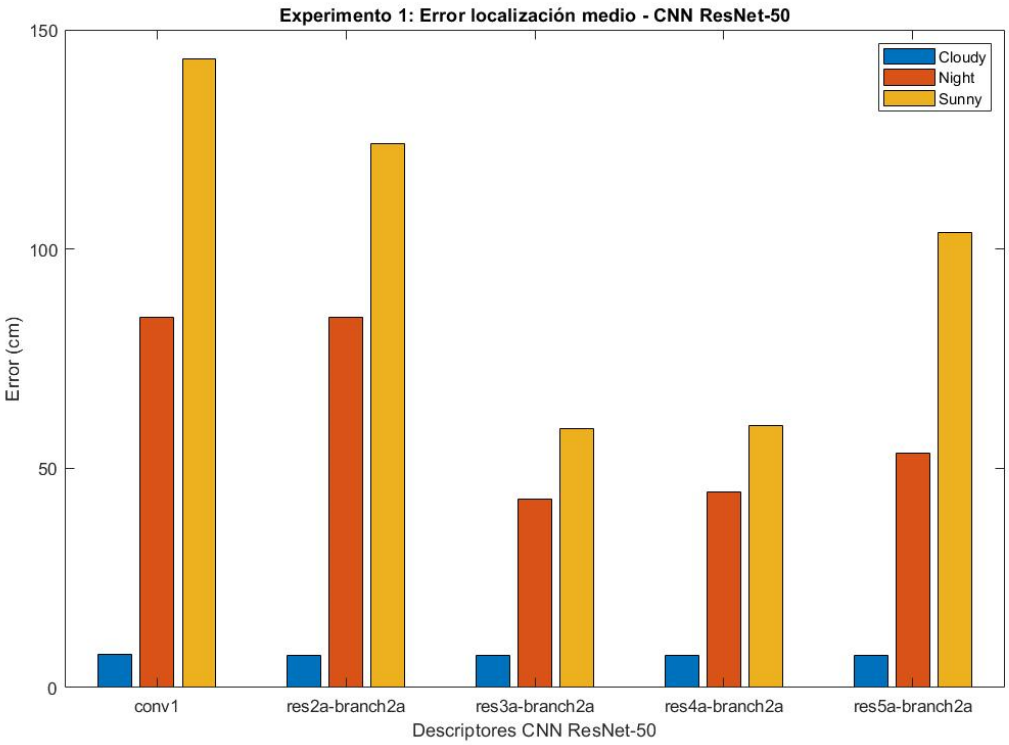


Figura 4.3: Error medio de localización de los descriptores obtenidos de las capas convolucionales de la CNN ResNet-50.

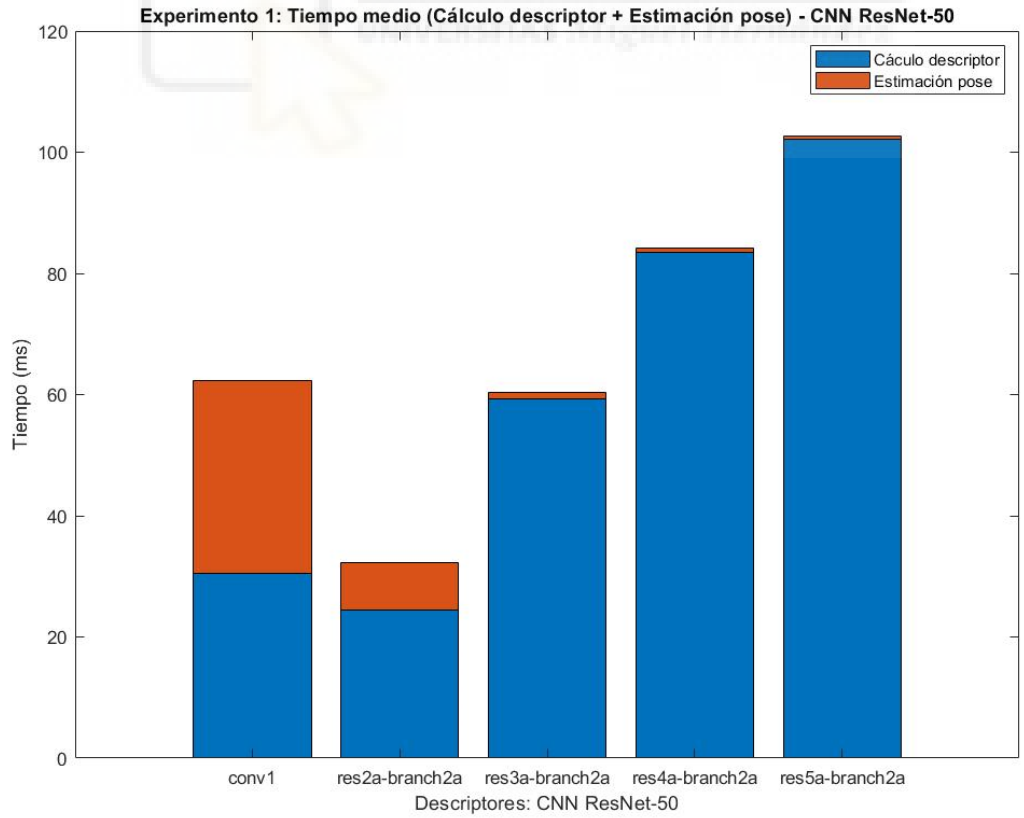


Figura 4.4: Tiempo medio total de cómputo de los descriptores obtenidos de las capas convolucionales de la CNN ResNet-50.

4.3 Experimento 2: Comparación entre CNNs

Durante el presente experimento, se realiza una comparativa entre las diferentes redes neuronales estudiadas, la CNN VGGNet-19 y la CNN ResNet-50. Previamente, hemos obtenido los datos asociados a cada una de las redes neuronales, de manera que hemos realizado una comparativa entre los diferentes descriptores holísticos obtenidos de cada una de las capas que componen cada red seleccionando la capa convolucional que mejores resultados ofrece. Tal y como se detalla en el experimento anterior, la capa que obtiene mejores resultados en la CNN VGGNet-19 es la 'conv4_1', mientras que en la CNN ResNet-50, es la capa 'res3a_branch2a', de manera que en este experimento se hace un análisis y estudio de la mejor capa seleccionada de cada una de las redes estudiadas.

A continuación, en la Figura 4.5 se muestra una gráfica donde se representa el parámetro correspondiente al error medio de localización para los descriptores obtenidos por la mejor capa seleccionada para cada red neuronal. En primer lugar, el error medio en el conjunto de imágenes de nublado y noche resultan muy similares tanto en la capa 'conv4_1' de la CNN VGGNet-19 como en la capa 'res3a_branch2a' de la CNN ResNet-50. La diferencia entre estas capas viene dada por el error medio en el conjunto de imágenes de soleado, siendo menor en la capa 'conv4_1' de la CNN VGGNet-19. En cuanto al tiempo medio de cálculo, sigue siendo la capa 'conv4_1' de la CNN VGGNet-19 la que presenta un tiempo menor. Concretamente, el tiempo medio de cálculo de la estimación de la posición es el que marca dicha diferencia ya que el tiempo de cálculo del descriptor también sigue siendo similar en ambas capas. En definitiva, la capa 'conv4_1' de la CNN VGGNet-19 es la que ofrece mejores resultados en el descriptor holístico obtenido, tanto en error medio como en tiempo medio de cálculo.

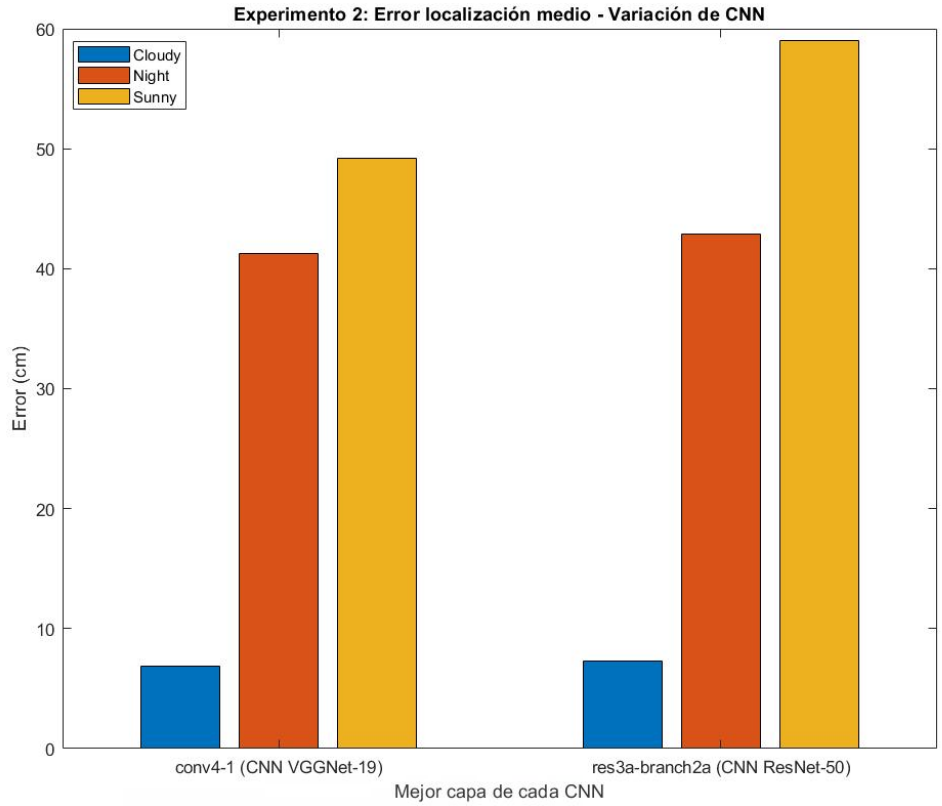


Figura 4.5: Error medio de localización de los descriptores seleccionados obtenidos de las capas convolucionales de la CNN VGGNet-19 y la CNN ResNet-50.

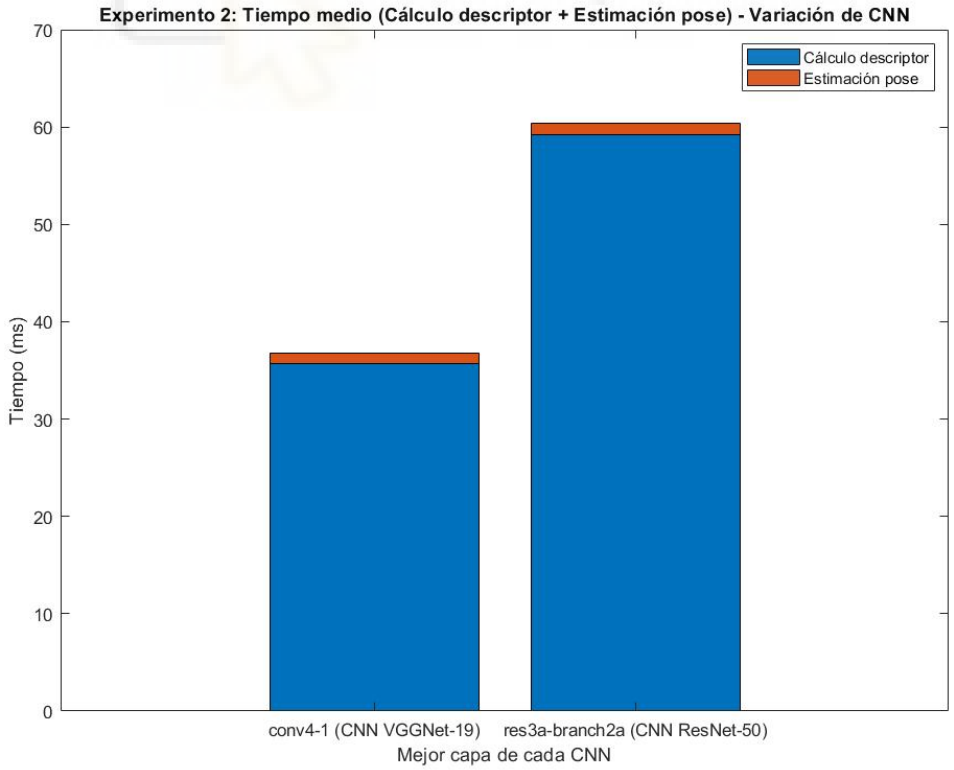


Figura 4.6: Tiempo medio total de cómputo de los descriptores seleccionados obtenidos de las capas convolucionales de la CNN VGGNet-19 y la CNN ResNet-50.

4.4 Experimento 3: Comparación entre las CNNs del presente estudio y las CNNs del estudio previo

Tal y como se ha explicado en capítulos anteriores, el presente trabajo de fin de grado sigue la línea de investigación de las tareas de *mapping* y localización de un trabajo previamente realizado [12]. Es por ello que, durante el presente experimento se va a realizar una comparativa entre los descriptores holísticos obtenidos en el presente trabajo y el trabajo previo [12] los cuales son obtenidos de las redes neuronales CNN Places, CNN AlexNet y CNN GoogLeNet. Durante dicho trabajo previo, se concluye que las capas que ofrecen mejores resultados corresponden a la 'conv2' para la CNN Places, la 'conv4' para la CNN AlexNet y la 'inception 3b_1x1' para la CNN GoogLeNet.

Aclaradas las capas que ofrecen descriptores holísticos con mejores resultados tanto para el previo trabajo como para el presente trabajo los cuales se describen en el experimento 2, procedemos a realizar un análisis de la comparación entre las cinco redes neuronales.

En primer lugar, en la Figura 4.7 se muestra la gráfica del error medio de localización obtenido en el presente trabajo junto con el obtenido en el estudio realizado previamente. Así, tal y como se observa en dicha gráfica, la capa 'conv2' de la CNN Places es la que ofrece mejores resultados respecto a las otras dos capas de las dos redes neuronales restantes que fueron estudiadas en el trabajo previo. Precisamente, el error medio en el conjunto de imágenes de nublado y noche resulta muy similar en las tres capas, por lo que la elección de la capa 'conv2' de la CNN Places viene dada por el error en el conjunto de imágenes de soleado, siendo este error mucho menor respecto a las capas de la CNN AlexNet y la CNN GoogLeNet. En cuanto a la comparación con las redes del presente estudio, el error medio de localización de la capa 'conv4_1' de la CNN VGGNet-19 es muy similar a la capa 'conv2' de la CNN Places. Sin embargo, es posible diferenciar que el error medio en el conjunto de imágenes de noche para la capa 'conv4_1' se incrementa en unos 10 cm aproximadamente respecto a la 'conv2' de la CNN Places.

Por otro lado, en la Figura 4.8 se observa la gráfica donde se representa el tiempo medio total de cálculo para las redes neuronales estudiadas tanto en el presente estudio como en el estudio previo. En este sentido, sigue siendo la capa ‘conv2’ de la CNN Places la que ofrece unos mejores resultados respecto a las dos otras redes neuronales que ya fueron estudiadas. No obstante, en la comparación con las redes del presente estudio, podemos observar que tanto la capa ‘conv4_1’ de la CNN VGGNet-19 como la capa ‘res3a_branch2a’ de la CNN ResNet-50 ofrecen un tiempo medio mucho menor. En concreto, el tiempo medio total de cálculo de la capa ‘conv2’ de la CNN Places duplica el tiempo medio total de cálculo de la ‘conv4_1’ de la CNN VGGNet-19. En consecuencia, a pesar de que la capa ‘conv2’ de la CNN Places presenta un error medio de localización menor, el tiempo medio total de cálculo que presenta es muchísimo mayor que en la capa ‘conv4_1’ de la CNN VGGNet-19, por lo que al ser la diferencia de errores sumamente mínima y la diferencia de tiempos más notable, se concluye que la capa que ofrece mejores resultados para obtener el descriptor holístico es la capa ‘conv4_1’ de VGGNet-19.

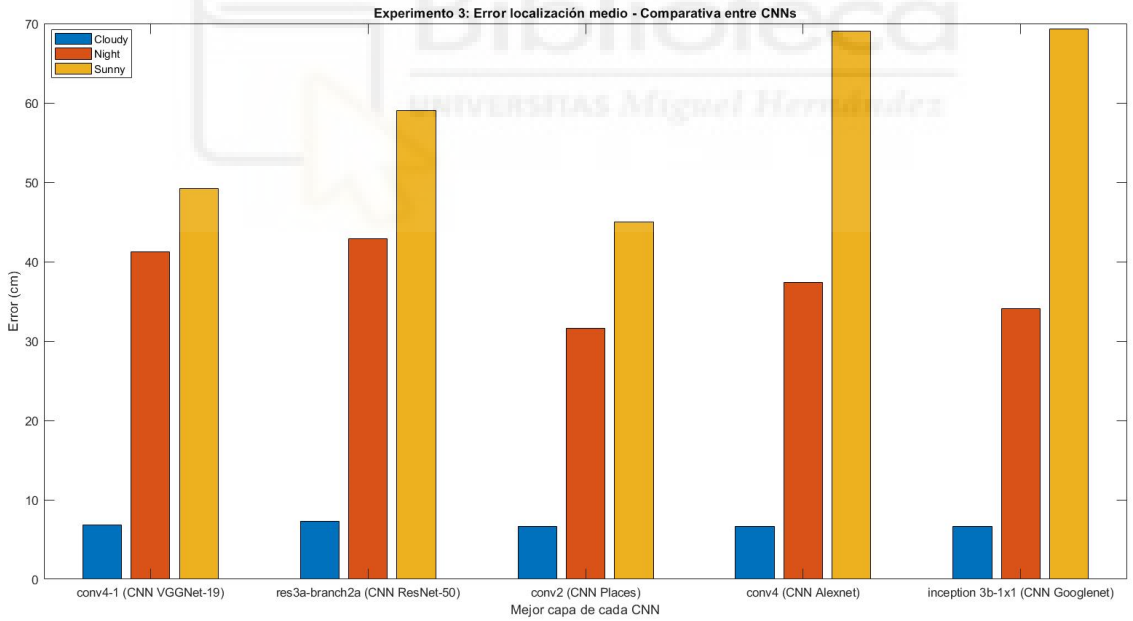


Figura 4.7: Error medio de localización de los descriptores seleccionados obtenidos de las capas convolucionales de la CNN VGGNet-19, ResNet-50, Places, AlexNet y GoogLeNet.

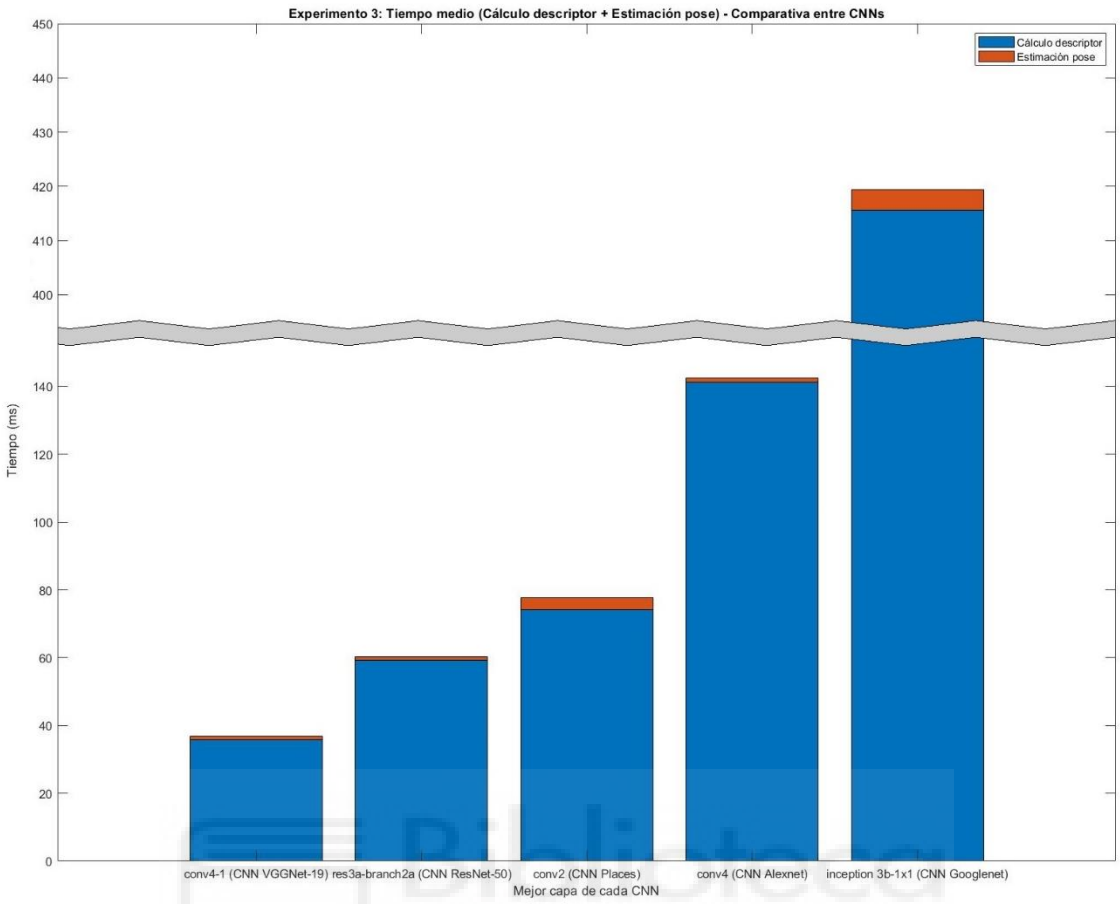


Figura 4.8: Tiempo medio total de cómputo de los descriptores seleccionados obtenidos de las capas convolucionales de la CNN VGGNet-19, ResNet-50, Places, AlexNet y GoogLeNet.

4.5 Experimento 4: Localización visual frente a efectos visuales

En los experimentos previamente realizados, se ha realizado la tarea de localización aplicando diferentes efectos de iluminación en la base de datos de las imágenes de nublado, consiguiendo así dos conjuntos de imágenes de soleado y de noche. En lo que respecta al experimento 4, se aplican nuevos efectos visuales al modelo visual de nublado el cual representa la base de datos original. Concretamente, se aplican tres efectos visuales diferentes que pueden dar lugar a situaciones reales de funcionamiento en el sensor visual del robot móvil:

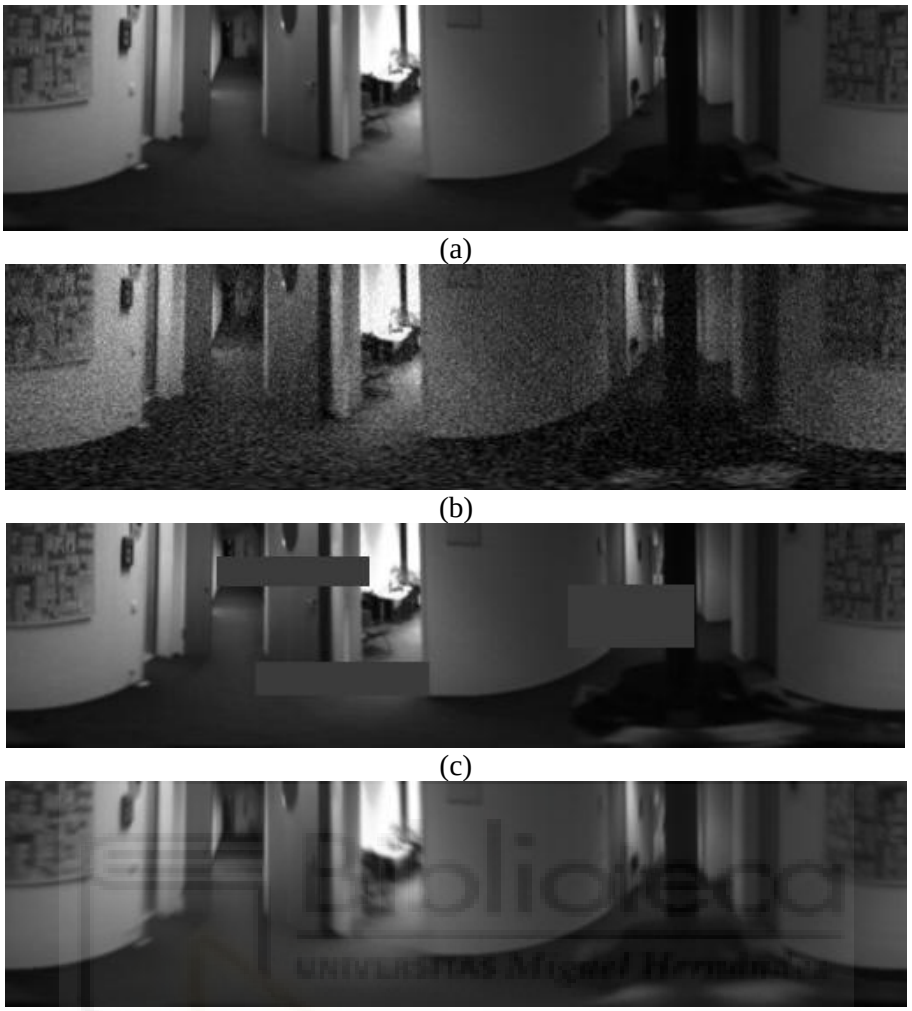
- La aplicación del primer efecto visual a las imágenes de nublado corresponde al efecto del ruido, donde se usa un ruido de tipo Gaussiano.

- En segundo lugar, se hará uso de unas oclusiones las cuales se refieren a unos rectángulos grises que se introducen en las imágenes con el fin de cubrir ciertas zonas de la imagen de test. Dichas oclusiones hacen referencia a los distintos obstáculos que puede encontrarse un robot móvil en su trayectoria, capturando imágenes incompletas.

- Por último, se aplica el efecto *blur* en las imágenes de test, cuyo efecto produce un desenfoque en la imagen que puede venir dado por un movimiento del sensor visual en el momento en el que captura la imagen. Es importante recalcar que cada uno de los efectos visuales se aplica individualmente a las imágenes de test de nublado, las cuales son las originales que han permitido obtener el *dataset* de entrenamiento, ya que en el presente estudio, el análisis de los resultados se hace en base a experimentos donde solo se aplica un efecto visual de forma individual.

En la Figura 4.9 se representa un conjunto de imágenes donde la primera imagen constituye la imagen original capturada en el edificio de Saarbrücken sin ningún efecto visual aplicado. La segunda imagen tiene aplicada un efecto de ruido de tipo Gaussiano, la tercera imagen incorpora diferentes oclusiones que cubren ciertas zonas de la imagen y, por último, la cuarta imagen tiene aplicada un efecto *blur* que produce en la imagen un desenfoque.

Una vez aclarados los efectos a utilizar para los diferentes conjuntos de imágenes que se introducen en las redes neuronales, se obtiene el parámetro correspondiente al error medio de localización para cada uno de los descriptores holísticos obtenidos a partir de las capas seleccionadas de la CNN VGGNet-19 y la CNN ResNet-50. Durante este experimento, no se hará uso de los parámetros referidos al tiempo ya que estos han sido estudiados ya en previos experimentos y, por lo tanto, no iban a aportar nuevos resultados en dicho estudio.



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

Figura 4.9: Imagen original capturada dentro del entorno de Saarbrücken (a) junto con un conjunto de imágenes donde se han aplicado diversos efectos visuales: (b) ruido Gaussiano, (c) oclusiones y (d) efecto *blur*.

Finalmente, en la Figura 4.10 se representa la gráfica con el error medio de localización para los descriptores seleccionados de las capas que obtienen mejores resultados tanto para la CNN VGGNet-19 como para la CNN ResNet-50. En este sentido, la capa 'conv4_1' de la CNN VGGNet-19 es la que mejores resultados ofrece. Tal y como podemos observar, su error medio de localización es muy similar al de la capa 'res3a_branch2a' de la CNN ResNet-50 en los conjuntos de imágenes afectados por las oclusiones y por el efecto *blur*. No obstante, en cuanto al error correspondiente a las imágenes que se le han aplicado un ruido de tipo Gaussiano se puede observar un gran incremento en la capa 'res3a_branch2a', dato que nos hace escoger como mejor descriptor holístico el obtenido por la capa 'conv4_1' de la CNN VGGNet-19.

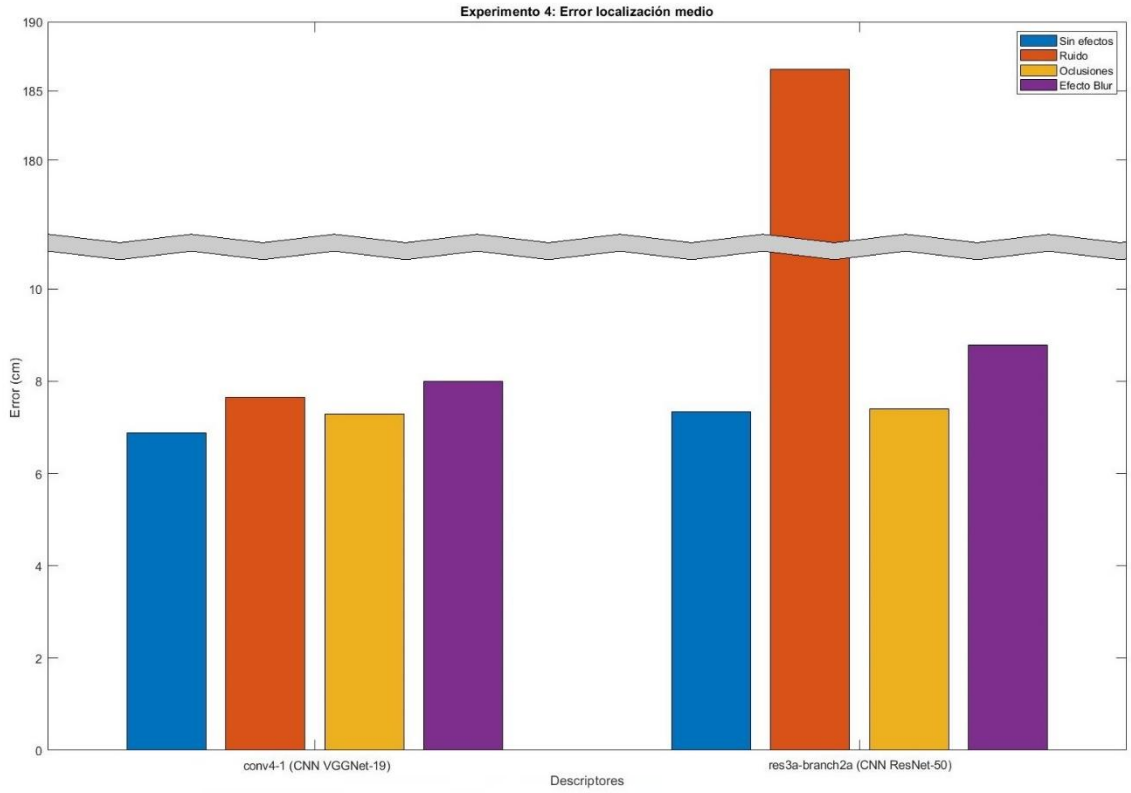
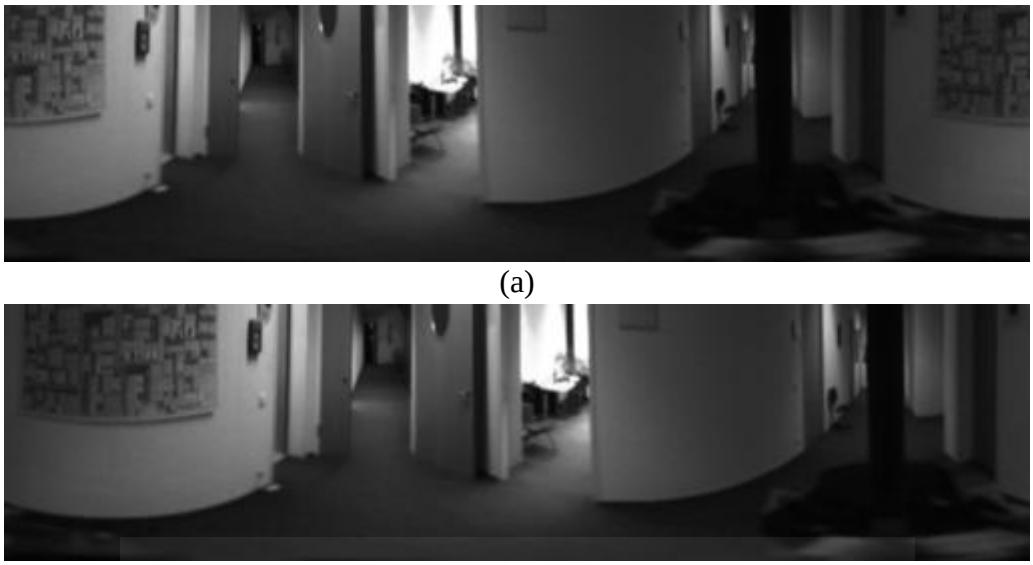


Figura 4.10: Error medio de localización de los descriptores seleccionados obtenidos de las capas convolucionales de la CNN VGGNet-19 y de la CNN ResNet-50 frente a los efectos visuales de ruido, oclusiones y efecto *blur*.

4.6 Experimento 5: Localización visual frente a rotaciones

El último experimento consiste en la aplicación de un nuevo efecto visual en las imágenes de test con el fin de obtener nuevos resultados y parámetros que nos ofrezcan un análisis de los descriptores holísticos obtenidos ante diferentes efectos visuales adicionales a los ya estudiados. En este caso, se hace uso del efecto de rotaciones el cual puede darse en situaciones reales cuando el sensor visual captura una imagen estando el robot móvil en una rotación diferente a cuando se capturaron las imágenes de entrenamiento para el proceso de *mapping*. Durante el presente experimento, se utiliza una rotación aleatoria entre 10 y 350 grados sobre la imagen omnidireccional, que viene a ser un desplazamiento horizontal en la imagen panorámica.

Para entender mejor en qué consiste dicha rotación, en la Figura 4.11 se muestra, por un lado, la imagen panorámica original de test y, por otro lado, una segunda imagen que corresponde a la imagen original pero donde se le ha aplicado un efecto de rotación aleatorio por lo que se puede observar un desplazamiento horizontal en ella.



(a)

(b)

Figura 4.11: Imagen original (a) capturada dentro del entorno de Saarbrücken junto con una imagen (b) donde se ha aplicado el efecto visual correspondiente al efecto de rotación.

Una vez aclarado el efecto visual a aplicar en las imágenes de test, se realiza el mismo procedimiento que en experimentos anteriores, de manera que se obtiene el descriptor holístico para cada una de las capas seleccionadas que ofrecen mejores resultados tanto para la CNN VGGNet-19 como para la CNN ResNet-50. En este sentido, al obtener dichos descriptores podremos extraer el parámetro correspondiente al error medio de localización y realizar un análisis más exhaustivo del funcionamiento de estas CNNs ante efectos de rotación en las imágenes de test. Cabe destacar, que durante este experimento tampoco se hará uso de los parámetros referidos al tiempo, ya que tal y como se explica en el anterior experimento, estos parámetros ya fueron obtenidos previamente, por lo que realizar un análisis de nuevo de dichos parámetros no aportaría información nueva a la ya estudiada.

Con todo ello, en la Figura 4.12 se muestra la gráfica con el error medio de localización para cada uno de los descriptores holísticos obtenidos de las capas seleccionadas de la CNN VGGNet-19 y de la CNN ResNet-50. Así, tal y como se puede

observar en la siguiente gráfica, el error de localización obtenido para el conjunto de imágenes con el efecto de rotaciones resulta realmente deficiente, ya que el error alcanza un gran incremento, de manera que la tarea de localización para este tipo de imágenes se muestra ciertamente empeorado. A pesar de obtener malos resultados en ambas redes neuronales, la CNN ResNet-50 presenta un menor error respecto a la CNN VGGNet-19.

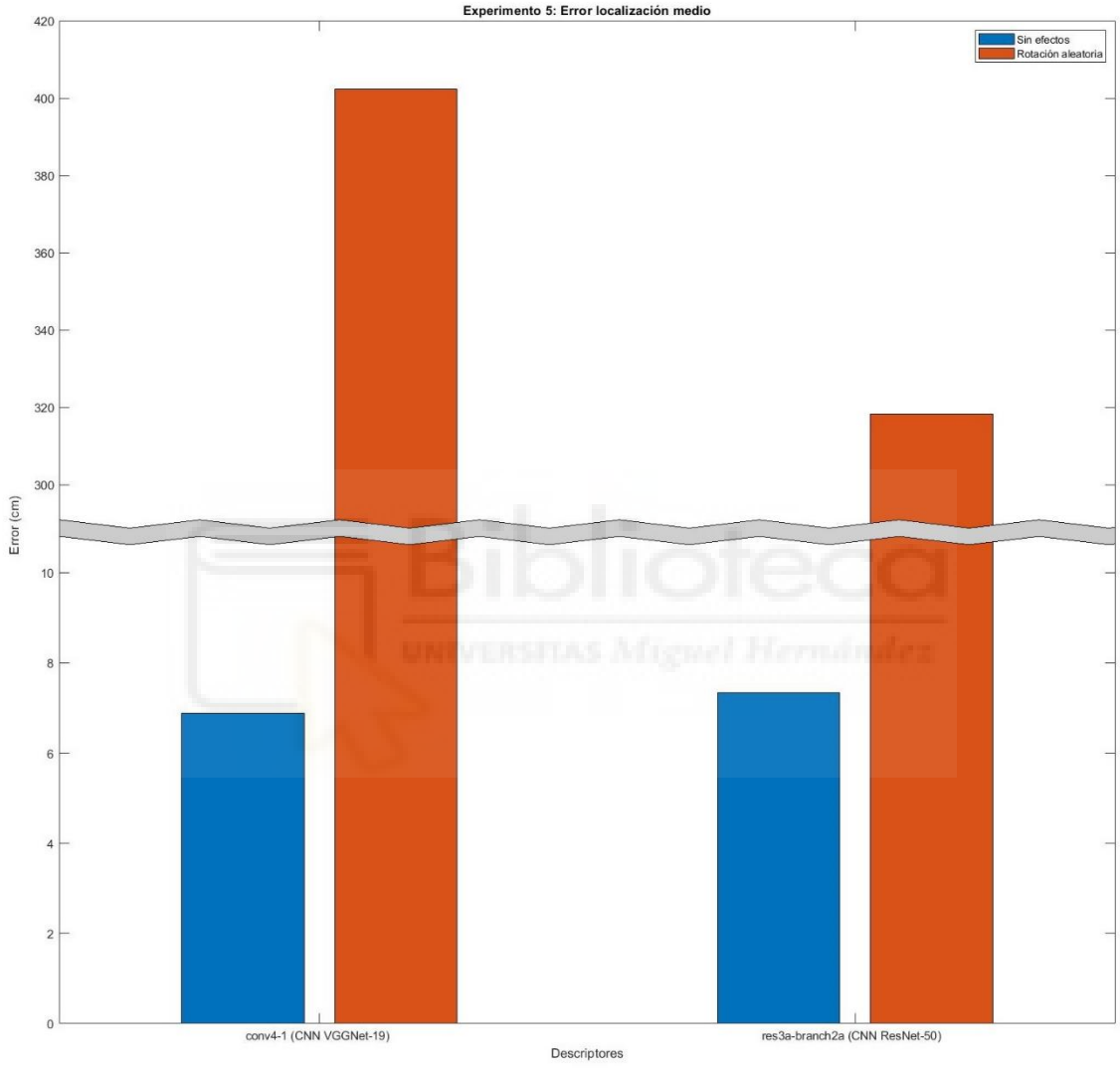


Figura 4.12: Error medio de localización de los descriptores seleccionados obtenidos de las capas convolucionales de la CNN VGGNet-19 y de la CNN ResNet-50 frente al efecto visual de rotaciones.

5 CONCLUSIONES Y TRABAJOS FUTUROS

A modo de cierre del presente trabajo, cabe señalar que el objetivo principal se basa en el análisis de redes neuronales convolucionales en la tarea de *mapping* y localización en la robótica móvil cuando se encuentran diferentes efectos de iluminación en el entorno. En este sentido, este estudio continúa con la línea de investigación realizada en el trabajo realizado previamente [12] donde se estudiaban las redes neuronales CNN Places, CNN AlexNet y CNN GoogLeNet, de manera que el objeto de estudio en este trabajo de fin de grado corresponde a las redes neuronales CNN VGGNet-19 y CNN ResNet-50.

Para llevar a cabo dicha investigación, se hace uso de un conjunto de imágenes ofrecido por la base de datos de COLD, siendo el conjunto de imágenes del Centro Alemán de Investigación en Inteligencia de Saarbrücken el escogido para los experimentos realizados. Concretamente, estas imágenes son capturadas por un equipo sensorial integrado en un robot móvil, de forma que las imágenes son ofrecidas en formato omnidireccional. Estas imágenes han sido tratadas con el fin de obtener diversos conjuntos de imágenes con efectos de iluminación diferentes. A continuación, se ha extraído la información visual de dichas imágenes mediante descriptores basados en apariencia global obtenidos a partir de métodos de *deep learning*, concretamente de redes neuronales convolucionales realizando así la tarea de *mapping*. La tarea de localización se desempeña una vez se obtienen dichos descriptores holísticos y, durante dicho proceso se han obtenidos los parámetros de error y de tiempo los cuales nos permiten realizar una comparación entre los diferentes descriptores según la red neuronal utilizada.

Con todo ello, se han llevado a cabo cinco experimentos diferentes que nos permiten llegar a ciertas conclusiones las cuales se detallan a continuación.

Experimento nº1

Durante este experimento se han estudiado las dos redes neuronales por separado, obteniendo gráficas de error de localización medio y de tiempo total de cálculo medio para cada una de las capas seleccionadas de la CNN VGGNet-19 y de la CNN ResNet-50.

Respecto a la CNN VGNet-19, se concluye que el descriptor holístico que ofrece mejores resultados es el obtenido a partir de la capa ‘conv4_1’. Concretamente, las capas ‘conv3_1’, ‘conv4_1’ y ‘conv5_1’ presentan un error medio de localización similar tanto en el conjunto de imágenes de nublado como en el conjunto de imágenes de noche, no obstante, el error medio es mucho menor en la capa ‘conv4_1’. Asimismo, el tiempo total de cálculo medio es también menor en la capa ‘conv4_1’.

Por otro lado, el menor error de localización medio representado para la CNN ResNet-50 corresponde a las capas ‘res3a_branch2a’ y ‘res4a_branch2a’, siendo este error muy similar en ambas capas. Sin embargo, a la hora de analizar el tiempo total de cálculo medio, la capa ‘res3a_branch2a’ es la que obtiene descriptores holísticos con mejores resultados. Por lo que la conclusión es que la mejor capa en la red CNN ResNet-50 corresponde a la ‘res3a_branch2a’.

Experimento nº2

En lo que refiere a este experimento, se ha procedido a realizar una comparación entre las mejores capas seleccionadas de cada una de las redes estudiadas. Con ello, podemos observar que los errores de localización medios son muy similares para ambas redes en el conjunto de imágenes de nublado y de noche. Ahora bien, el error se incrementa especialmente en la CNN ResNet-50 cuando se hace referencia al conjunto de imágenes de soleado. Adicionalmente, si se realiza un análisis de la gráfica correspondiente al tiempo total de cálculo medio, observamos que el mejor resultado es el ofrecido por VGGNet-19. Con todo ello, se selecciona la capa ‘conv4_1’ de la CNN VGGNet-19 como la capa que obtiene descriptores holísticos con mejores resultados entre todas las capas seleccionadas y estudiadas tanto de la CNN VGGNet-19 como de la CNN ResNet-50.

Experimento nº3

Tal y como se ha comentado anteriormente, el presente trabajo de fin de grado continúa con la línea de investigación de un trabajo ya realizado [12] para la CNN Places, la CNN AlexNet y la CNN GoogLeNet. Por ello, durante este experimento se ha realizado

una comparativa entre los resultados obtenidos en el presente trabajo y las redes neuronales estudiadas previamente. En la comparación del parámetro referido al error medio de localización, observamos que los errores más bajos corresponden a la CNN Places y a la CNN VGGNet-19 siendo en ambos muy similares. Por otro lado, en lo que respecta al parámetro del tiempo, la capa 'conv4_1' de la CNNVGGNet-19 presenta un tiempo total de cálculo de casi la mitad del tiempo que ofrece la capa 'conv2' de la CNN Places. Es por ello que, tras un análisis de los diferentes parámetros de error y de tiempo, se concluye que la capa que ofrece mejores resultados es la capa 'conv4_1' de la CNN VGGNet-19.

Experimento nº4

El cuarto experimento analiza los resultados obtenidos en la tarea de *mapping* y localización a la hora de trabajar con imágenes afectadas por distintos efectos visuales de iluminación. Concretamente, se han estudiado los efectos referidos al ruido de tipo Gaussiano, a las oclusiones y al efecto *blur* de desenfoque. De esta forma, analizando la gráfica del error de localización medio, se concluye que la capa 'conv4' de la CNN VGGNet-19 es la que ofrece mejores resultados ya que no produce ningún incremento desmesurado en el error para ninguno de los efectos estudiados.

Experimento nº5

Finalmente, el último experimento estudia el comportamiento de los descriptores holísticos ante imágenes que han sufrido ciertas rotaciones, simulando una situación real en la que el robot móvil captura la imagen de test con una rotación diferente a la empleada en la tarea de *mapping*. En conclusión, se observa que el error de localización medio se incrementa considerablemente para ambas redes neuronales. No obstante, a la hora de escoger una de las dos CNNs, la capa 'res3a_branch2a' de la CNN ResNet-50 presenta un error mucho menor, a pesar de que sigue siendo elevado.

En definitiva, la conclusión final es que los descriptores basados en apariencia global que utilizan herramientas de *deep learning* constituyen un campo de investigación considerablemente amplio que pueden llegar a darnos resultados muy eficientes en las

tareas de *mapping* y localización aplicadas en la robótica móvil. Los resultados obtenidos durante el presente trabajo nos llevan por lo tanto a considerar varias líneas de trabajo futuras que se pueden llevar a cabo con el fin de seguir con la investigación en dicho ámbito tal y como se expone en el trabajo previo [12]. A continuación, se detallan cuáles son:

- Realizar los experimentos especificados en el presente estudio en entornos de exterior con el objetivo de obtener un análisis de cómo afectan los efectos lumínicos en zonas al aire libre.
- Desarrollar esta investigación en una base de datos alternativa para contrastar los resultados.
- Aplicar el mismo procedimiento adquiriendo distintos descriptores holísticos basados en métodos de *deep learning*. En este caso, el presente trabajo ha seguido con esta línea de investigación, de manera que se han implementado diferentes redes neuronales a las previamente estudiadas.
- Desarrollar un sistema de navegación autónomo de un robot móvil a través del entorno a través de algoritmos de control visual. En esta línea de investigación, destaca la tarea de SLAM, ya que el robot móvil deberá realizar tanto la tarea de *mapping* como la tarea de localización simultáneamente.

REFERENCIAS

[1] E. Zamora, A. A. Pérez, J. G. Close, M. R. Costa-jussà, J. Martínez-Miranda, H. Pérez-Espinosa and W. A. Luna-Ramírez, "Robots Autónomos: Navegación", *Komputer Sapiens, Sociedad Mexicana de Inteligencia Artificial*, 2015.

[2] G. Grisetti, C. Stachniss and W. Burgard, (2007). "Improved techniques for grid mapping with rao-blackwellized particle filters". *IEEE transactions on Robotics*, 23(1), 34-46.

[3] D. Hahnel, W. Burgard, D. Fox and S. Thrun, (2003, October). An efficient FastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No. 03CH37453)* (Vol. 1, pp. 206-211). IEEE.

[4] P. Biber, H. Andreasson, T. Duckett and A. Schilling, (2004, September). 3D modeling of indoor environments by a mobile robot with a laser scanner and panoramic camera. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No. 04CH37566)* (Vol. 4, pp. 3430-3435). IEEE.

[5] R. M. Eustice, H. Singh and J. J. Leonard, (2005, April). Exactly sparse delayed-state filters. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation* (pp. 2417-2424). IEEE.

[6] R. Triebel and W. Burgard, (2005, July). Improving simultaneous mapping and localization in 3d using global constraints. In *aaai* (pp. 1330-1335).

[7] Ó. Boullosa García, (2011). *Estudio comparativo de descriptores visuales para la detección de escenas cuasi-duplicadas* (Bachelor's thesis).

[8] J. Huang, S. Ravi Kumar, M. Mitra, W. J. Zhu and R. Zabih, (1999). Spatial color indexing and applications. *International Journal of Computer Vision, 35*(3), 245-268.

[9] K. Mikolajczyk and C. Schmid, (2004). Scale & affine invariant interest point detectors. *International journal of computer vision*, *60*(1), 63-86.

[10] L. Payá, A. Gil and Ó. Reinoso, (2017). A state-of-the-art review on mapping and localization of mobile robots using omnidirectional vision sensors. *Journal of Sensors*, *2017*.

[11] T. Kramer, (2002). *Assessment of the Commercial Applicability of Artificial Intelligence in Electronic Businesses*. diplom. de.

[12] O. J. Céspedes, "Localización de un robot móvil utilizando información visual y redes neuronales convolucionales", Trabajo de Fin de Grado, Ingeniería Electrónica y Automática Industrial, Universidad Miguel Hernández de Elche, Elche, Alicante, España, 2020.

[13] D. Scaramuzza and K. Ikeuchi, (2014). Omnidirectional camera.

[14] J. J. Kumler and M. L. Bauer, (2000, October). Fish-eye lens designs and their relative performance. In *Current developments in lens design and optical systems engineering* (Vol. 4093, pp. 360-369). SPIE.

[15] S. Baker and S. K. Nayar, (1999). A theory of single-viewpoint catadioptric image formation. *International journal of computer vision*, *35*(2), 175-196.

[16] T. as Svoboda and T. as Pajdla, (1997). Central panoramic cameras: Geometry and design.

[17] L. Payá, L. Fernández, Ó. Reinoso, A. Gil and D. Úbeda, "Appearance-based Dense Maps Creation - Comparison of Compression Techniques with Panoramic Images", in *Proceedings of the 6th International Conference on Informatics in Control, Automation and Robotics*, Milan, Italy, 2009, pp. 250-255.

[18] J. Gaspar, N. Winters and J. Santos-Victor, "Vision-based navigation and environmental representations with an omnidirectional camera", *IEEE Transactions on Robotics and Automation*, vol. 16, no. 6, pp. 890-898, December 2000.

[19] N. Dalal and B. Triggs, (2005, June). Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)* (Vol. 1, pp. 886-893). Ieee.

[20] A. Oliva and A. Torralba, "Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope", *International Journal of Computer Vision*, vol. 42, no. 1, pp. 145-175, May 2001.

[21] Pytorch, «Pytorch,» [Online]. Available: https://pytorch.org/docs/stable/generated/torch.nn.Conv2d.html.

[22] ICHI.PRO, «ICHI.PRO,» [Online]. Available: https://ichi.pro/es/matematicas-de-las-redes-neuronales-convolucionales-30697178023872.

[23] MM. M. Ullah, A. Pronobis, B. Caputo, J. Luo and P. Jensfelt, "The COLD Database", KTH Royal Institute of Technology, Stockholm, Sweden, Technical Report TRITA-CSC-CV 2007:1, 2007.

[24] K. Simonyan and A. Zisserman, (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[25] K. He, X. Zhang, S. Ren and J. Sun, (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[26] S. Cebollada, L. Payá, W. Mayol and Ó. Reinoso, "Evaluation of Clustering Methods in Compression of Topological Models and Visual Place Recognition Using Global Appearance Descriptors", *Applied Sciences*, vol. 9, no. 3, pp. 1-30, January 2019.