

**Universidad Miguel Hernández de Elche**

**MASTER UNIVERSITARIO EN**

**ROBÓTICA**



“Localización de un robot móvil mediante visión por computador ante grandes cambios de iluminación”

Trabajo de Fin de Máster

Curso académico 2016/17

Autor: Vicente Román Erades

Tutor: Luis Payá Castelló



# Lista de Figuras

2-1.	Sistema de captación de imágenes Ladybug. . . . .	9
2-2.	Sistema de captación de imágenes Gear 360. . . . .	10
2-3.	Modelo de proyección de sistemas catadióptricos con un único punto de proyección. a: Espejo hiperbólico y lente convencional. b: Espejo parabólico y lente ortográfica. . . . .	11
2-4.	Ejemplo de dos sistemas de visión omnidireccional. . . . .	12
2-5.	Proyección del modelo esfera unidad. . . . .	13
2-6.	Proyección del modelo cilíndrico. . . . .	13
2-7.	Proyección del modelo en perspectiva. . . . .	14
2-8.	Proyección del modelo ortográfico. . . . .	15
2-9.	(a)Imagen omnidireccional y transformaciones calculadas sobre ella: (b)proyección esfera unidad, (c)proyección cilíndrica, (d)proyección en perspectiva en un plano vertical, y (e)proyección ortográfica. . . . .	15
3-1.	Transformada Radon de una imagen omnidireccional.(a)imagen omnidireccional, (b) transformada Radon de la imagen y (c) transformada Radon final con tamaño $360 \times 0,5 * N_x$ . . . . .	24
3-2.	Propiedad de desplazamiento en la transformada Radon. . . . .	25
4-1.	Imágenes adquiridas en la universidad de Friburgo . . . . .	28
4-2.	Recorrido del robot móvil por el laboratorio de Friburgo. . . . .	29
4-3.	Sistema ActivMedia Pioneer-3 . . . . .	30
5-1.	FS. Error de localización con el <b>día nublado</b> al variar el número de columnas $k_1$ . . . . .	33
5-2.	HOG. Error de localización con el <b>día nublado</b> al variar el número de celdas horizontales $k_3$ y el número de bins por histograma $b_3$ . . . . .	33
5-3.	gist. Error de localización con el <b>día nublado</b> al variar el número de bloques horizontales $k_4$ y el número de filtros de Gabor $m_1$ . . . . .	34
5-4.	RT Método 1. Error de localización con el <b>día nublado</b> al variar el número de grados entre ángulos consecutivos $p_1$ y el número de columnas $K_5$ . . . . .	34
5-5.	RT Método 2. Error de localización con el <b>día nublado</b> al variar el número de grados entre ángulos consecutivos $p_1$ . . . . .	35

5-6. HOG detallado. Error de localización con el <b>día nublado</b> al variar el número de celdas horizontales $k_3$ y el número de bins por histograma $b_3$ . . . . .	36
5-7. gist detallado. Error de localización con el <b>día nublado</b> al variar el número de bloques horizontales $k_4$ y el número de filtros de Gabor $m_1$ . . . . .	36
5-8. RT Método 1 detallado. Error de localización con el <b>día nublado</b> al variar el número de grados entre ánguos consecutivos $p_1$ y el número de columnas $K_5$ . . . . .	37
5-9. FS. Error de localización con el <b>día soleado</b> al variar el número de columnas $k_1$ . . . . .	38
5-10.HOG. Error de localización con el <b>día soleado</b> al variar el número de celdas horizontales $k_3$ y el número de bins por histograma $b_3$ . . . . .	38
5-11.gist. Error de localización con el <b>día soleado</b> al variar el número de bloques horizontales $k_4$ y el número de filtros de Gabor $m_1$ . . . . .	39
5-12.RT Método 1. Error de localización con el <b>día soleado</b> al variar el número de grados entre ánguos consecutivos $p_1$ y el número de columnas $K_5$ . . . . .	39
5-13.RT Método 2. Error de localización con el <b>día soleado</b> al variar el número de grados entre ánguos consecutivos $p_1$ . . . . .	40
5-14.HOG detallado. Error de localización con el <b>día soleado</b> al variar el número de celdas horizontales $k_3$ y el número de bins por histograma $b_3$ . . . . .	41
5-15.gist detallado. Error de localización con el <b>día soleado</b> al variar el número de bloques horizontales $k_4$ y el número de filtros de Gabor $m_1$ . . . . .	42
5-16.RT Método 1 detallado. Error de localización con el <b>día soleado</b> al variar el número de grados entre ánguos consecutivos $p_1$ y el número de columnas $K_5$ . . . . .	42
5-17.FS. Error de localización cuando ha <b>anochecido</b> al variar el número de columnas $k_1$ . . . . .	43
5-18.HOG. Error de localización cuando ha <b>anochecido</b> al variar el número de celdas horizontales $k_3$ y el número de bins por histograma $b_3$ . . . . .	44
5-19.gist. Error de localización cuando ha <b>anochecido</b> al variar el número de bloques horizontales $k_4$ y el número de filtros de Gabor $m_1$ . . . . .	44
5-20.RT Método 1. Error de localización cuando ha <b>anochecido</b> al variar el número de grados entre ánguos consecutivos $p_1$ y el número de columnas $K_5$ . . . . .	45
5-21.RT Método 2. Error de localización cuando ha <b>anochecido</b> al variar el número de grados entre ánguos consecutivos $p_1$ . . . . .	45
5-22.HOG detallado. Error de localización cuando ha <b>anochecido</b> al variar el número de celdas horizontales $k_3$ y el número de bins por histograma $b_3$ . . . . .	46
5-23.gist detallado. Error de localización cuando ha <b>anochecido</b> al variar el número de bloques horizontales $k_4$ y el número de filtros de Gabor $m_1$ . . . . .	47
5-24.RT Método 1 detallado. Error de localización cuando ha <b>anochecido</b> al variar el número de grados entre ánguos consecutivos $p_1$ y el número de columnas $K_5$ . . . . .	47
5-25.FS. Tiempo de localización por imagen al variar el número de columnas $k_1$ . . . . .	49

5-26.HOG. Tiempo de localización por imagen al variar el número de celdas horizontales $k_3$ y el número de bins por histograma $b_3$ . . . . .	49
5-27.gist. Tiempo de localización por imagen al variar el número de bloques horizontales $k_4$ y el número de filtros de Gabor $m_1$ . . . . .	50
5-28.RT Método 1. Tiempo de localización por imagen al variar el número de grados entre ángulos consecutivos $p_1$ y el número de columnas $K_5$ . . . . .	50
5-29.RT Método 2. Tiempo de localización por imagen al variar el número de grados entre ángulos consecutivos $p_1$ . . . . .	51
6-1. Mapa de Friburgo y los puntos elegidos para estudiar el cálculo del cambio en la orientación. . . . .	54
6-2. Ejemplo de un cruce en el que se ha calculado cambio de orientación . . . . .	55
6-3. Ejemplo de un cruce en el que se ha calculado cambio de orientación . . . . .	56
6-4. Ejemplo de un giro en el que se ha calculado cambio de orientación . . . . .	57
6-5. Ejemplo de un giro en el que se ha calculado cambio de orientación . . . . .	58
6-6. FS. Error de orientación con el <b>día nublado</b> al variar el número de columnas $k_2$ . . . . .	60
6-7. HOG. Error de orientación con el <b>día nublado</b> al variar la anchura de las celdas verticales $l_1$ y el número de bins por histograma $b_2$ . . . . .	60
6-8. gist. Error de orientación con el <b>día nublado</b> al variar la anchura de las celdas verticales $l_2$ y el número de filtros de Gabor $m_2$ . . . . .	61
6-9. RT Método 1. Error de orientación con el <b>día nublado</b> al variar el número de grados entre ángulos consecutivos $p_2$ y el número de columnas $K_6$ . . . . .	61
6-10.RT Método 2. Error de orientación con el <b>día nublado</b> al variar el número de grados entre ángulos consecutivos $p_2$ . . . . .	62
6-11.FS. Error de orientación con el <b>día soleado</b> al variar el número de columnas $k_2$ . . . . .	63
6-12.HOG. Error de orientación con el <b>día soleado</b> al variar la anchura de las celdas verticales $l_1$ y el número de bins por histograma $b_2$ . . . . .	64
6-13.gist. Error de orientación con el <b>día soleado</b> al variar la anchura de las celdas verticales $l_2$ y el número de filtros de Gabor $m_2$ . . . . .	64
6-14.RT Método 1. Error de orientación con el <b>día soleado</b> al variar el número de grados entre ángulos consecutivos $p_2$ y el número de columnas $K_6$ . . . . .	65
6-15.RT Método 2. Error de orientación con el <b>día soleado</b> al variar el número de grados entre ángulos consecutivos $p_2$ . . . . .	65
6-16.FS. Error de orientación cuando ha <b>anochecido</b> al variar el número de columnas $k_2$ . . . . .	67
6-17.HOG. Error de orientación cuando ha <b>anochecido</b> al variar la anchura de las celdas verticales $l_1$ y el número de bins por histograma $b_2$ . . . . .	67

6-18.gist. Error de orientación cuando ha <b>anohecido</b> al variar la anchura de las celdas verticales $l_2$ y el número de filtros de Gabor $m_2$ . . . . .	68
6-19.RT Método 1. Error de orientación cuando ha <b>anohecido</b> al variar el número de grados entre ángulos consecutivos $p_2$ y el número de columnas $K_6$ . . . . .	68
6-20.RT Método 2. Error de orientación cuando ha <b>anohecido</b> al variar el número de grados entre ángulos consecutivos $p_2$ . . . . .	69



# Lista de Tablas

3-1. Parámetros con influencia en los procesos de localización y orientación . . . .	26
4-1. Parámetros de las cámaras . . . . .	29
5-1. Medidas de distancia . . . . .	32



# 1. Introducción

A lo largo de los últimos años, la presencia de robots móviles tanto en entornos industriales como en nuestros hogares se ha incrementado notablemente experimentando un gran desarrollo gracias a la variedad de problemas que puede solucionar esta tecnología. El aumento del número de este tipo de robots se ha visto facilitado por el desarrollo de capacidades de percepción y computación que permite mejorar la habilidad de operar en zonas amplias y heterogéneas sin necesidad de introducir cambios en la estructura del robot. Sin embargo, el despegue no será definitivo hasta que no se incremente sustancialmente el nivel de autonomía y adaptabilidad a diferentes circunstancias.

Un robot autónomo debe ser capaz de trabajar en áreas desconocidas, planear trayectorias y navegar a puntos deseados. Más concretamente, un robot móvil tiene que encontrar una buena solución a dos problemas críticos: construir un modelo del entorno (mapping) y estimar la posición del robot en el modelo (localización), si realiza ambas tareas de manera simultánea estaremos hablando de que el robot realiza una tarea de SLAM. Además debe resolver estos problemas de mapping y localización con una precisión y coste computacional aceptable.

La utilización de imágenes omnidireccionales permite obtener información relevante para solucionar los problemas de mapping y localización. La forma de extraer información de estas imágenes ha seguido dos vertientes. La manera clásica es mediante la extracción de puntos característicos, una alternativa madura y bastante desarrollada. SIFT y SURF son algunos ejemplos (aunque hay muchos más tipos). Su uso se basa en la extracción, matching y tracking de puntos a lo largo de un conjunto de escenas, lo cual suele suponer un alto coste computacional. Además, ciertos métodos no son demasiado robustos ante cambios de perspectiva, cambios en el entorno, etc. Se requiere de un entorno mínimamente estructurado para que estos enfoques funcionen.

Frente a ellos, más recientemente se ha propuesto el uso de descriptores de apariencia global (Firma de Fourier, HOG, gist, PCA...). Estos descriptores se basan en la creación de un único descriptor por imagen, que contiene información de su apariencia global. El hecho de tener un único descriptor hace que los métodos de mapping y localización sean sencillos, basados muchas veces en la comparación de pares de imágenes. Asimismo, la aplicación a entornos estructurados es más directa y su coste computacional suele ser menor. Dado que



se trata de una técnica menos madura, es interesante estudiar con profundidad su comportamiento en tareas de localización y su robustez ante determinados cambios en el entorno. De ahí el interés de este trabajo.

Teniendo estos datos en cuenta, el objetivo principal de este trabajo es el diseño y análisis de un sistema de localización de un robot móvil en entornos sociales de interior cambiantes. El sensor usado ha sido un sistema de visión catadióptrico montado sobre el propio robot, que proporciona imágenes con un campo de visión de 360° alrededor del robot. En primer lugar, se ha creado un modelo visual del entorno a partir de las imágenes omnidireccionales capturadas por el robot. A continuación, se ha abordado el problema de localización. Para ello, se compara la información visual capturada por el robot desde una posición desconocida con la información almacenada en el modelo. El resultado, es la estimación y orientación del robot tras haber utilizado únicamente información visual.

Durante el proceso de localización se ha realizado un estudio de la robustez de los descriptores de apariencia global comprobando su comportamiento cuando la apariencia visual del entorno cambia sustancialmente ya que en entornos de trabajo reales, el robot debe ser capaz de afrontar con éxito diversos fenómenos:

- Las condiciones de iluminación en el instante en el que se va a realizar la localización pueden ser muy distintas a las condiciones que había cuando se creó el modelo (por tratarse de otro momento del día o del año, o por la presencia o no de iluminación artificial).
- Las escenas pueden encontrarse parcialmente ocluidas por la presencia de personas u otros robots móviles.
- El entorno puede haber sufrido otros cambios a largo plazo, como cambios en la posición del mobiliario.

En resumen, en este trabajo se realiza un evaluación comparativa de cuatro descriptores de carácter global para la tarea de localización, estudiando para cada uno de ellos la precisión y el coste computacional. La finalidad es conocer la robustez de cada uno de estos algoritmos de descripción ante los fenómenos habituales que pueden surgir en un entorno de trabajo real y sintonizar adecuadamente sus parámetros para llegar a un compromiso entre precisión y coste computacional. La base de datos COLD [1] con la que hemos trabajado, ofrece cambios de iluminación y cambios introducidos por la actividad humana (aparecen personas por las habitaciones, objetos y muebles se mueven o reorganizan...) esto ofrecerá un ambiente comparable a la realidad.

## 2. Mapping y localización de robots móviles utilizando sensores de visión omnidireccionales

Para abordar el problema de mapping, localización y navegación es necesario que el robot tenga información relevante de su movimiento y del entorno por el que desarrolla su tarea, ya que este suele ser desconocido en la mayoría de aplicaciones. Para realizar las tareas de forma autónoma, los robots están equipados de diversos equipos de sensores que les permiten recopilar información sobre su movimiento y el entorno por el que trabajan. Por ello, desde los primeros trabajos en robótica móvil [2], en mayor o menor medida se ha debido de tener cuidado e interés en el buen funcionamiento y mejora de los sistemas de captación de información.

Siguiendo esta dinámica han surgido diferentes métodos de captación de información considerando diferentes tipos de sensores, métodos o algoritmos. En los últimos años el uso de cámaras omnidireccionales se ha convertido en un recurso bastante popular en muchos trabajos de robótica móvil. Como veremos más adelante, este tipo de cámaras nos devuelven la misma información de la escena independientemente de la orientación del robot, de esta manera se puede estimar la posición independientemente de la orientación, pudiendo posteriormente estimar también la orientación (esto con las cámaras normales no es posible). Además, las cámaras omnidireccionales devuelven la información en modo de matriz la cual podemos comprimir en un único vector por imagen utilizando los descriptores globales. Debido a esta propiedad de compresión de información esta técnica resulta muy útil para tareas de mapping, localización y navegación.

Los métodos utilizados para captar y procesar información han tenido una evolución exponencial con la mejora de las capacidades de percepción y computación de los sistemas y la aparición de nuevos algoritmos y métodos. En su trabajo, Paya et al. [3] hacen una síntesis de diferentes equipos de sensores y métodos de captación presentando los trabajos más relevantes en la solución de mapping y localización.

Este capítulo se divide en dos secciones. En la primera, se expone los tipos de sensores que pueden ser utilizados para extraer información del entorno y del propio robot móvil. En la

sección 2.2 se presenta una serie de conceptos relacionados con la visión omnidireccional, las posibles geometrías y las transformadas que se pueden construir a raíz de los datos extraídos de la información visual.

## 2.1. Sensores en robótica móvil

Esta sección se divide en cuatro subsecciones. En la subsección 2.1.1 se analiza los sensores que tradicionalmente se han utilizado en robótica móvil para trabajar en los campos de mapping, localización y navegación. A continuación, se exponen los sistemas de visión; se estudia su ventajas e inconvenientes en la subsección 2.1.2, sus posibles configuraciones en la subsección 2.1.3 y se expone las últimas tendencias de creación de modelos visuales en la subsección 2.1.4

### 2.1.1. Tipos de sensores tradicionales

Como se ha podido comprobar, el problema de mapping, localización y navegación de robots móviles solo es posible de abordar con un fuerte conocimiento del estado del robot y gran información del exterior. Aportar esta información ha sido el principal propósito de los diferentes tipos de sensores a lo largo de los años.

Los sensores admiten una clasificación general en la que pueden ser separados en propioceptivos y exteroceptivos. Por un lado, los sensores propioceptivos ofrecen información sobre el robot, la odometría o el nivel de batería podrían ser ejemplos de sensores que se incluyen en este lado de la clasificación. Por otro lado se encuentran los sensores exteroceptivos, como el GPS, SONAR, el laser o sensores infrarrojos; este tipo de sensores presentan medidas con información de la distribución exterior al robot.

La odometría es la principal fuente de información dentro de los sensores propioceptivos. Habitualmente, la odometría se calcula con la instalación de encoders en las ruedas y de esta manera puede ser estimada la posición del robot gracias al conocimiento de la posición inicial del robot y el movimiento realizado. El problema es que esta información no puede utilizarse como absoluta debido a la acumulación de errores que puede haber: deslizamientos, desniveles o baches, diferente rugosidad del terreno...y por ello en aplicaciones reales es usual encontrar este tipo de sensores complementados con alguna medida exteroceptiva.

Por contraste encontramos los sensores exteroceptivos, estos tienen asociados otro tipo de problemas. El GPS ofrece una buena localización en lugares de exterior, sin embargo dentro de edificios o en calles muy estrechas no tiene fiabilidad. El SONAR es un instrumento de

medida que trabaja con ondas acústicas, este sensor tiene un precio relativamente bajo y determina la distancia de objetos y obstáculos que se encuentran próximos [4] midiendo el tiempo de vuelo de las ondas acústicas que genera, no obstante su precisión no es muy alta debido a la incertidumbre angular que se genera y al ruido creado por la reflexión de las señales acústicas [5]. El laser toma medidas mediante el tiempo de vuelo de un pulso laser cuando este es reflejado en el objeto próximo, su precisión y campo de trabajo es más alto que los SONAR ya que permiten medir con buena resolución y precisión objetos desde pocos centímetros a decenas de metros, sin embargo el coste y el consumo de energía del equipo es bastante alto. Los láseres permiten crear mapas 2D y 3D y habitualmente se utilizan combinados con la odometría [6].

### 2.1.2. Sensores de visión

En contraposición a los sensores vistos en el apartado anterior encontramos los sistemas de percepción por visión los cuales han levantado mucho interés en los últimos años debido a las grandes ventajas que ofrecen. Las cámaras proporcionan gran cantidad de información exterior permitiendo extraer datos 2D e incluso 3D. La calidad de esta información es mejor que la que puede ofrecer los sensores SONAR y el precio y consumo de energía menor que los láseres, esta ventaja es bastante significativa cuando se desea o necesita trabajar sin grandes baterías durante grandes periodos de tiempo. Por otro lado, su comportamiento es estable tanto en zonas de interior o exterior de edificios no como los GPS donde su comportamiento no es el adecuado en ciertas zonas. Por último, los sensores de visión ofrecen no solo ventajas en la tarea de mapping, localización y navegación, si no que también ofrecen otra serie de atributos como la detección de personas, reconocimiento de caras y la identificación de obstáculos y objetos específicos como puede ser puertas, escalones o luces.

Los sensores de visión pueden usarse como únicos sistemas de percepción del robot móvil o combinados con la información ofrecida por otro tipo de sensores. En la bibliografía son muchos los ejemplos que se puede encontrar de situaciones en las que se ha utilizado varios sistemas de percepción cooperando para realizar tareas de mapping, localización o navegación. Por ejemplo, Choi et al. [7] presentan un sistema que combina un sistema SONAR con información visual para realizar tareas de navegación o Hara et al. [8] que combina la información visual con medidas láser. Muchos autores han expuesto la gran variedad de soluciones y propuestas ofrecidas por los investigadores durante los últimos años presentando diferentes técnicas visuales, variación en la combinación de sensores y distintos algoritmos para procesar la información. La evolución de estas técnicas hasta mitad de los años 90 está altamente documentada en [9] donde se muestra las técnicas de mapping, localización y navegación investigadas. Por otro lado en [10] se complementa el trabajo con la evolución durante la década de los noventa.

### 2.1.3. Configuración de los sistemas de visión

Existen varias configuraciones a la hora de utilizar un sistema de visión dependiendo del número de cámaras y el campo de visión que ofrecen. En la literatura se ha propuesto sistemas basados en una sola cámara, sistemas binoculares e incluso configuraciones trinoculares. Con los sistemas binoculares y trinoculares es posible medir la profundidad de las imágenes, aunque se limita el campo de visión. Estos sistemas poseen un ángulo de visión delimitado y es necesario el uso de varias imágenes para poder tener información completa del entorno y poder crear un modelo. Sin embargo los sistemas que proporcionan información visual omnidireccional han ganado popularidad recientemente gracias a que proporcionan datos del entorno con un ángulo de visión de  $360^\circ$  alrededor del robot. Para la obtención de imágenes omnidireccionales se puede hacer uso de dos técnicas: realizar una composición de varias imágenes tomadas con diferentes direcciones o la opción más extendida que sería con una única cámara que refleja sobre una superficie espejo, esta técnica se le denomina sistema de visión catadióptrico.

Los sistemas catadióptricos están mucho más extendidos. Normalmente estos sistemas están formados por una cámara convencional y un espejo convexo que dependiendo de la forma y tamaño ofrecerá diferentes tipos de imágenes. Las imágenes obtenidas se denominan imágenes omnidireccionales sin embargo existen ciertas transformaciones que estas imágenes pueden sufrir para adoptar otras formas que quizá sean más útiles dependiendo de la tarea que vaya a realizar o los grados de libertad que el robot pueda tener. La información relativa a este tema se podrá encontrar con mayor profundidad en el apartado 2.2.

Los sistemas de visión omnidireccionales tienen grandes ventajas respecto a los sensores que se han descrito al inicio de la sección. Las características obtenidas por la imagen son mucho más estables ya que permanecen durante más tiempo en el campo de visión del robot mientras este se mueve. Además este tipo de sistemas proporcionan información necesaria para detectar la posición del robot independientemente de la orientación y mediante ciertos procesos es posible incluso estimar la orientación. Asimismo, los sistemas omnidireccionales contienen datos suficiente de la escena para poder extraer información si durante el recorrido algún objeto o persona obstruye parte de la imagen. Para concluir, los sistemas de visión omnidireccional proporcionan mayor información del ambiente que otros sistemas de visión y permiten crear un modelo de la escena con menor cantidad de imágenes.

### 2.1.4. Modelos visuales en la actualidad

Con las imágenes que el sistema de visión nos ofrece se puede crear un modelo del entorno. Existen tres tendencias a la hora de construir un modelo que represente y organice el espacio: los mapas métricos, los mapas topológicos y los mapas híbridos o de organización compuesta.

Dentro de los mapas métricos se puede distinguir diferentes formas de representación. Los mapas basados en celdas de ocupación dividen el entorno en celdas y a estas se les da una probabilidad de ocupación. Por otro lado están los mapas basados en características en los que el entorno se representa por un conjunto de puntos destacados a los que se les denomina *landmarks*. El robot reconoce estos puntos y conoce su posición exacta en coordenadas del mundo por tanto se podrá posicionar relativamente a ellos. Estos mapas no hay fronteras solo puntos detectados con sus coordenadas formando así una representación compacta y escalable.

Los mapas topológicos por su parte no ofrece ninguna información estructural ni sistema de coordenadas, son simples grafos en los que los nodos adyacentes aparecen conectados permitiendo una navegación entre nodos consecutivos. En comparación con los mapas métricos los mapas topológicos ofrecen una localización más tosca con un coste computacional razonablemente bajo.

Por último los mapas híbridos ofrecen una representación de la información compuesta en diferentes capas con distinto nivel de detalle. El modo de utilización suele ser combinar mapas topológicos en las primeras capas realizando una localización más basta y a continuación entrar en una capa con mapa métrico para afinar la localización teniendo que comparar un menor número de imágenes. Esta tendencia trata de combinar las ventajas de los mapas topológicos, como el bajo coste computacional, y de los mapas métricos, como es la localización más exacta.

Cabe recordar que la creación de modelos con información visual tiene algunas desventajas reseñables. Por un lado, el propio movimiento del robot puede interferir en el escenario haciendo que el entorno se modifique. Por otro lado, otras circunstancias presentes en condiciones normales como los cambios de iluminación natural entre días, alteraciones producidas por los humanos u oclusiones pueden producir en la escena cambios substanciales que impidan el buen funcionamiento de la localización. Teniendo esto en cuenta es necesario llegar a un modelo en el que independientemente de estos fenómenos se pueda extraer información relevante de la escena. Esta información independiente será útil para identificar la posición y orientación del robot móvil. La obtención de esta información relevante e independiente sigue dos vertientes: la extracción de puntos característicos basados en características locales de la imagen, una opción bastante madura y estudiada y la otra opción es hacer uso de los

descriptores globales y crear un descriptor de apariencia global que comprima la información de la imagen en un único vector. En el capítulo 3 se explicará y discutirá sobre este problema.

## 2.2. Sistemas de visión omnidireccional

Como se ha expuesto en la sección anterior, la principal ventaja de usar cámaras omnidireccionales en aplicaciones robóticas es la posibilidad de obtener imágenes con un amplio campo de visión. Por un lado, en esta sección se va a explicar las configuraciones que permiten obtener información con un campo de visión completo. Por otro lado, se exponen los diferentes formatos con los que se puede expresar esta información y su aplicación robótica.

Actualmente hay diferentes mecanismos que permiten capturar imágenes omnidireccionales: lentes de ojo de pez, sistemas catadióptricos... Por ejemplo, los sistemas *ladybug* hacen varias tomas de la imagen desde diferentes puntos y con ello reconstruyen la imagen omnidireccional. Se puede observar este sistema en la figura 2-1 y comprobar que está formado por diferentes cámaras orientadas hacia distintas direcciones.



**Figura 2-1.:** Sistema de captación de imágenes Ladybug.

Otra posibilidad son los ojos de pez. Este mecanismo está compuesto únicamente por una cámara capaz de captar imágenes con un campo de visión mayor a 180 grados. El sistema *Gear 360* que se puede ver en la figura 2-2 combina dos lentes de ojo de pez que conjuntamente logran la imagen omnidireccional.

Estos métodos son eficaces y devuelven la información deseada. El problema es que obtener imágenes esféricas por combinación de varias cámaras no es una tarea trivial y supone un esfuerzo grande ya que los parámetros de la cámara y las condiciones exteriores como la luminosidad deben de ser iguales. El último mecanismo que se presenta es el sensor de visión catadióptrico. Se trata del sistema más usado y combina una cámara y un espejo convexo. La escena se refleja en el espejo y se capta con la cámara. Dada la importancia que tiene este mecanismo de visión en robótica se le dedica la siguiente sección a la explicación del sistema.



Figura 2-2.: Sistema de captación de imágenes Gear 360.

### 2.2.1. Sistemas de visión catadióptrico

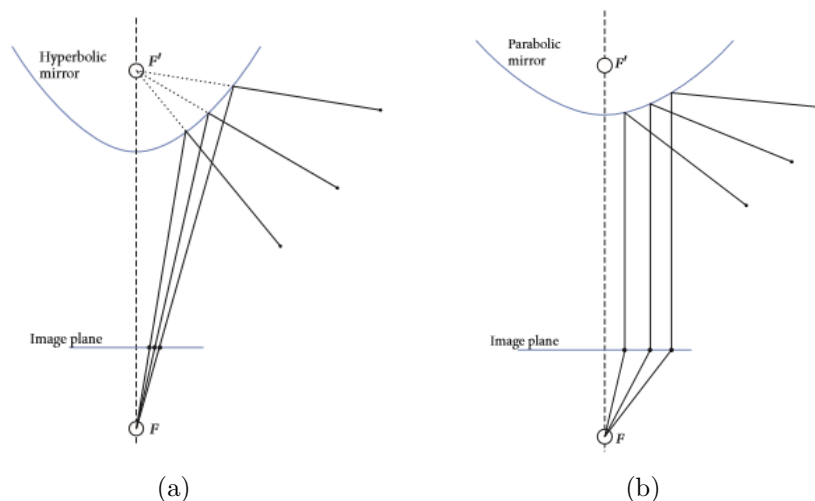
Los sistemas catadióptricos utilizan un espejo convexo para reflejar la información del entorno y de esta manera la cámara obtiene un campo de visión omnidireccional. La superficie convexa puede tener diferentes geometrías: esférica, hiperbólica, parabólica, elíptica o cónica. La forma, posición y orientación de esta superficie respecto de la cámara va a definir las proyecciones de la información de la escena en la cámara.

El centro de proyección del sistema y la calibración son los grandes elementos a tener en cuenta a la hora de usar un sistema de visión catadióptrico para realizar tareas de mapping y localización.

Por un lado, cuando utilizamos estos sistemas es importante tener un único punto de proyección (que constituya el centro de la cámara). En estos sistemas todos los rayos que llegan al espejo convergen en un único punto que será el centro óptico. Esta característica es importante ya que gracias a ella se obtienen las imágenes sin distorsión. De las posibles geometrías que el espejo convexo puede adoptar existen dos configuraciones que satisfacen esta característica. Combinar una superficie hiperbólica con una cámara con lente convencional o modelo pin-hole, como se muestra en la figura 2-3 a, o un sistema formado por un espejo parabólico y una cámara con lente de proyección ortográfica, figura 2-3 b. En ambas opciones  $F$  y  $F'$  son los focos de la cámara y el espejo. Con el resto de configuraciones los rayos convergen en diferentes puntos dependiendo de su incidencia; esto no supone un gran error si el objeto está alejado de la cámara, sin embargo sí supondrá un error grande cuando está cerca del sistema.

Por otro lado, es importante conocer los parámetros de la cámara y el sistema dado que algunos algoritmos de mapping y localización requieren una calibración adecuada para funcionar correctamente. [11]. El proceso de calibración puede realizarse de dos formas diferentes. El primer proceso puede ser el cálculo de los parámetros intrínsecos de la cámara, los coeficientes de los espejos y la posición relativa entre cámara y espejo. La otra forma es





**Figura 2-3.:** Modelo de proyección de sistemas catadióptricos con un único punto de proyección. a: Espejo hiperbólico y lente convencional. b: Espejo parabólico y lente ortográfica.

conocer una lista de correspondencias entre cada píxel de la imagen y el rayo de luz de la escena que se ha proyectado en dicho píxel.

### 2.2.2. Proyecciones de la información omnidireccional

La imagen omnidireccional que se utilizará en las tareas de mapping y localización contiene la información capturada por la cámara del sistema de visión catadióptrica. Esta información omnidireccional puede ser adquirida y tratada de diferentes formas como veremos a continuación. La imagen final puede ser considerada como una representación polar del entorno con origen en el que el foco se refleja en el plano imagen. La figura 2-4 muestra dos ejemplos de sistemas catadióptricos y la imagen omnidireccional que ofrecen, ambos sistemas están compuestos por una cámara y un espejo hiperbólico. El espejo del sistema a es el modelo *Wide 70* fabricado por *Eizoh* y el sistema b muestra el modelo *Super-Wide View* creado por *Accowle*.

Scaramuzza et al. [12] describen un método que permite pasar la imagen omnidireccional que se obtiene del sistema catadióptrico a una imagen panorámica que es más familiar para el ser humano. El método está basado en la extracción de los radios de la imagen omnidireccional y hacerlos corresponder con líneas verticales de la imagen.

De la imagen omnidireccional inicial es posible obtener distintas representaciones ya que se puede proyectar la información visual en diferentes planos y superficies generando así



**Figura 2-4.:** Ejemplo de dos sistemas de visión omnidireccional.

diferentes perspectivas de la escena original. Cada proyección representará propiedades específicas útiles para diferentes tareas de navegación.

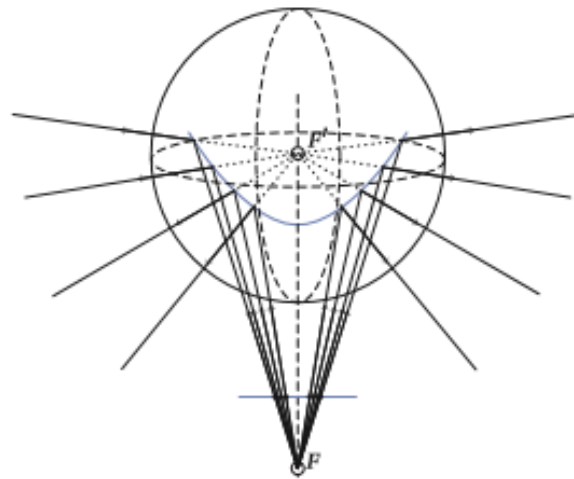
En los próximos párrafos se presentan las proyecciones más importantes y los trabajos que se han desarrollado con ellas en el campo de la robótica. Se puede encontrar una descripción matemática completa de estas proyecciones en [13].

Una información omnidireccional puede ser proyectada en una *esfera unidad* con centro en el foco del espejo. Para calcular cada píxel de la esfera se toma el valor del rayo de luz que tiene la misma dirección hacia el foco del espejo que el radio de la esfera hacia ese píxel. La figura 2-5 muestra el modelo de rayos del modelo esfera unidad cuando se utiliza un espejo hiperbólico y una lente convencional. La línea de color azul representa la superficie cóncava,  $F$  y  $F'$  son respectivamente los focos de la cámara y el espejo. Los píxeles de la imagen son re proyectados en el espejo convexo y cada punto del espejo se proyecta en una esfera. Para poder realizar esta técnica es necesario una correcta calibración de la cámara.

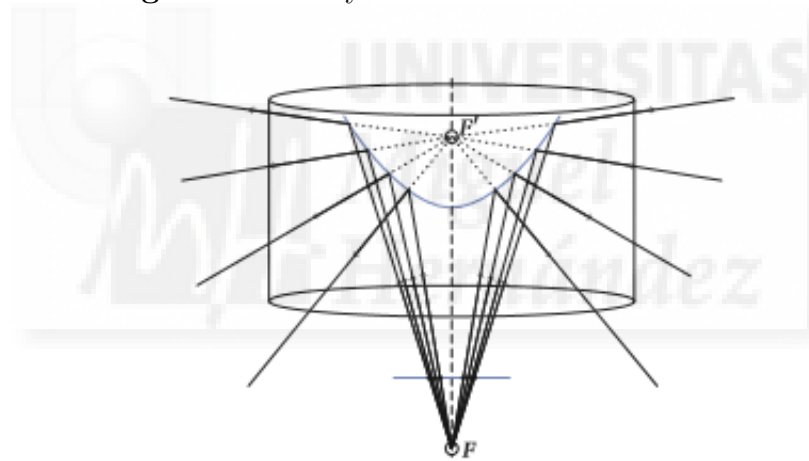
Este tipo de proyecciones se han utilizado tradicionalmente para estudiar rotaciones 3D en el espacio, para ello se hace uso de transformadas como la Transformación de Fourier Esférica.

La proyección *cilíndrica* consiste en proyectar la escena omnidireccional en un cilindro cuyo eje es paralelo al eje del espejo convexo. Conceptualmente se puede conseguir este efecto cambiando las coordenadas polares del sistema omnidireccional en coordenadas rectangulares. De este modo, cada circunferencia en la imagen omnidireccional será reconvertida en líneas horizontales en la nueva imagen.

La figura 2-6 muestra la proyección con el modelo cilíndrico con un espejo hiperbólico y una lente convencional. La línea de color azul representa el espejo y  $F$  y  $F'$  son los focos de la cámara y el espejo. Los píxeles de la imagen son re proyectados en una superficie cilíndrica. Para poder realizar esta técnica no es necesario realizar una calibración del sistema



**Figura 2-5.:** Proyección del modelo esfera unidad.



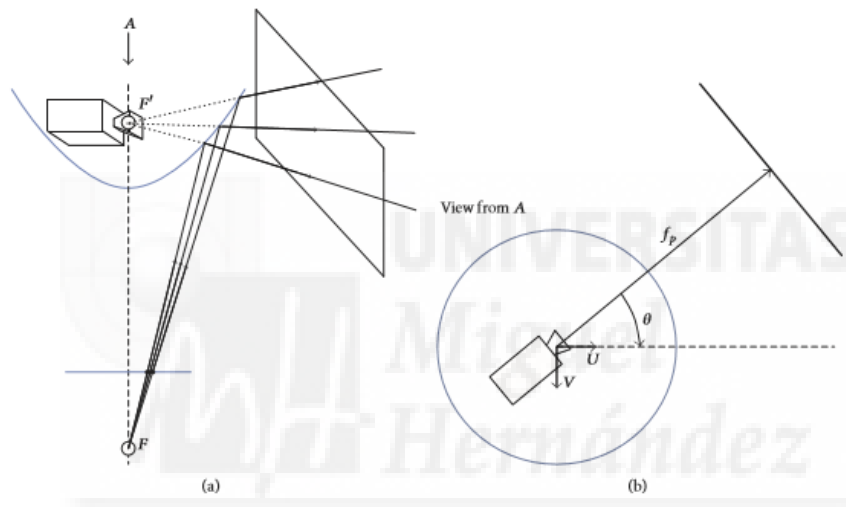
**Figura 2-6.:** Proyección del modelo cilíndrico.

catadióptrico.

El sistema de proyección cilíndrico es comúnmente conocido como imagen panorámica y es el más utilizado en robótica móvil para realizar tareas de mapping y localización, SLAM y navegación visual. La razón por la cual este tipo de proyección es el más utilizado es porque las imágenes panorámicas son más comunes y fáciles de entender para el ojo humano y ello ha permitido que sea posible utilizar en ellas las mismas técnicas y algoritmos que se emplean en las imágenes en perspectiva.

Por último tenemos la proyección en *perspectiva* que consiste en obtener imágenes proyectadas en cualquier dirección. El resultado será como capturar una imagen con una cámara virtual convencional colocada en el foco del espejo.

La figura 2-7 muestra la proyección del modelo en perspectiva con un espejo hiperbólico y una lente convencional. El espejo se representa con una línea azul y  $F$  y  $F'$  son respectivamente los focos de la cámara y el espejo. Los píxeles del plano de imagen son re proyectados a el espejo y a continuación cada punto del espejo se proyecta en un plano deseado. Esto es equivalente colocar una cámara convencional en  $F'$ . Para realizar esta proyección en perspectiva es necesario calibrar el sistema ya que se necesita que la cámara esté calibrada para realizar el paso de la re proyección en el espejo.



**Figura 2-7.:** Proyección del modelo en perspectiva.

La proyección *ortográfica* es un caso particular de la proyección en perspectiva. En esta proyección el plano se sitúa perpendicular al eje de la cámara y el eje de la cámara paralelo al eje del espejo; es decir, es equivalente a colocar una cámara virtual en el foco del espejo apuntado hacia el suelo. La figura 2-8 muestra este tipo de proyección particular en la que los píxeles se re proyectan en un plano horizontal al suelo.

Para terminar con este capítulo se presenta la figura 2-9 en ella se muestra una imagen omnidireccional (figura a) capturada por el sistema catadióptrico descrito en la figura 2-4(a). A esta imagen se le ha realizado una serie de re proyecciones que se muestran en (b) proyección esfera unidad, (c) proyección cilíndrica, (d) proyección en perspectiva con un plano vertical, (e) proyección ortográfica.

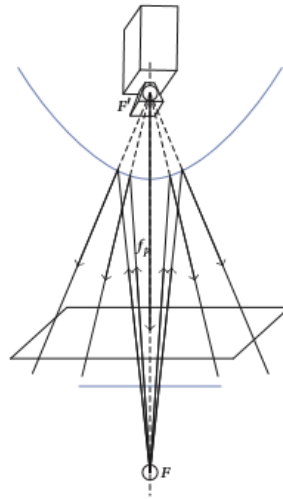


Figura 2-8.: Proyección del modelo ortográfico.

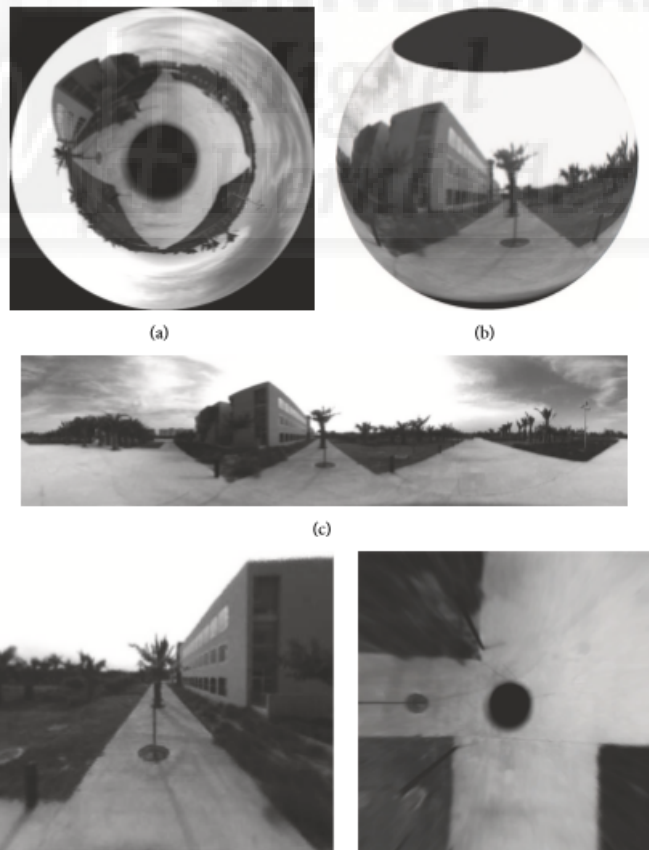


Figura 2-9.: (a)Imagen omnidireccional y transformaciones calculadas sobre ella: (b)proyección esfera unidad, (c)proyección cilíndrica, (d)proyección en perspectiva en un plano vertical, y (e)proyección ortográfica.

## 3. Descripción de la información visual

### 3.1. Necesidad de descriptores

Existe mucha literatura sobre la utilización de imágenes omnidireccionales o cualquiera de sus proyecciones para la realización de tareas de robótica como mapping y localización. Las imágenes ofrecen una gran cantidad de información. Estos datos son bastante cambiantes tanto por cómo afecta el robot a la escena cuando se mueve como por cambios de luz en el ambiente o el cambio en la posición de algunos objetos. Teniendo esto en cuenta es necesario extraer información estable de la escena que permita solucionar los problemas de creación de mapas y de localización. Existen dos soluciones que permiten extraer información que dependen del método empleado para extraerla; las soluciones basadas en características locales y las soluciones basadas en características globales de la imagen. La solución basada en características locales o puntos característicos es una solución bastante madurada y estudiada a lo largo de los años, sin embargo la opción de los descriptores globales está siendo una opción bastante novedosa la cual está demostrando ofrecer una alternativa robusta.

Existen muchos algoritmos para trabajar con ambos tipo de soluciones. Cada uno de los algoritmos tiene una serie de parámetros asociados. Al variar estos parámetros las soluciones dadas en la creación de mapas y localización varían. En los últimos capítulos del presente trabajo se realiza un estudio sobre la variación de los parámetros en los métodos utilizados. De momento, en las próximas subsecciones se detalla los descriptores basados en puntos característicos y los descriptores basados en apariencia global.

#### 3.1.1. Metodos basados en la extracción y descripción de características locales

Las características locales están basadas en la extracción de puntos, objetos o regiones características de la escena. Cada característica se describe en un descriptor que será invariante a cambios en la posición y orientación del robot. Una vez detectado y descrito el punto característico se puede resolver el problema de localización. Primero se extraen características a lo largo de la escena y se detecta zonas susceptibles a que las características no varíen, a continuación se realiza un método de comparación entre imágenes y se realiza la localización.

Existen diferentes filosofías a la hora de utilizar estos descriptores a la hora de elegir características o el proceso para encontrar la coincidencia. Por ejemplo, Pears and Liang [14] extraían esquinas localizadas en el suelo y utilizaban homografías para realizar el seguimiento. Otros autores detectan características predeterminadas como ventanas o extraen características de forma más compleja como con el detector de esquinas Harris [15]. También es posible que se utilicen técnicas como el filtro de Kalman para realizar el seguimiento y correspondencia de características.

Entre los descriptores de extracción de puntos característicos destacan los descriptores SIFT y SURF. Por un lado, el descriptor SIFT (Scale Invariant Feature Transform) fue desarrollado por Lowe [16, 17] y proporciona características invariantes a escala, rotación, cambios en condiciones de luz y del punto de vista de la cámara. Por otro lado, el descriptor SURF (Speeded-Up Robust Features), desarrollado por Bay et al. [18, 19]. Los puntos SURF están inspirados en los SIFT pero presentan un coste computación más bajo y una robustez mayor ante transformaciones. Recientemente han aparecido otros descriptores como BRIEF [20], el cual está diseñado para realizar operaciones en tiempo real; ORB [21], el cual se basa en BRIEF pero mejora la invariancia ante rotaciones y ruidos, pero no es tan robusto ante cambios de escala; BRISK [22] and FREAK [23], que tratan de tener la robustez de los puntos SURF pero con una mejora en el tiempo computacional.

Estos descriptores han sido muy utilizados para la creación de mapas y solucionar problemas de localización. Autores como Angeli et al.[23] y Se et al.[24] utilizaron características SIFT para realizar estas tareas o Murillo et al.[25] y Valgren y Lilienthal [26] que con la extracción de puntos SURF estiman la posición de un robot en un mapa previamente construido.

La utilización de esta técnica con puntos locales característicos combinada con técnicas de probabilidad permite construir mapas métricos. Sin embargo, este método presenta algún inconveniente. El principal es que en la imagen debe haber gran cantidad de detalles susceptibles de ser puntos característicos, además, la detección de características no es un método completamente robusto ante cambios en el entorno por ello su descripción no es absoluta e incorpora variaciones cuando hay un cambio en la posición u orientación del robot. Por último, la extracción, descripción y comparación de puntos característicos no es una tarea fácil computacionalmente hablando ya que se trata de un proceso complejo que normalmente impide realizar tareas como la creación de un mapa en tiempo real mientras el robot explora el escenario.

Estas características, junto a que la cámara debe de estar calibrada correctamente ya que un fallo en los parámetros intrínsecos o extrínsecos del sistema añaden un error de medida, hacen que durante los últimos años se haya investigado en los descriptores de apariencia

global.

### 3.1.2. Metodos basados en la descripcion global de la escena

Los métodos basados en descriptores globales trabajan con la imagen completa sin extraer ningún tipo de información local. Cada imagen se representa por un único descriptor que contiene información global de la escena. Este método ofrece ventajas en ambientes dinámicos o con pocas características donde es difícil detectar puntos o regiones estables. El describir cada imagen como un único vector favorece las tareas de localización y creación de mapas ya que reduce la tarea a una simple comparación de vectores.

Este método de descripción y compresión permite optimizar el tamaño de la base de datos. El almacenamiento y comparación de escenas se realizan con un coste computacional más eficiente. Además, estos descriptores no presentan invariancias producidas por la orientación del robot; es decir, si el robot está localizado en el mismo lugar pero con orientación diferente el descriptor que obtenemos es el mismo. Esta será la principal ventaja en el uso de estos descriptores.

Por otro lado, el uso de los descriptores globales supone una dificultad a la hora de extraer datos de medida. Estas técnicas se acercarían más a formar mapas topológicos pudiendo crear únicamente mapas métricos con la combinación de otra serie de medidas como podría ser información laser, odometría...

Entre las técnicas de descripción global de la imagen encontramos descriptores como PCA (Principal Components Analysis) [27] una de las primeras alternativas robustas que aparecieron. PCA considera la imagen como un conjunto de información multidimensional que puede ser proyectada en dimensiones bajas del espacio reteniendo información original pero únicamente la más importante. Pero el descriptor PCA tiene dos grandes desventajas. Por un lado, el descriptor depende de la orientación del robot, característica que se ha comentado con fundamental en los descriptores globales. Por otro lado, es necesario todas las imágenes del mapa para poder crearlo lo que conlleva a no poder crear mapas online mientras el robot explora la estancia.

Otros autores han propuesto y utilizado otros descriptores como aplicar la transformada de Fourier a la imagen para extraer la información más relevante, de esta operación obtenemos el descriptor FS (Fourier Signature). Existen varias alternativas a la hora de trabajar con la transformada de Fourier. Por un lado, si se realiza la transformada a imágenes panorámicas encontramos autores que han escrito sobre los resultados obtenidos como Payá et al. [28]. Por otro lado, también es posible aplicar la transformada de Fourier esférica (SFT) a la



imagen omnidireccional como en el trabajo propuesto por Rossi et al. [29]. En ambos casos el resultado del descriptor es la compresión de la información original en un menor número de componentes.

Otros autores han utilizado otra serie de técnicas para realizar estas descripciones como puede ser la operación gradiente o con la convolución de filtros. De esta manera aparecen los métodos HOG (Histograms of Oriented Gradients) o *gist*.

Por último. Como se ha expuesto, los descriptores de apariencia global son más útiles a la hora de crear mapas y realizar tareas de localización pero se debe considerar que la apariencia global de las imágenes depende mucho de las condiciones de iluminación. Se cree que la descripción global de las imágenes no va a tener grandes variaciones con cambios en las condiciones de iluminación ni con pequeños cambios en la escena como cambios en la posición de los objetos y puertas, pequeñas oclusiones... Parte de este trabajo se dedicará a realizar un estudio de este fenómeno comprobando si realmente es factible utilizar estos descriptores para dicha finalidad. Estos serán los principales puntos de estudio de los capítulos 5 y 6 del trabajo.

## 3.2. Descripción matemática de los descriptores de apariencia global

El objetivo de esta sección es doble. Por un lado se expone un estado del arte de los descriptores de apariencia global. Por otro lado, se explica una breve descripción matemática del método utilizado. Durante este trabajo se estudian cuatro métodos los cuales se explican en esta sección: en la subsección 3.2.1 se explica un método basado en la transformada de Fourier, el método HOG basado en el histograma del gradiente se expone en la subsección 3.2.2, en la 3.2.3. habla sobre el método *gist* basado en filtros de Gabor y por último en la subsección 3.2.4. se expone el método Radon.

Se considera que el movimiento planar de un robot móvil que captura imágenes a través de una cámara omnidireccional en lo alto de la estructura. Este sistema de visión está compuesto por una cámara y un espejo convexo con sus ejes alineados con el eje vertical. Dicho sistema se explica más detalladamente en el capítulo 4.

### 3.2.1. Descriptor basado en la Firma de Fourier

La transformada de Fourier Discreta (DFT) ha sido utilizada por muchos investigadores para extraer la información más relevante de las escenas. Cuando se calcula la transformada 2D-DFT de una escena  $f(x,y)$  se obtiene una nueva matriz  $\mathbf{F}(u,v)$ , donde  $u,v$  son las frecuencias variables. De esta nueva matriz se puede obtener una matriz de módulos  $\mathbf{A}(u,v)=\|\mathbf{F}(u,v)\|$  y una matriz de argumentos  $\Phi(u,v)$ . Por un lado, la matriz de magnitudes contiene información de la aparición o no de las características principales que aparecen en la escena sin tener la capacidad de localizar estas características además ofrece otro tipo de información 'no-localizada' como orientaciones, suavizados y tamaño y profundidad de contorno. Por otro lado, la matriz de argumentos contiene información local de propiedades de la imagen (sombras y posición de los objetos que aparecen en la escena). Teniendo esto en cuenta,  $\mathbf{A}(u,v)$  puede considerarse como un descriptor de apariencia global de una localización específica, ya que contiene la información estructural dominante de la escena. Muchos autores han mostrado como hacer tareas de clasificación sencillas con este tipo de información 'no-localizada'. Sin embargo, este descriptor no es completo ya que no tiene información espacial sobre estructuras de la imagen. Para cubrir esa falta de información algunos autores han propuesto algún tipo de solución, por ejemplo Oliva y Torralba [30] propusieron una ventana 2D que realizase la transformada de Fourier, esto permite la definición de ventanas circulares para la extracción de información espacial alrededor de píxeles específicos en la escena. Ishiguro y Tsuji [31] propusieron otra alternativa denominada Firma de Fourier (Fourier Signature, FS) diseñada para usarse en imágenes panorámicas y Stürzl [32] et al expusieron un algoritmo visual basado en la Firma de Fourier pero la imagen panorámica se reduce antes a un vector unidimensional.

La Firma de Fourier permite obtener un descriptor invariante a rotaciones del robot en el plano en el que este se mueve. El proceso de descripción comienza en la escena panorámica  $f(x,y) \in \mathbb{R}^{N_1 \times N_2}$ . Inicialmente la imagen puede ser submodificada para obtener un menor número de filas  $k_1 \downarrow N_1$  ( $k_1=1$  en [32]). La firma de Fourier de la escena resultante  $f(x,y) \in \mathbb{R}^{k_1 \times N_2}$  es la matriz  $\mathbf{F}(u,v) \in \mathbb{C}^{k_1 \times N_2}$  obtenida después de calcular la transformada unidimensional DFT de cada fila. En el dominio de la frecuencia la información más importante se encuentra concentrada en las bandas de frecuencia más bajas y las bandas de frecuencia altas tienden a estar contaminadas por el posible ruido existente en la imagen original. Teniendo esto en cuenta, se puede retener las  $k_2$  primeras columnas y descartar las demás consiguiendo una compresión bastante efectiva. La nueva matriz compleja tiene  $k_1$  filas y  $k_2$  columnas y puede expresarse como una matriz de módulos  $\mathbf{A}(u,y)=\|\mathbf{F}(u,y)\|$  y una matriz de argumentos  $\Phi(u,y)$ .

Cuando dos imágenes panorámicas han sido capturadas desde el mismo punto pero con una orientación sobre el eje vertical diferente ambas imágenes presentan la misma matriz de

módulo y la matriz de argumentos puede ser utilizada para estimar la orientación relativa del robot debido a que será igual pero desplazada cierto número de columnas. Gracias a esta propiedad, la matriz  $\mathbf{A}(u,y)=\|\mathbf{F}(u,y)\|$  puede ser considerada como un descriptor de posición del robot (debido a que es invariante a la rotación) y la matriz  $\Phi(u,y)$  puede considerarse como un descriptor de la orientación del robot (debido a que permite estimar la orientación). De este modo se puede estimar la posición y la orientación independientemente o de forma secuencial. Primero se estima la posición del robot con ayuda de la matriz de módulos, a continuación se estima la orientación relativa con la matriz de argumentos.

Durante el estudio realizado no se ha hecho ninguna reducción inicial a la imagen por ello el descriptor de posición es la matriz  $f(u,y) \in \mathbb{R}^{N_1 \times k_1}$  y el descriptor de orientación la matriz  $\Phi(u,y) \in \mathbb{R}^{N_1 \times k_2}$ . En los experimentos, diferentes valores de  $k_1$  y  $k_2$  son considerado para detectar la influencia de estos parámetros en la precisión y coste computacional del proceso de localización y estimación de la orientación.

### 3.2.2. Descriptor basado en el Histograma de Orientación del Gradiente. HOG

La técnica HOG considera la orientación del gradiente en partes específicas de la escena y destaca por su simplicidad, coste computacional y precisión. Sin embargo la experiencia con este tipo de descriptores para localización de robots móviles es limitada y se restringe a ambientes simples y pequeños.

La idea bajo este descriptor es que la apariencia y sombras de los objetos en una imagen puede ser descrita por la intensidad de los gradientes y sus direcciones. La implementación básica consiste en dividir la imagen en regiones (celdas) pequeñas y conectadas y compilar un histograma de gradientes con distintas orientaciones usando la información de los píxeles conexos. En sus inicios HOG fue descrito por Dalal y Triggs [33], quienes lo usaban para detectar personas en secuencias de imágenes. Después, muchos investigadores presentaron y mejoraron la versión tanto en detección como en coste computacional [34].

Hofmeister et al [35] dieron uso al descriptor HOG para solucionar problemas de localización en pequeños robots móviles, en ambientes visuales sencillos y cuando la orientación del robot era similar a la orientación que tenía el robot en la imagen con la que se compara.

Originalmente, HOG está hecho para describir pequeñas áreas locales de la escena. Payá et al [36] redefinen el descriptor como un descriptor de apariencia global utilizando un exhaustivo número de regiones que cubren toda la imagen y permite describirla globalmente.

De la imagen panorámica inicial, un descriptor de posición y orientación se obtiene utilizando la filosofía del histograma del gradiente. La imagen inicial  $f(x,y) \in \mathbb{R}^{N_1 \times N_2}$  se filtra para obtener dos imágenes que contienen los bordes horizontales  $f_1(x,y)$  y otra con los bordes verticales  $f_2(x,y)$ . De estas dos imágenes se obtiene la magnitud y la orientación del gradiente de cada píxel y se almacenan respectivamente en las matrices  $\mathbf{M}(x,y)$  y  $\Theta(x,y)$ . Desde ahora, las celdas se definen sobre la matriz  $\Theta(x,y)$ . Por un lado, para construir el descriptor de posición se define una serie de  $k_3$  celdas horizontales, con anchura igual a  $N_2$ , sin superposición y con anchura y cantidad de celdas coherentes para que toda la imagen sea cubierta. Por cada celda se completa un histograma con  $b_1$  recipientes denominados bins. Durante el proceso cada píxel de  $\Theta(x,y)$  se pondera con la magnitud correspondiente en su píxel de la matrix  $\mathbf{M}(x,y)$ . Al final del proceso el descriptor de posición estará definido de la siguiente manera  $\vec{h}_1 \in \mathbb{R}^{k_3 * b_1 \times 1}$ . Por otro lado, el descriptor de orientación se construye usando los mismos pasos pero considerando las celdas verticales, con tamaño  $N_1$  y anchura  $l_1$ . La cantidad de celdas verticales viene definido por  $x_1 = N_1/l_1$ . Tras calcular el histograma de orientación del gradiente y tras separar en  $b_1$  bins de la misma manera que en el descriptor de posición, el resultado del descriptor de orientación es  $\vec{h}_2 \in \mathbb{R}^{x_1 * b_1 \times 1}$ , aunque durante el estudio el parámetros que se varía es  $l_1$  y no  $x_1$ , aunque  $x_1$  variará por ser dependiente de  $l_1$ .

El descriptor  $\vec{h}_1$  es invariante a rotaciones del robot así que se considera como un descriptor visual de la posición del robot, y con la información que contiene  $\vec{h}_2$  es posible estimar la orientación relativa del robot respecto a una referencia.

### 3.2.3. Descriptor basado en Gist

El descriptor *gist* trata de imitar la habilidad de la percepción humana para reconocer una escena por identificar una región concreta por tener un color y/o textura que destaca respecto a la de sus vecinos. Este concepto fue introducido por Oliva y Torralba [37] con la idea de crear un descriptor de imagen con dimensión global. Muestran como utilizando este método de descripción, un espacio multidimensional puede ser creado donde las escenas que pertenecen a la misma categoría se proyectan en localizaciones cercanas. Matemáticamente, el método codifica la distribución espacial con medias de 2D-DFT de regiones uniformemente distribuidas por la imagen y utiliza PCA (Principal Components Analysis) para reducir la dimensión del descriptor final. Torralba et al. [38] proponen el uso de series de pirámides orientables en vez del cálculo de DFT.

Más recientemente el concepto '*prominencia*' aparece junto a *gist*. Siagian et al [39] intentan establece sinergias entre ambos conceptos creando un único descriptor cuyo coste computacional se ve reducido. Este descriptor se construye basándose en tres características de la escena: intensidad orientación y color.

El método utilizado en este estudio está basado en el de Siagian et al [39] y está descrito más profundamente en [36]. Está construido por información de orientación conseguida por un banco de filtro de Gabor con diferentes orientaciones que se pasa a diferentes niveles de resolución. Primero se consideran dos versiones de la imagen panorámica original: una como la original y otra con una resolución menor. Ambas imágenes son filtradas con  $m_1$  filtros de Gabor cuya orientación está distribuida entre 0 y 180 grados. Finalmente, para reducir la cantidad de información, los píxeles en cada imagen resultante se agrupan en bloques y se calcula la media de intensidad de los píxeles contenidos en cada bloque. La división de los bloques es idéntica a las celdas en el descriptor HOG. Primero  $k_4$  bloques horizontales definirán el descriptor de posición  $\vec{g}_1 \in \mathbb{R}^{2*k_4*m_1 \times 1}$ , que es invariante a rotaciones. Después, una serie de  $x_2$  bloques verticales definen el descriptor de orientación  $\vec{g}_2 \in \mathbb{R}^{2*x_2*m_2 \times 1}$ , donde el parámetro que se varía en este estudio no será el número de bloques verticales  $x_2$  si no la anchura de las venatanas  $l_2$ .

### 3.2.4. Descriptor basado en la transformada Radon

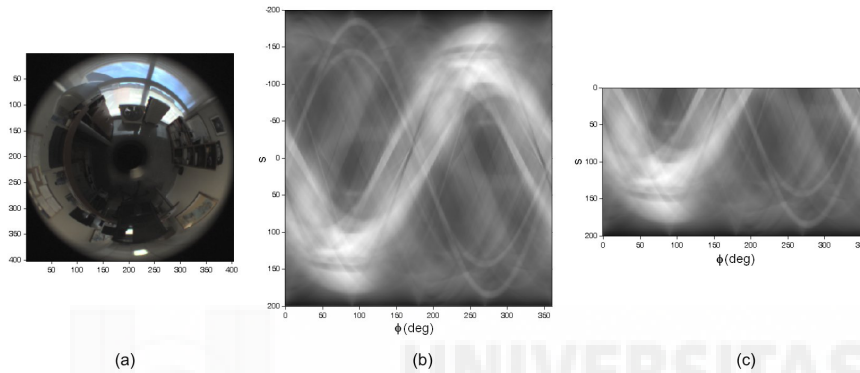
La transformada de Radon consiste en describir una función en términos de las proyecciones de sus integrales lineales. Una línea  $c_l$  puede ser parametrizada respecto su longitud de arco  $z$  como  $c_l := (x(z), y(z)) = ((z * \sin \Phi + s * \cos \Phi), (-z + \cos \Phi + s * \sin \Phi))$  donde  $s$  es la distancia del punto de  $c_l$  al origen y  $\Phi$  es el ángulo entre el vector perpendicular a  $c_l$  y el eje  $x$ . Teniendo estos parámetros en cuenta la transformada de Radon de una imagen  $f(x, y)$  puede obtenerse con la siguiente expresión.

$$R\{f(x, y)\} = r_{im}(\Phi, s) = \int_{-\infty}^{+\infty} f(x(z), y(z)) dz = \int_{-\infty}^{+\infty} f(z * \sin \Phi + s * \cos \Phi, -z + \cos \Phi + s * \sin \Phi) dz$$

Donde las coordenadas de la nueva función son  $(\Phi, s)$  que pueden ser consideradas como coordenadas de todas las líneas en  $\mathbb{R}^2$ . De este modo, una nueva función 2D se obtiene por la integración de la función original  $f(x, y)$  a lo largo de un grupo de líneas paralelas con distancia  $s$  entre cada una de ellas y diferente orientación  $\Phi$ . El tamaño del nuevo descriptor es  $r_{im} \in \mathbb{R}^{M_x \times M_y}$ , donde  $M_x$  es el número de orientaciones considerando  $\Phi = \{\Phi_1, \Phi_2, \dots, \Phi_{M_x}\}$  y  $M_y$  es el número de líneas paralelas. La transformada de Radon es invertible, la operación inversa reconstruye la función original.

La transformada de Radon presenta interesantes propiedades. La simetría y el poder para desplazar los datos son especialmente importantes desde el punto de vista de las imágenes omnidireccionales. Es posible obtener descriptores de apariencia global que permitan estimar la posición y la orientación. La figura 3-1 (a) muestra la imagen omnidireccional de tamaño  $N_x \times N_x$  y la figura 3-1 (b) la transformada de Radon de esta imagen que tiene tamaño

$360 \times N_x$ . Para obtener esta transformada la distancia entre líneas es igual a 1 píxel, de este modo  $s=1,2,3,\dots,N_x$  píxeles; y las orientaciones elegidas cubren toda la imagen  $\Phi=0,1,2,\dots,359$  grados. El resultado de la matriz es siempre antisimétrica, por ello podemos deshacernos de la mitad de la información ya que está duplicada. La figura 3-1 (c) muestra el descriptor final con un tamaño de  $360 \times 0,5 * N_x$



**Figura 3-1.:** Transformada Radon de una imagen omnidireccional. (a) imagen omnidireccional, (b) transformada Radon de la imagen y (c) transformada Radon final con tamaño  $360 \times 0,5 * N_x$ .

La propiedad de desplazamiento se puede observar en la figura 3-2. Esta figura 3-2 muestra tres imágenes omnidireccionales capturadas por el mismo robot en el mismo punto del suelo y con una orientación diferente alrededor del eje vertical. La figura muestra claramente el efecto que tiene la orientación en la transformada Radon. Si el robot rota  $\Delta\Phi$  grados, el nuevo descriptor presenta la misma información que el descriptor original pero con un desplazamiento de columnas igual a  $d=\Delta\Phi * M_x/360$ , con  $\Delta\Phi$  medido en grados.

Con esta característica, la transformada de Radon es un descriptor que contiene información de la posición y de la orientación del robot.

Para realizar las tareas de localización y creación de mapa la información resultante de la transformada de Radon puede ser tratada de diferente forma, esto da lugar a dos métodos para realizar tareas de navegación. El primer método hace uso de la Firma de Fourier y el método número 2..

La transformada de Radon no es invariante a la rotación como se ha visto en la figura 3-2. Una rotación pura produce un desplazamiento de las columnas en el descriptor. Este efecto debe ser tenido en cuenta durante la implementación del método de localización. En el **método 1** la Firma de Fourier (FS) se utiliza para transformar el descriptor RT y obtener un nuevo descriptor que es completamente invariante a rotaciones. Del descriptor de

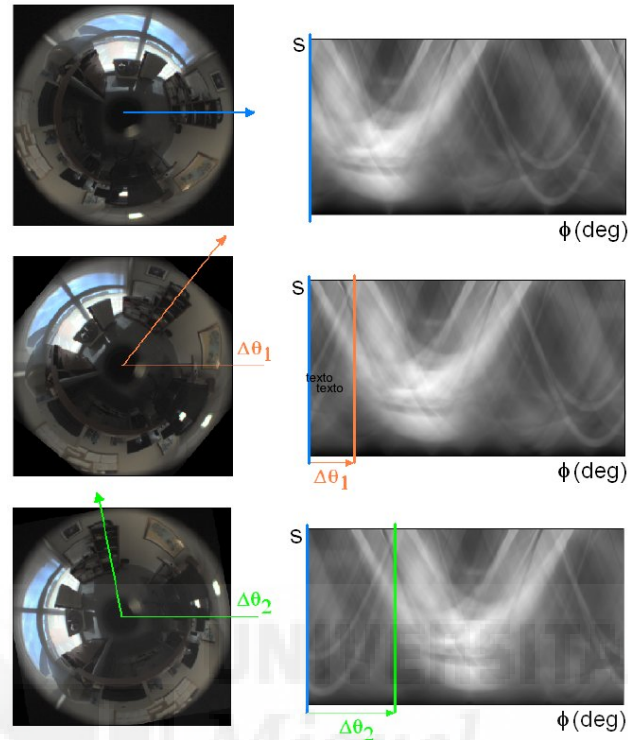


Figura 3-2.: Propiedad de desplazamiento en la transformada Radon.

la transformada de Radon se obtiene  $r_j \in \mathbb{R}^{M_x \times M_y}$  y a este descriptor se le pasa FS. Como resultado, cada nodo  $d_j, j = 1, \dots, n$  contiene dos matrices: una matriz de módulos  $A_j \in \mathbb{R}^{M_x \times k}$  y una matriz de argumentos  $\Phi_j \in \mathbb{R}^{M_x \times k}$ . La primera posee la información relevante a la posición y la segunda información relevante a la orientación y se emplea el mismo proceso que cuando se trabajaba únicamente con la firma de Fourier, tal y como se ha explicado en la subsección 3.2.1.

El **método 2** consiste en describir cada imagen con la transformada de Radon obteniendo un descriptor  $r_j \in \mathbb{R}^{M_x \times M_y}$  y usar el método POC (Phase Only Correlation) para comparar dos descriptores. De este modo cada nodo  $d_j, j = 1, \dots, n$  contiene únicamente una componente.

La localización y estimación de la orientación se puede resolver con la operación POC. Esta operación es llevada a cabo en el dominio de la frecuencia y proporciona un coeficiente de correlación que permite estimar la similitud entre dos escenas. Con POC es posible calcular tanto la similitud entre dos imágenes como el desplazamiento debido al cambio de fase.

En la figura refRadon3 observamos el resultado de la operación POC en dos descriptores  $r_1$  y  $r_2$  que habían sido descritos con la transformada de Radon. Ambas imágenes habían sido capturadas en el mismo punto pero con distinta orientación. El resultado de aplicar POC

entre  $r_1$  y  $r_2$  es una matriz  $C_{r_1r_2}$  cuyos elementos cogen valor entre 0 y 1. EL máximo calor de la matriz  $C_{r_1r_2}$  es el valor de similitud ente  $r_1$  y  $r_2$  (cuanto mayor sea el valor significa que más similares son las matrices) y es independiente de los cambios de orientación. La posición del máximo valor permite estimar la orientación relativa del robot, si se interpolación entre el número de columnas a la que se encuentra el máximo y se pasa a grados se es capaz de calcular el desplazamiento  $\Phi_{21}$ .

### 3.3. Parametros con influencia en el trabajo

Durante el estudio se va a evaluar los diferentes métodos y dentro de cada método se examina la influencia de diversos parámetros en el proceso de localización y estimación de la orientación. En la tabla **3-1** podemos encontrar como se denominarán dichos parámetros variables durante el presente trabajo.

**Tabla 3-1.:** Parámetros con influencia en los procesos de localización y orientación

Descriptor	Parámetros
Firma de Fourier	$k_1 \Rightarrow$ número de columnas del descriptor de posición.
	$k_2 \Rightarrow$ número de columnas del descriptor de orientación
HOG	$b_1 \Rightarrow$ número de bins por histograma en el descriptor de posición.
	$k_3 \Rightarrow$ número de celdas horizontales en el descriptor de posición.
	$b_2 \Rightarrow$ número de bins por histograma en el descriptor de orientación.
	$l_1 \Rightarrow$ anchura de las celdas verticales en el descriptor de orientación.
Gist	$m_1 \Rightarrow$ número de orientaciones del filtro de Gabor en el descriptor de posición.
	$k_4 \Rightarrow$ número de bloques horizontales en el descriptor de posición.
	$m_2 \Rightarrow$ número de orientaciones del filtro de Gabor en el descriptor de orientación.
	$l_2 \Rightarrow$ anchura de las celdas verticales en el descriptor de orientación.
Radon	$k_5 \Rightarrow$ número de columnas del descriptor de posición.
	$p_1 \Rightarrow$ grados entre ángulos consecutivos en el descriptor de posición.
	$k_6 \Rightarrow$ número de columnas del descriptor de orientación.
	$p_2 \Rightarrow$ grados entre ángulos consecutivos en el descriptor de orientación.



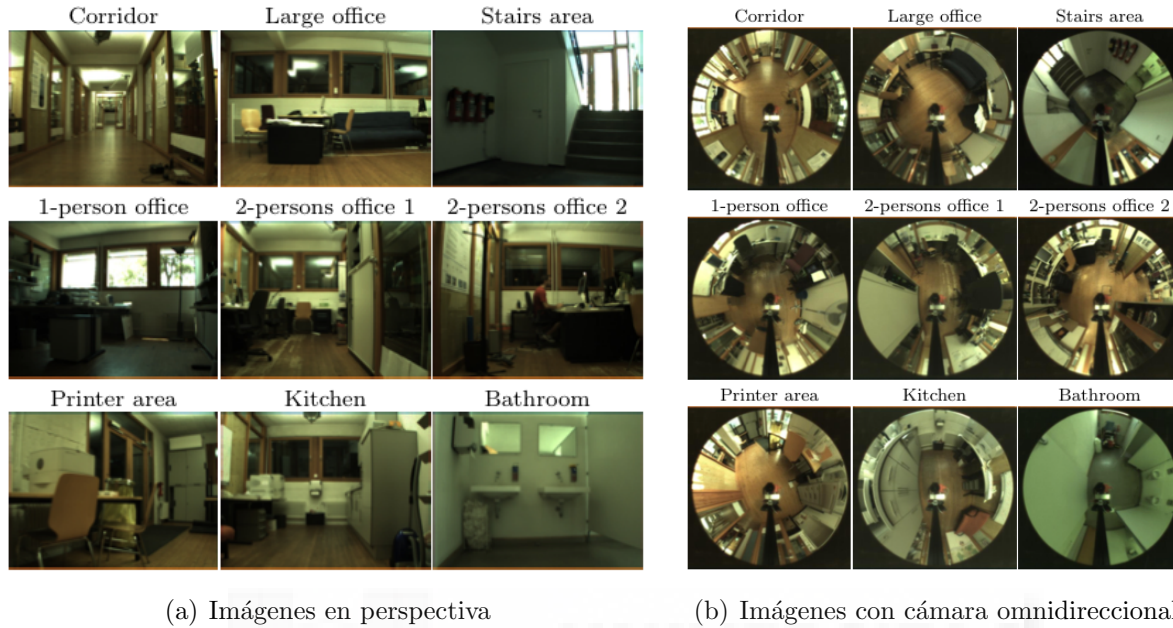
## 4. Material utilizado

En esta sección se expone la información relevante a las herramientas y equipos empleados durante los experimentos. Los experimentos han sido realizados con una base de datos ajena a nosotros, la base de datos con la que se ha decidido trabajar es con la base de datos COLD [1] la cual detallaremos a continuación. El capítulo se va a dividir en dos secciones, en la primera se describe la base de datos y como esta ha sido adquirida y en la siguiente se evalúa el sistema de adquisición de imágenes utilizado.

### 4.1. Base de datos

La base de datos COLD ofrece tres entornos de trabajo diferentes: 'Autonomus Intelligent Systems Laboratory' en la universidad de Friburgo, Alemania; 'Visual Cognitive Systems Laboratory' en la universidad de Ljubljana, Eslovenia y 'Language Technology Laboratory' en el Centro de Inteligencia Artificial alemán de Saarbrücken, Alemania. Las tres bases de datos ofrecen secuencias de imágenes capturadas con una cámara omni-direccional tomadas en diferentes momentos y bajo distintas condiciones e iluminación natural (durante tiempo nublado, soleado y de noche). En cada uno de los tres entornos ofrecidos, la adquisición de imágenes lleva un recorrido por el que pasa por estancias y habitaciones con diferente utilidad como podrían ser cocinas, despachos, pasillo, aseo... Consecuentemente, la base de datos COLD es un banco de pruebas ideal para evaluar la robustez de los algoritmos de localización y mapping ofreciendo no solo cambios de iluminación sino también cambios dinámicos (actividad humana, variación en la distribución y aspecto de los muebles...)

Finalmente se escogió la base de datos de Friburgo. La figura 4.1 muestra un ejemplo de las imágenes que se han adquirido en el interior de las habitaciones del laboratorio de Friburgo en 4.1 *a* las imágenes han sido capturadas con una cámara en perspectiva y en 4.1 *b* se han capturado con una cámara omnidireccional. Como se puede observar las habitaciones tienen diferentes propósitos y por ello estarán organizadas de manera distinta y no les afectará igual la actividad humana. Por ejemplo la cocina y una oficina tendrán organizaciones totalmente diferentes y habitaciones como el pasillo o la habitación de la impresora permanecerán bastante estables. Esto ofrece una situación que se asemeja bastante a la realidad, característica ideal para la realización del proyecto. La figura 4-2 muestra el mapa del ambiente interior del laboratorio de Friburgo. En el mapa se muestran dos rutas que se tomaron para la captura



**Figura 4-1.:** Imágenes adquiridas en la universidad de Friburgo

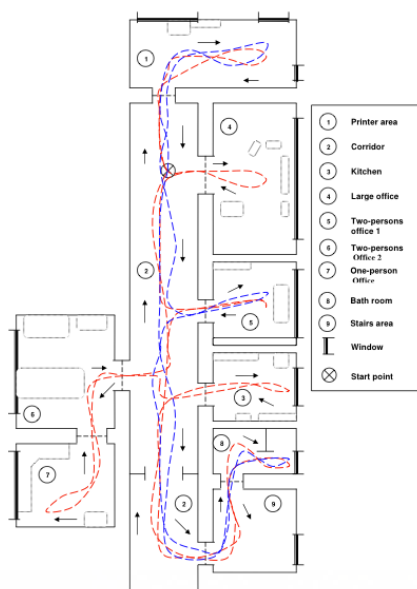
de imágenes, un camino corto de color azul y un camino más largo de color rojo. Para la realización de este trabajo se han empleado las imágenes capturadas durante el recorrido descrito por la trayectoria roja.

La base de datos COLD nos ofrece la estimación de la pose (posición 'x-y' y ángulo de giro) del robot para cada imagen adquirida durante el proceso. Esta información servirá de ayuda para estudiar el error cometido en la fase de localización ya que dispondremos de la posición real donde se adquirió la imagen y la que se ha detectado como localización. Este proceso se explica en el siguiente capítulo.

El primer paso para la realización del trabajo es crear un modelo donde se recoja el recorrido del robot móvil empleando una menor cantidad de imágenes. El robot se mueve a una velocidad media de 0.3 m/s y continuamente adquiere imágenes con un ratio de 5 imágenes por segundo. Para la creación del modelo se decidió coger una de cada cinco imágenes de la base de datos adquiridas durante un día nublado, de este modo nuestro modelo está creado por la adquisición de una imagen cada 30cm.

## 4.2. Sistema de adquisición de imágenes

Las capturas de imágenes en el laboratorio de Friburgo fueron realizadas con un sistema móvil equipado con una cámara omnidireccional. El móvil empleado se puede observar en Fig



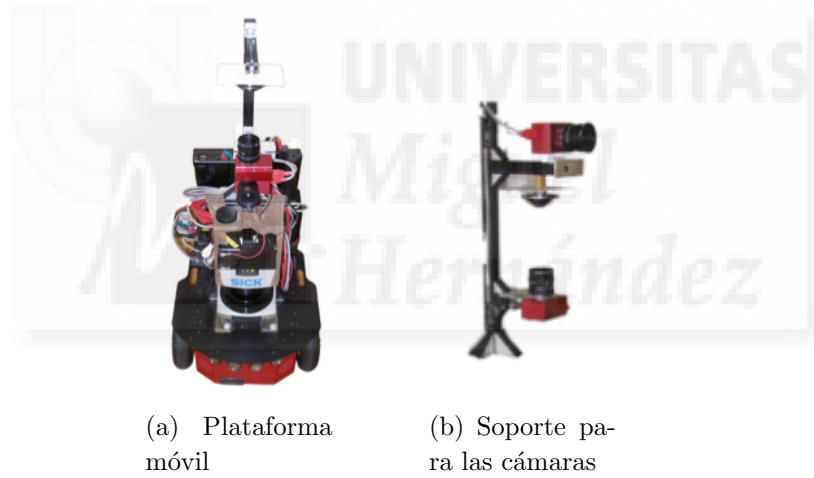
**Figura 4-2.:** Recorrido del robot móvil por el laboratorio de Friburgo.

**Tabla 4-1.:** Parámetros de las cámaras

Robot móvil	ActivMedia Pioneer-3 Friburgo	
Tipo de cámara	Perspectiva	Omnidireccional
Ratio	5 imágenes por segundo	
Resolución	640x480 píxeles	
Campo de visión	68.9°x54.4°	-
Altura de la cámara	66cm	91cm

**4-3** y se denomina ActivMedia Pioneer-3. Además, el sistema está equipado con un escáner laser SICK y encoders en las ruedas para detectar la odometría del robot. En el momento en el que se capturaron las imágenes el robot era controlado manualmente con un joystick.

La plataforma móvil estaba equipada con dos cámaras digitales 'Videre Design MDCS2'. La perspectiva omnidireccional se construyó con un espejo hiperbólico. Las dos cámaras y el espejo fueron montados juntos en un soporte portátil que se puede observar en Fig **4-3**. En la tabla **4-1** podemos encontrar los parámetros detallados sobre las cámaras utilizadas.



**Figura 4-3.:** Sistema ActivMedia Pioneer-3

## 5. Estimación de la posición del robot

En esta sección se revisa la eficacia de los descriptores de apariencia global haciendo una comparativa de la influencia de los parámetros principales que utiliza cada uno en la precisión y coste computacional. Para la realización de este estudio se ha hecho una gran cantidad de experimentos variando los parámetros y las condiciones de luminosidad en el ambiente. Esta sección se va a dividir en tres subsecciones. Primero, en la subsección 5.1., se hablará sobre los tipos de distancia y las medidas de error que se va a utilizar, es decir, el método usado en los experimentos. Después, en la subsección 5.2., se discutirá sobre los resultados obtenidos ante tres condiciones ambientales (Nublado, soleado y de noche). Por último, en la subsección 5.3. se realiza un estudio sobre el coste computacional de los experimentos.

### 5.1. Método

Durante el proceso de localización, el primer paso es comparar nuestra posición con las localizaciones almacenadas en la base de datos y obtener el vecino más cercano. Teniendo esto en cuenta, se utiliza un método que evalúe la habilidad de cada descriptor para identificar geoméricamente la posición del vecino más cercano de la base de datos de forma correcta.

Para obtener los k-vecinos más cercanos se pueden utilizar diferentes tipos de distancias. En este estudio se van a comparar los resultados obtenidos por cuatro distancias:  $d1$  es la distancia *cityblock*,  $d2$  es la distancia *Euclidea*,  $d3$  es la distancia de *correlación* y  $d4$  es la distancia *coseno*. La tabla 5-1 muestra la definición de cada tipo de distancias. En las expresiones matemáticas de las distancias,  $\vec{a} \in \mathbb{R}^{l \times 1}$  y  $\vec{b} \in \mathbb{R}^{l \times 1}$  son dos vectores donde:  $a_i, b_i, i = 1, \dots, l$ .

Primero se estudia la eficacia de los algoritmos, es decir la habilidad de poder calcular correctamente la localización del robot encontrando su vecino más cercano en el modelo (identificar geoméricamente la posición más cercana de las imágenes que se tienen guardadas en el modelo).

Una vez tenemos las imágenes descritas con el descriptor global deseado, el proceso es el siguiente. Primero se busca la imagen más próxima de las imágenes del modelo utilizando las cuatro medidas de distancia vistas. Una vez identificado el vecino más cercano del mapa, el

**Tabla 5-1.:** Medidas de distancia

Medida de distancia	Tipo de distancia	Expresión matemática
d1	Cityblock	$d_1(\vec{a}, \vec{b}) = \sum_{i=1}^l  a_i - b_i $
d2	Euclidean	$d_2(\vec{a}, \vec{b}) = \sqrt{\sum_{i=1}^l (a_i - b_i)^2}$
d3	Correlation	$d_3(\vec{a}, \vec{b}) = 1 - \frac{\vec{a}^T * \vec{b}_d}{ \vec{a}_d  *  \vec{b}_d }$
d4	Cosine	$d_4(\vec{a}, \vec{b}) = 1 - \frac{\vec{a}^T * \vec{b}}{ \vec{a}  *  \vec{b} }$

Donde:

$$\vec{a}_d = [a_1 - \bar{a}, \dots, a_l - \bar{a}]; \bar{a} = \frac{1}{l} * \sum_j * a_j$$

$$\vec{b}_d = [b_1 - \bar{b}, \dots, b_l - \bar{b}]; \bar{b} = \frac{1}{l} * \sum_j * b_j$$

error de localización se estima calculando la distancia geométrica entre el punto de captura de la imagen de test y el punto de captura del vecino más cercano.

Este proceso se repite para todas las imágenes de test, considerando los diferentes descriptores y distintas configuraciones de sus parámetros más relevantes. En el siguiente apartado se mostrarán los resultados obtenidos.

## 5.2. Resultados y discusión

Las figuras 5-1, 5-2, 5-3, 5-4 y 5-5 muestran los resultados del error medio en metros entre la posición del robot y la detectada como más cercana con imágenes tomadas un día nublado, mismo ambiente que las imágenes del modelo. Para una mejor resolución de las gráficas se añaden las figuras 5-6, 5-7 y 5-8 . En general se obtienen buenos resultados con errores del orden de los centímetros.

El comportamiento con la firma de Fourier es prácticamente constante con un error de 0,05 y 0,06 metros. Los más bajos se obtienen con  $d_3$  además de ser los resultados con menor varianza. Aunque los resultados son muy parecidos, para todas las distancias los mejores han sido los obtenidos con valores de  $k_1$  entre 4 y 8 siendo los de menor error al utilizar la distancia  $d_4$  y  $k_1 = 4$  donde el error es de 0.04967m.

Con HOG los resultados con mayor precisión los obtenemos con  $d_3$  y  $d_4$ . Los peores resultados se obtienen con valores de las líneas horizontales  $k_4$  bajos y son especialmente malos con  $d_1$  y  $d_2$  cuando los valores de  $k_4$  y de bins por histograma  $b_1$  son muy elevados. Si tenemos en cuenta las cuatro distancias, los mejores resultados se obtienen cuando los valores de las variables son intermedios rondando  $k_4 = b_1 = 16$  pero el mejor resultado absoluto se

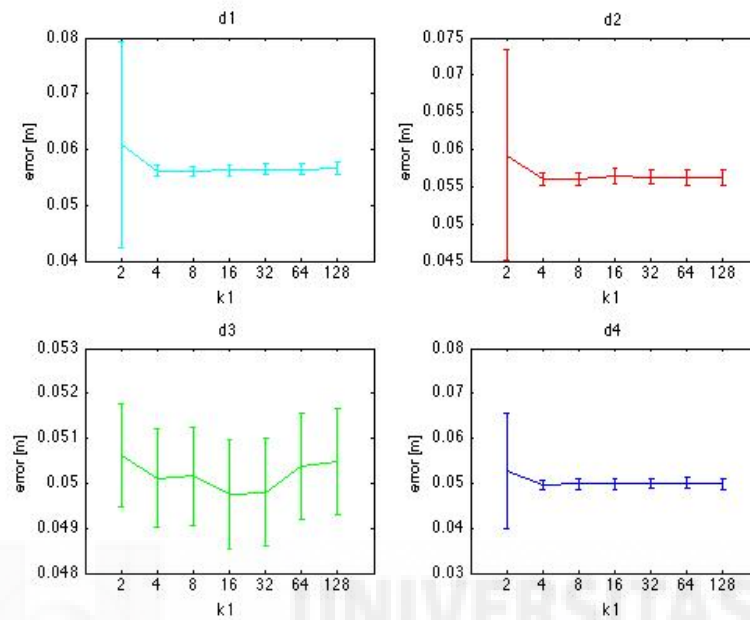


Figura 5-1.: FS. Error de localización con el **día nublado** al variar el número de columnas  $k_1$ .

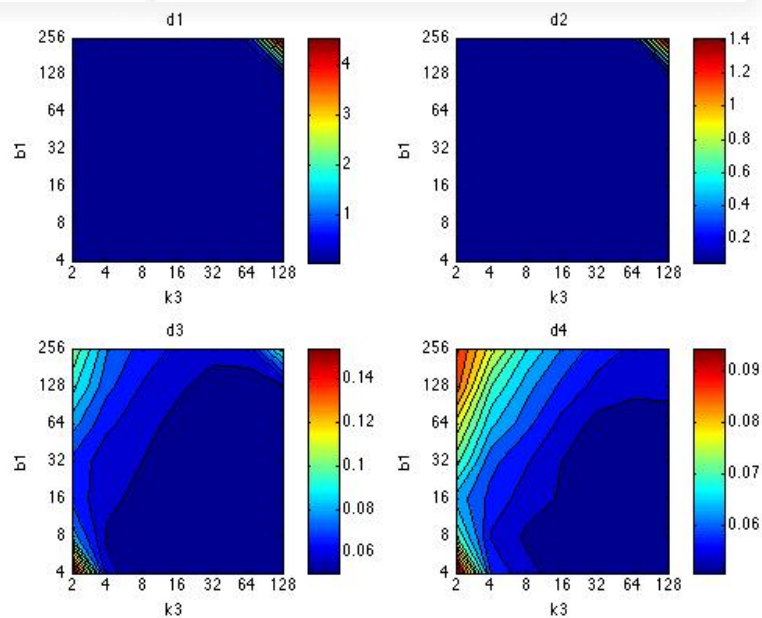
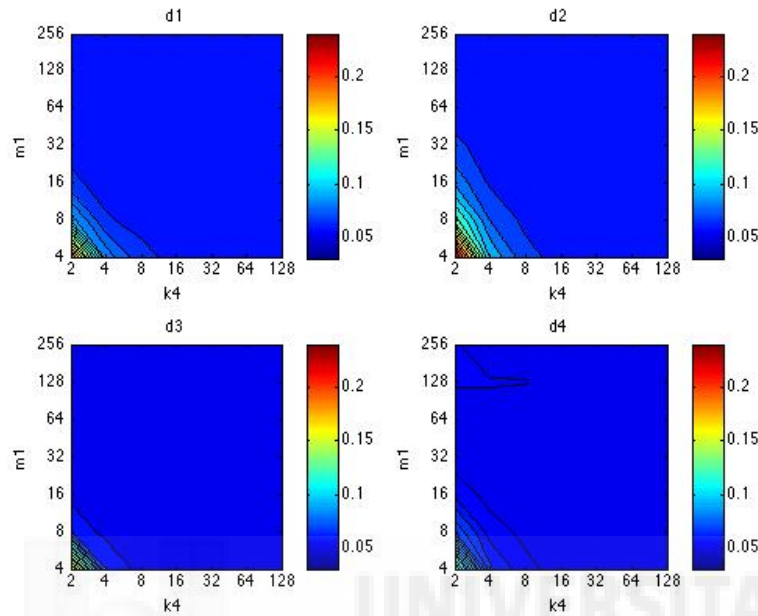
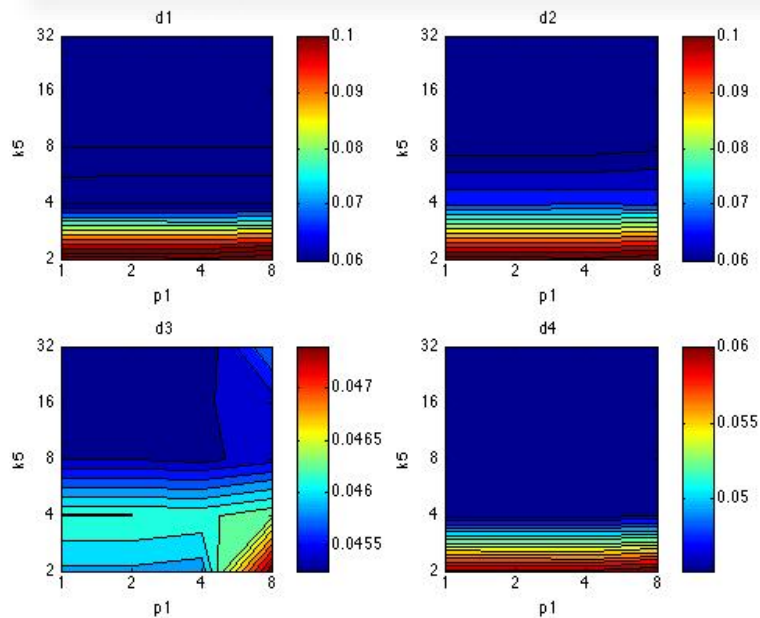


Figura 5-2.: HOG. Error de localización con el **día nublado** al variar el número de celdas horizontales  $k_3$  y el número de bins por histograma  $b_3$ .

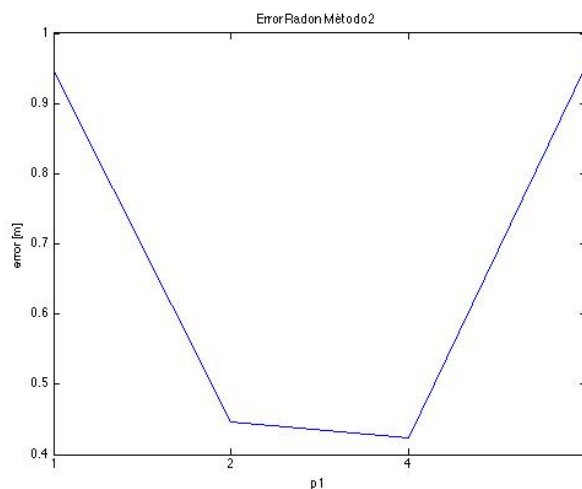


**Figura 5-3.:** gist. Error de localización con el **día nublado** al variar el número de bloques horizontales  $k_4$  y el número de filtros de Gabor  $m_1$ .



**Figura 5-4.:** RT Método 1. Error de localización con el **día nublado** al variar el número de grados entre ángulos consecutivos  $p_1$  y el número de columnas  $K_5$ .





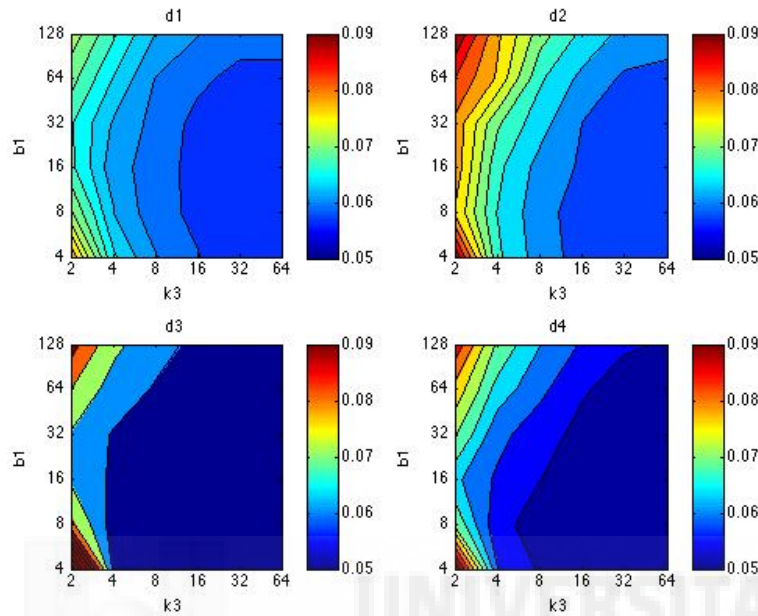
**Figura 5-5.:** RT Método 2. Error de localización con el **día nublado** al variar el número de grados entre ángulos consecutivos  $p_1$ .

encuentra al utilizar la distancia  $d_3$  con valores  $k_4 = 64$  y  $b_1 = 8$  donde se tiene un error de 0.05057m.

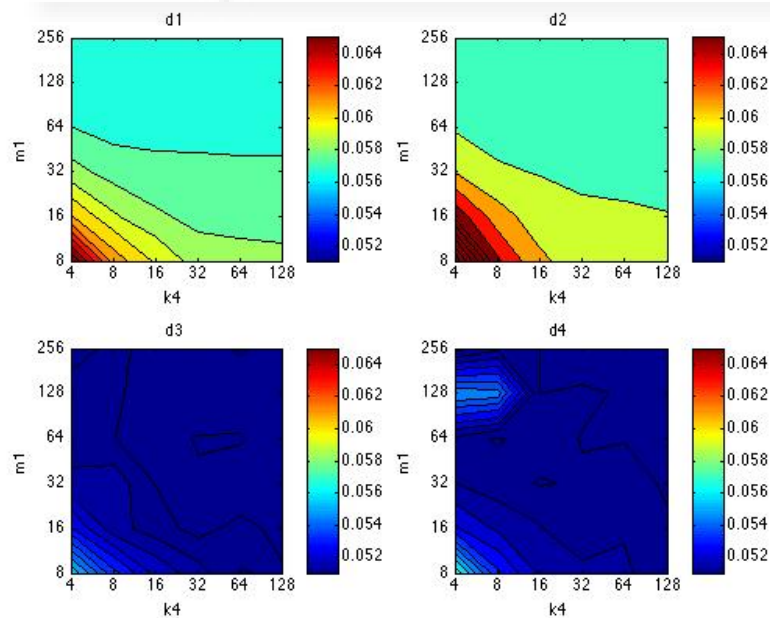
Utilizando el descriptor gist los mejores resultados se obtienen con  $d_3$  y  $d_4$ . El error se va reduciendo cuando incrementamos el valor de líneas horizontales  $k_4$  sin tener que aumentar mucho la cantidad de orientaciones de mascarás de Gabor  $m_1$ . Hay que tener en cuenta que el error aumenta mucho cuando las variables  $k_4$  y  $m_1$  son bajas. En general, obtenemos gran precisión con valores en torno a  $k_4 = 64$  y  $m_1 = 128$  pero con  $k_4 = 32$  y  $m_1 = 256$  utilizando la distancia  $d_4$  se obtiene la mejor solución con un error de 0.0501m.

Por último se analizan los resultados obtenidos con la transformada de Radon. Utilizando el primer método los mejores resultados se obtienen con la distancia  $d_3$ . El valor se va reduciendo cuando aumentamos el número de columnas  $k_5$  sin tener apenas influencia el parámetro  $p_1$  en los resultados de error. En general los resultados con mejor solución se obtienen con  $k_5=32$  y  $p_1=1$  y la solución más precisa se obtiene con  $d_3$  donde el error de localización es 0.0452. Por otro lado, al utilizar el segundo método los resultados no dependen de las cuatro distancias descritas ya que utiliza POC y por ello solo hay una gráfica. Además la solución es unidimensional porque no depende de la variable  $k_5$ . Los errores de localización son menores que el metro y los mejores resultados se consiguen con un paso de  $p_1=2$  donde el error es de 0.446m y cuando  $p_1=4$  donde el error es de 0.4234m.

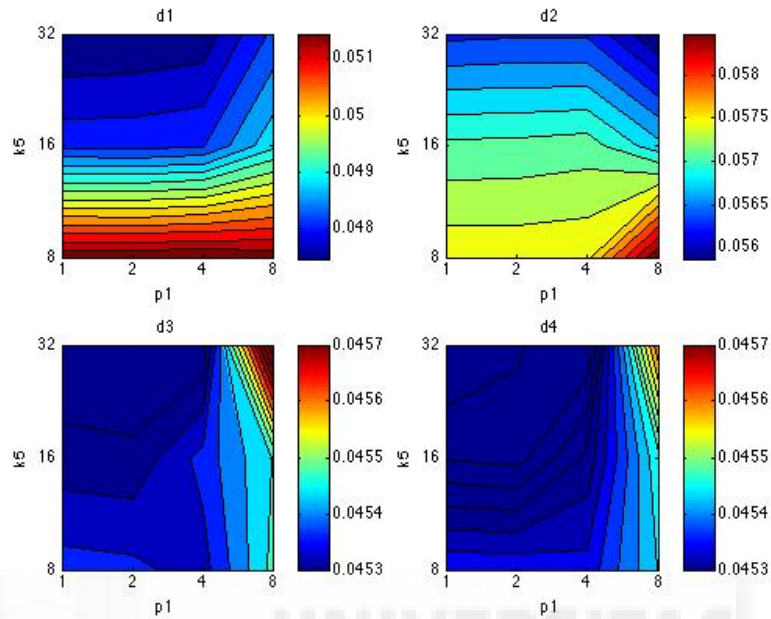
Analizando los resultados, si elegimos los valores adecuados los cuatro métodos logran sus mejores resultados con la distancia  $d_3$  consiguiendo un error en torno a 0,05 m, con la excepción de la transformada de Radon cuando se lleva a cabo por el segundo método que



**Figura 5-6.:** HOG detallado. Error de localización con el **día nublado** al variar el número de celdas horizontales  $k_3$  y el número de bins por histograma  $b_3$ .



**Figura 5-7.:** gist detallado. Error de localización con el **día nublado** al variar el número de bloques horizontales  $k_4$  y el número de filtros de Gabor  $m_1$ .



**Figura 5-8.:** RT Método 1 detallado. Error de localización con el **día nublado** al variar el número de grados entre ángulos consecutivos  $p_1$  y el número de columnas  $K_5$ .

el error mínimo es de 0.4234m. Teniendo esto en cuenta, y pese a haber cambio de mobiliario y que las escenas puedan aparecer parcialmente ocluidas, el descriptor tiene resultados remarcables cuando no hay cambio de iluminación entre el modelo y el momento en el que se hace la localización.

A continuación, y utilizando la misma base de datos, se trata de localizar el robot en un día soleado. Las figuras 5-9, 5-10, 5-11, 5-12 y 5-12 muestran los resultados. De esta manera se puede estudiar si el robot es capaz de localizarse cuando hay un cambio de luminosidad entre la situación actual y la que había al crear el modelo. Como antes, para una mayor apreciación de detalle se añaden las figuras 5-14, 5-15 y 5-16 con menos número de variables. Los datos están dados en unidades del sistema internacional y los resultados muestran un error de unos pocos metros.

Al utilizar el descriptor de la firma de Fourier en este caso, el error aumenta situándose entre los 3 y 7 metros. Pero sobre todo se puede observar un aumento en la varianza, es decir la variabilidad de los datos es grande habiendo puntos en los que se localiza bien y otros en los que se localiza muy mal. Aunque la media del error al variar el parámetro  $k_1$  es parecida, al aumentar el valor de este el error se va reduciendo teniendo los mejores resultados cuando el valor de  $k_1$  es 64 o 128. La solución con menos error se ha conseguido utilizando la distancia  $d1$  y el valor de  $k_1=128$  donde se ha localizado el robot con un error de 3,56m.

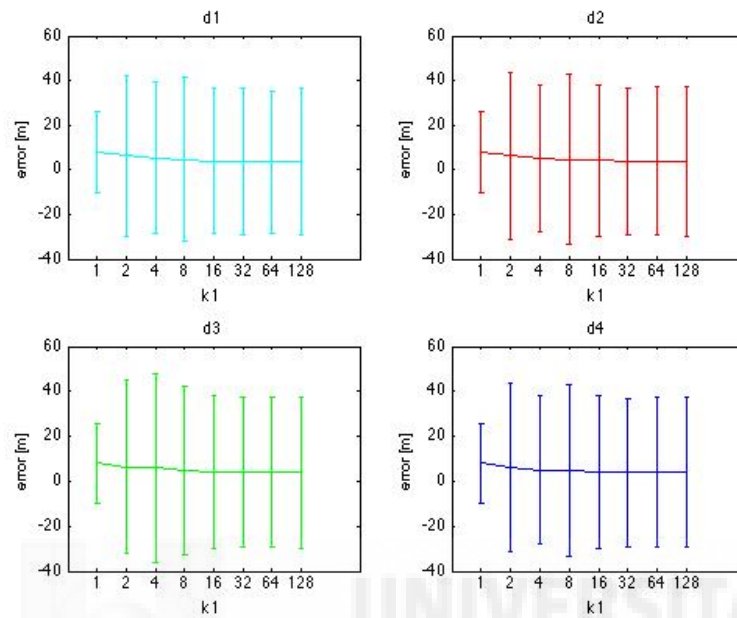


Figura 5-9.: FS. Error de localización con el **día soleado** al variar el número de columnas  $k_1$ .

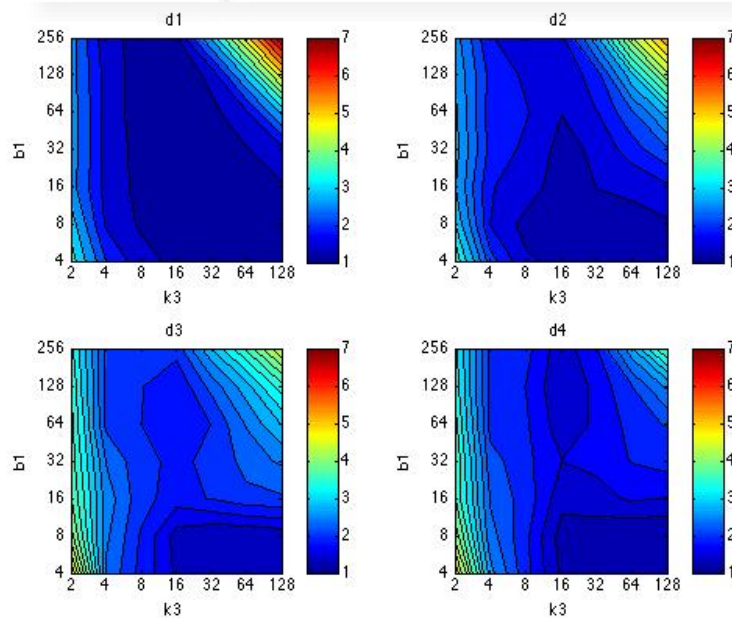


Figura 5-10.: HOG. Error de localización con el **día soleado** al variar el número de celdas horizontales  $k_3$  y el número de bins por histograma  $b_3$ .

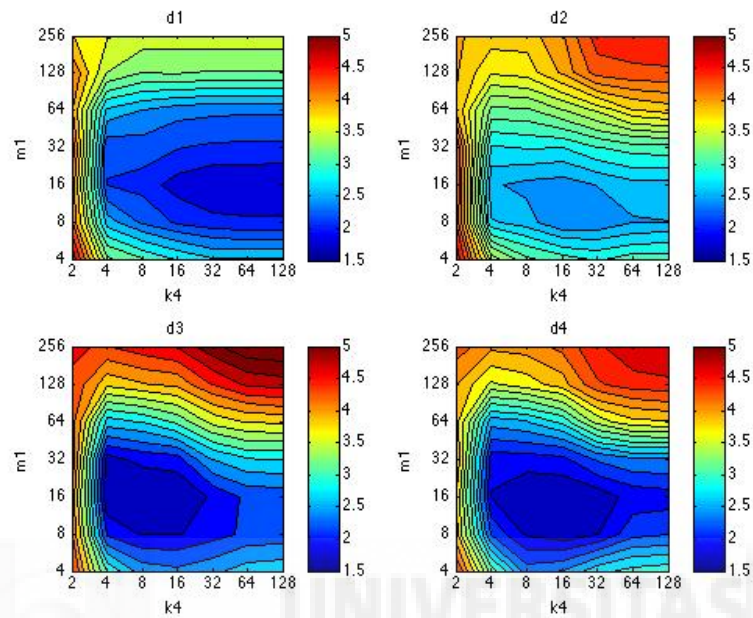


Figura 5-11.: gist. Error de localización con el **día soleado** al variar el número de bloques horizontales  $k_4$  y el número de filtros de Gabor  $m_1$ .

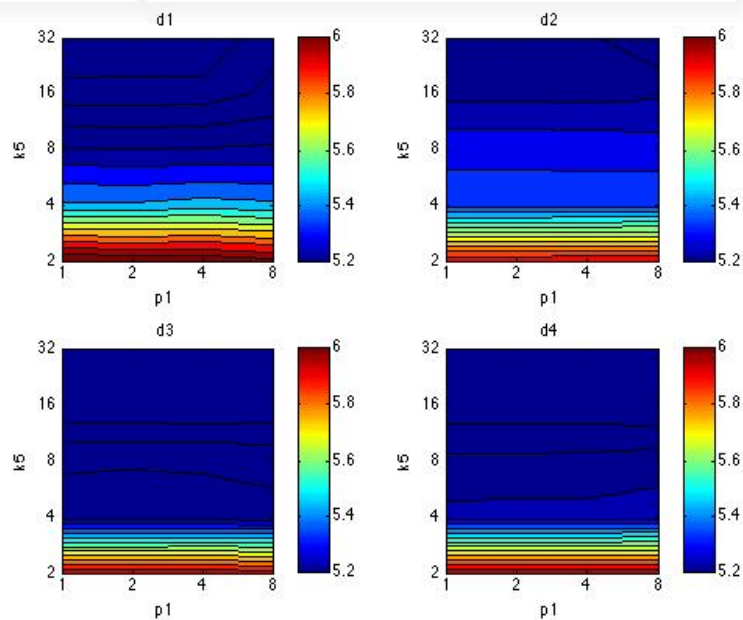
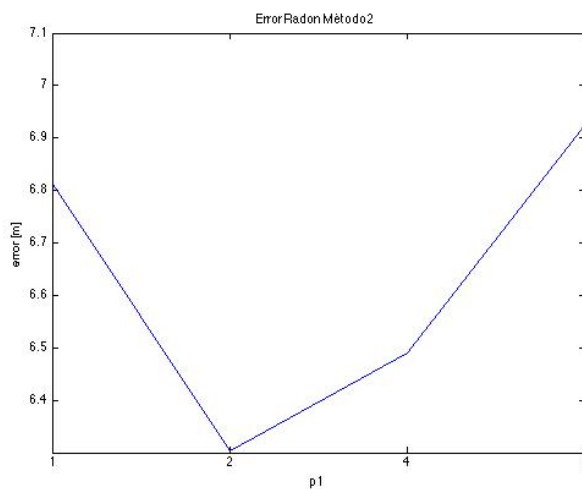


Figura 5-12.: RT Método 1. Error de localización con el **día soleado** al variar el número de grados entre ángulos consecutivos  $p_1$  y el número de columnas  $K_5$ .

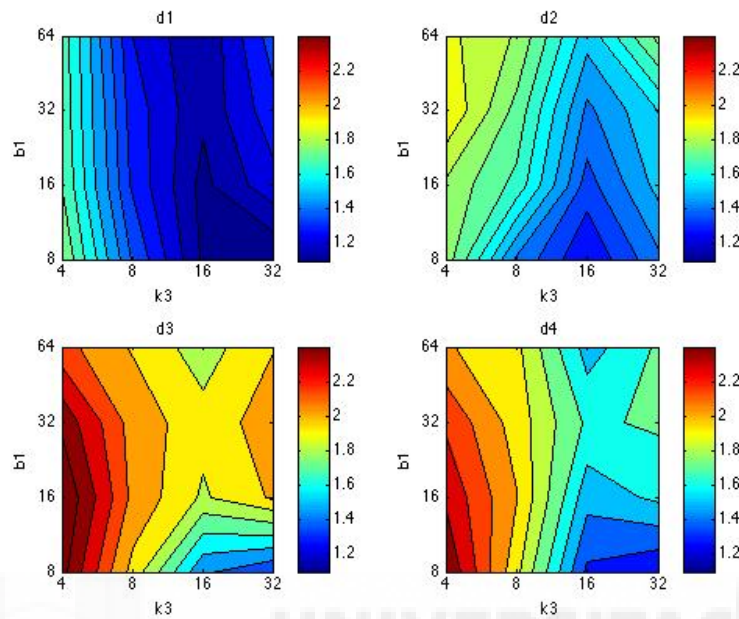


**Figura 5-13.:** RT Método 2. Error de localización con el **día soleado** al variar el número de grados entre ángulos consecutivos  $p_1$ .

Con el descriptor de apariencia global HOG se consiguen buenos resultados con cualquiera de las cuatro distancias. Con la distancia  $d1$  se puede conseguir mejores resultados si se excogen bien los parametros, pero una mala elección de estos puede suponer grandes errores de localización. Sin embargo, la respuesta al utilizar las distancias  $d3$  y  $d4$  es mucho más constante. En general, el descriptor empieza a tener malos resultados cuando el número de celdas horizontales  $k_3$  es bajo o cuando el valor de  $k_3$  y  $b_1$  son altos. De esta manera, se obtiene los mejores resultados con un valor medio-alto de la variable  $b_1$  y un valor medio de  $k_3$ . La máxima precisión se ha conseguido utilizando  $d1$  y los valores  $b_1 = 32$  y  $k_3 = 8$  donde el error de posición ha sido de 1,09m.

El comportamiento con el descriptor gist tiene un error en torno a 2m, teniendo los mejores resultados al utilizar las distancias  $d3$  y  $d4$  y los peores con la distancia  $d2$  donde el error no baja de 2,4m. De forma general, el error aumenta mucho con valores de  $k_4$  bajos o con valores de  $m_1$  altos. Así, se obtienen las mejores soluciones con valores medios de  $k_4$  y valores medio-bajos de  $m_1$ . El error más bajo se ha detectado al utilizar la distancia  $d3$  y los valores  $k_4=m_1=16$  valores con los que el error ha sido 1,69m.

Por último se evalúa los resultados ofrecidos por Radon. En este caso Radon tiene unos resultados bastante peores ya que el error de posición está entre 5m y 6m con el método 1 y es mayor que 6m con el método 2. Con el primer método se obtienen los mejores resultados con la distancia  $d1$ . Como en el caso del día nublado el error no varía mucho con el cambio del parámetro  $p_1$  y su dependencia está más ligada a  $k_5$ . Los mejores resultado se obtienen con valores altos de  $k_5$  y la mejor solución cuando  $k_5=32$  y  $p_1=1$ , en este caso con la distancia

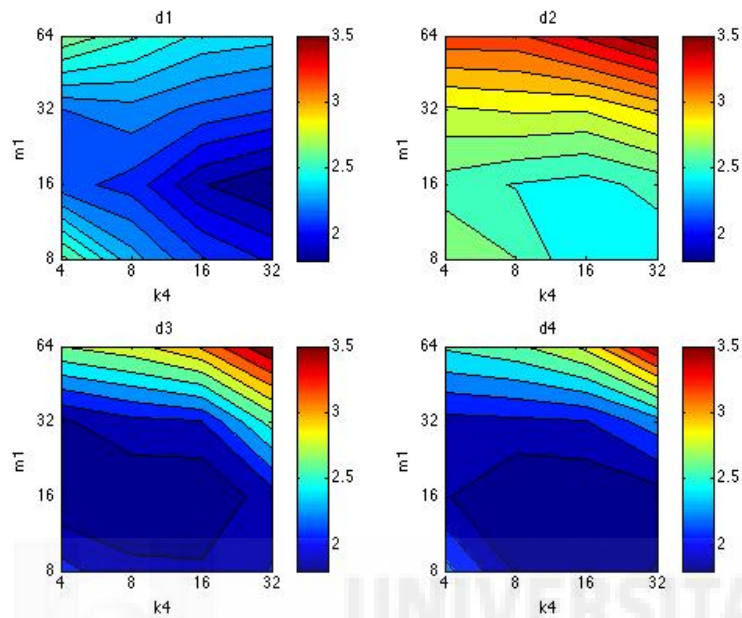


**Figura 5-14.:** HOG detallado. Error de localización con el **día soleado** al variar el número de celdas horizontales  $k_3$  y el número de bins por histograma  $b_3$ .

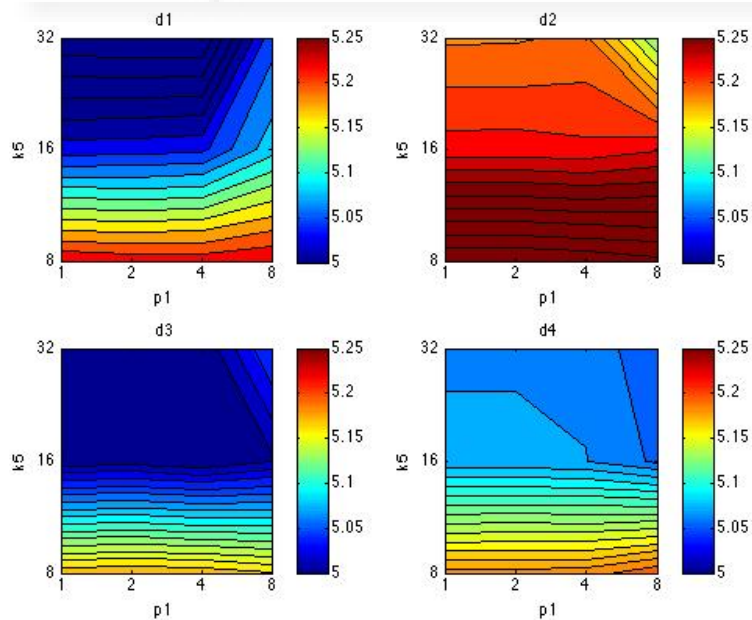
*d1* donde el error de localización es 4.9321m. En el caso del segundo método el resultado empeora con valores bajos o altos de la variable  $p_1$  teniendo la mejor solución cuando  $p_1=2$  en la que el error de localización es 6.3046m.

Como conclusión se obtiene que al haber un cambio de iluminación entre el momento que se realiza la base de datos y el momento en el que se intenta localizar al robot el error aumenta pero este no es desorbitado ya que si elegimos adecuadamente los parámetros puede suponer un error de 1,5-2m que, teniendo en cuenta que la escena no solo ha sufrido un cambio de iluminación si no que también puede haber habido cambio en el mobiliario, oclusiones..., se trata de un error a tener en cuenta pero no es extremo. También se puede destacar que los resultados con el descriptor HOG son mejores, es decir, es el descriptor con el que obtenemos los resultados de error más bajos; pero una mala elección de los parámetros de entrada puede suponer un error bastante considerable que ronda los 7m sin embargo con descriptores como gist, aunque los resultados no sean tan buenos, el error máximo no supera los 5.5m. Por otro lado hay que destacar que el descriptor Transformada de Radon no ha funcionado bien en este caso en ninguno de sus dos métodos ya que sus soluciones mínimas son 4.9321m y 6.3046m respectivamente, resultados bastante alejados a los obtenidos por los otros tres métodos.

Por último, se realiza otra prueba con cambio de iluminación. En esta ocasión se mantiene

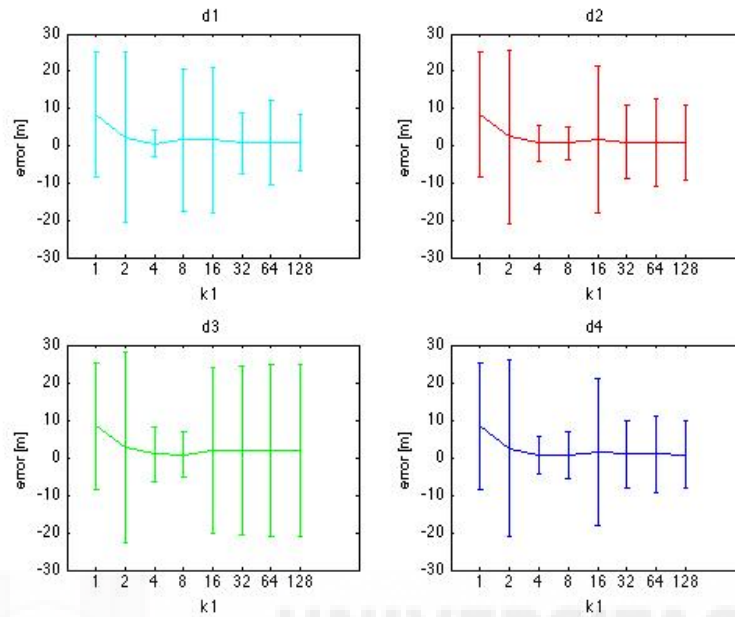


**Figura 5-15.:** gist detallado. Error de localización con el **día soleado** al variar el número de bloques horizontales  $k_4$  y el número de filtros de Gabor  $m_1$ .



**Figura 5-16.:** RT Método 1 detallado. Error de localización con el **día soleado** al variar el número de grados entre ángulos consecutivos  $p_1$  y el número de columnas  $K_5$ .



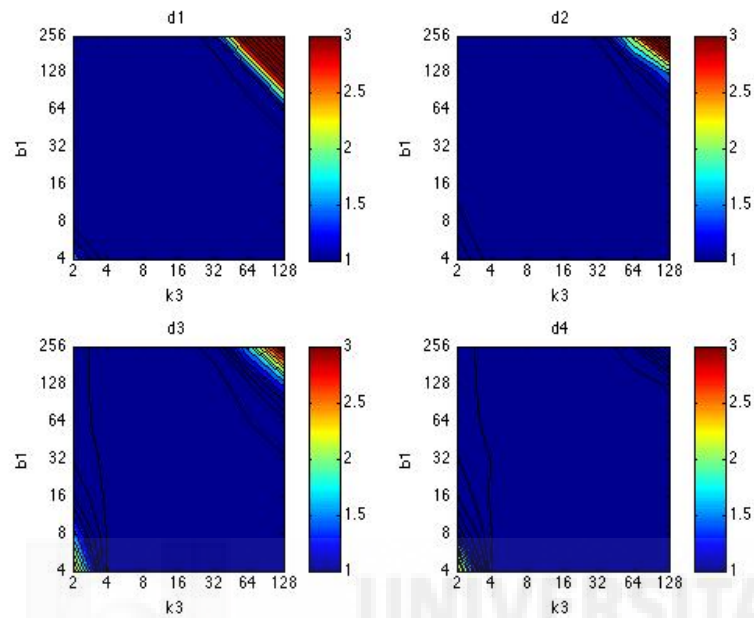


**Figura 5-17.:** FS. Error de localización cuando ha **anochecido** al variar el número de columnas  $k_1$ .

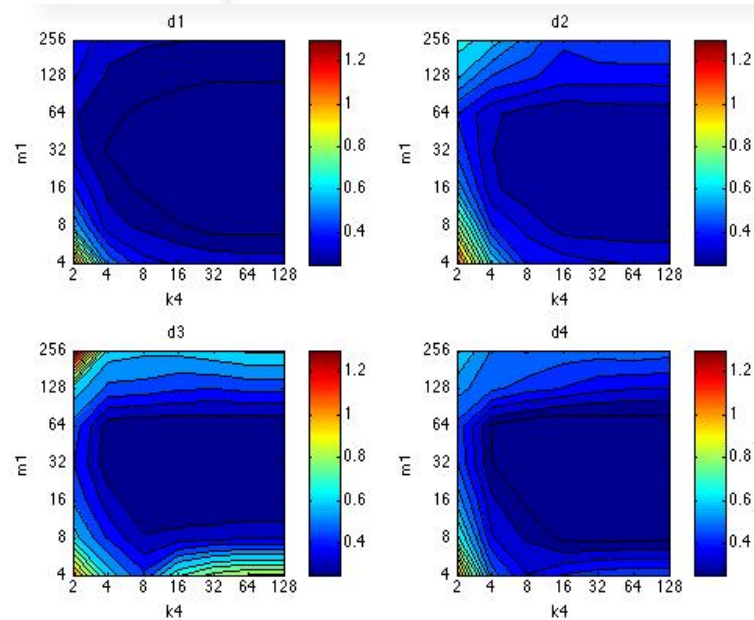
la base de datos que se obtuvo en un día nublado pero se trata de localizar un robot de noche. Las figuras 5-17, 5-18, 5-19, 5-20 y 5-21 muestran graficado el resultado del error medio al realizar la localización e ir variando los parámetros correspondientes. También se han incluido las figuras 5-22, 5-23 y 5-24 para un mayor nivel de detalle ya que incluyen menos parámetros. Los resultados están expresados en metros, unidades del sistema internacional.

Los resultados de localización al utilizar la firma de Fourier ofrecen un error entre los 0,6 y 2,5m (exceptuando el caso en el que  $k_1=1$  donde el error está en torno a los 8m) y se obtiene la mejor solución con los tipos de distancias  $d1$  y  $d2$ . Aunque se tiene mejores resultados que cuando se realizó la localización con el día soleado, se tiene el mismo problema; la varianza de los resultados es muy alta y solo se obtiene una varianza que podríamos admitir como aceptable con valores medios de  $k_1$ . La mejor solución se obtiene con la distancia  $d1$  y un número de columnas  $k_1 = 4$  donde se obtiene un error de 0,597m.

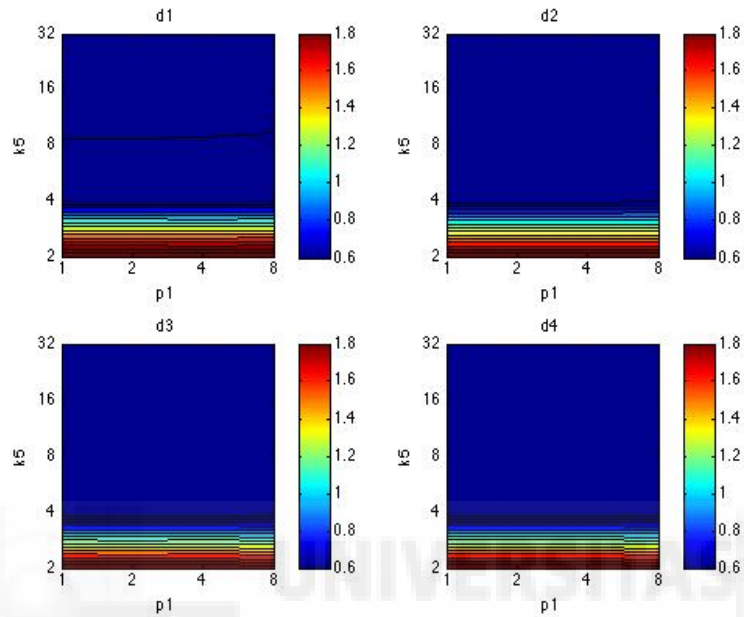
El comportamiento ante el descriptor global HOG trae un error de posición medio entre 0.2 y 0.6 m y los mejores resultados se obtienen con  $d1$ . El error aumenta cuando los valores  $k_3$  y  $b_1$  son extremos, es decir con valores bajos o altos de estos parámetros, destacando que con valores altos de  $k_3$  y  $b_1$  el error de localización aumenta mucho obteniendo errores de 4-6m. Por ello, al utilizar el descriptor HOG es mejor tener unos valores medios de  $k_3$  y  $b_1$ . El menor error se ha obtenido al utilizar la distancia  $d1$  y los parámetros  $k_3 = 32$  y  $b_1 = 16$



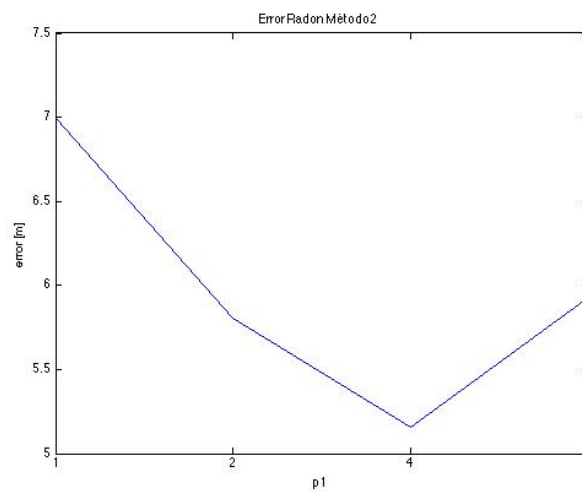
**Figura 5-18.:** HOG. Error de localización cuando ha **anochecido** al variar el número de celdas horizontales  $k_3$  y el número de bins por histograma  $b_3$ .



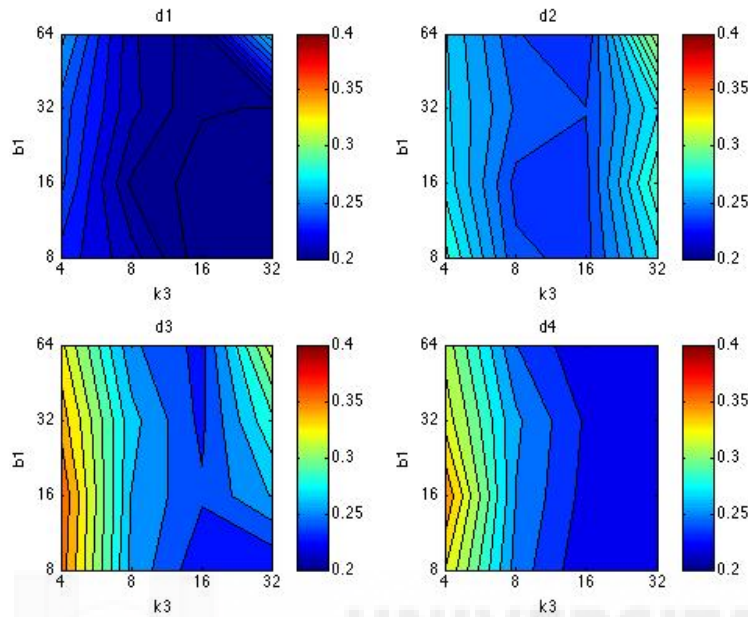
**Figura 5-19.:** gist. Error de localización cuando ha **anochecido** al variar el número de bloques horizontales  $k_4$  y el número de filtros de Gabor  $m_1$ .



**Figura 5-20.:** RT Método 1. Error de localización cuando ha **anochecido** al variar el número de grados entre ángulos consecutivos  $p_1$  y el número de columnas  $K_5$ .



**Figura 5-21.:** RT Método 2. Error de localización cuando ha **anochecido** al variar el número de grados entre ángulos consecutivos  $p_1$ .



**Figura 5-22.:** HOG detallado. Error de localización cuando ha **anochecido** al variar el número de celdas horizontales  $k_3$  y el número de bins por histograma  $b_3$ .

donde la solución tiene un error de 0.1892m.

Al utilizar como descriptor el método gist se obtiene un error de localización medio entre 0,2 y 1 m. El error aumenta su valor si se tiene como parámetros valores de  $k_4$  o  $m_1$  bajos sin que este resultado baje mucho cuando los parámetros aumentan, es por ello es por ello que las mejores soluciones se consiguen al tener valores intermedios en  $k_4$  y  $m_1$ . El error más bajo se ha obtenido con  $d3$  y los parámetros  $k_4 = 64$  y  $m_1 = 16$  con los que se obtiene un error de 0,209m. Cabe destacar que el error máximo que se obtuvo durante las pruebas utilizando el descriptor gist y teniendo condiciones de luminosidad exterior bajas es de 1,37m obtenido con la distancia  $d3$  y los parámetros  $k_4=2$  y  $m_1=256$ , un error bastante bajo si se compara con el máximo error al utilizar otros descriptores.

Para finalizar, con el descriptor Radon se tiene un error de localización entre 0.4 y 1.8m. El error aumenta con valores bajos de  $k_5$  y tiene poca dependencia de  $p_1$ . Los mejores resultados se obtienen con la distancia  $d3$ . Con los parámetros  $k_5 = 32$  y  $p_1 = 1$  se obtienen los mejores resultados, en concreto cuando la distancia es  $d3$  el error de localización es 0.3281m. Sin embargo el método 2 sigue sin ser eficaz ya que el error medio es superior a los 5m, el mejor resultado se obtiene con  $p_1=4$  que se obtiene un error de 5.1551m, pero sigue sin ser un error aceptable para un localizador.

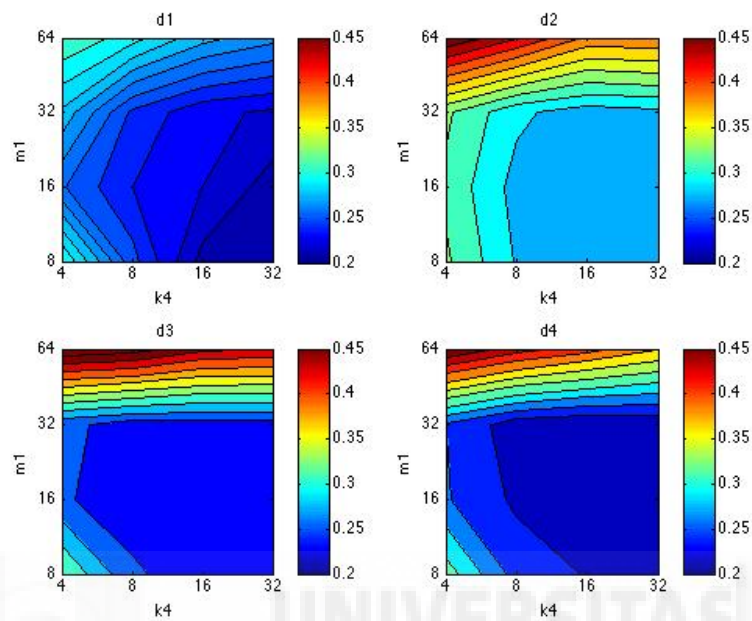


Figura 5-23.: gist detallado. Error de localización cuando ha **anochecido** al variar el número de bloques horizontales  $k_4$  y el número de filtros de Gabor  $m_1$ .

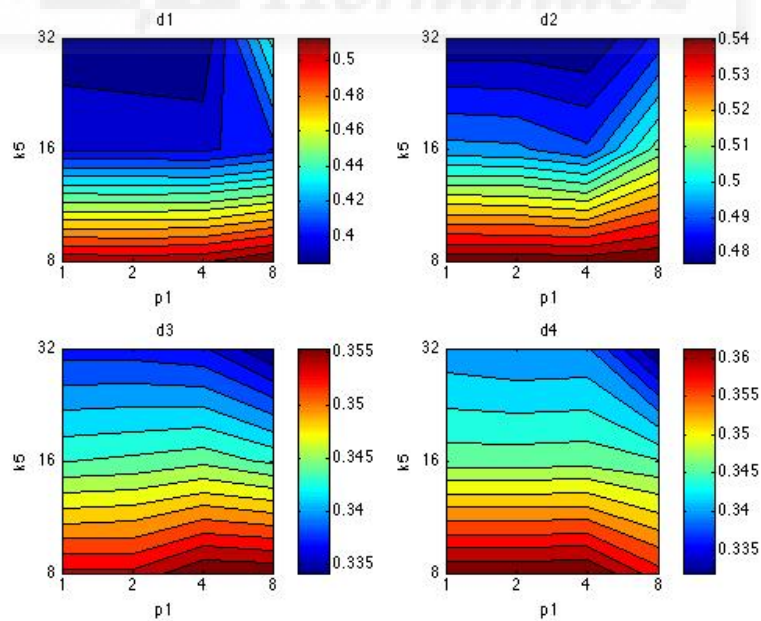


Figura 5-24.: RT Método 1 detallado. Error de localización cuando ha **anochecido** al variar el número de grados entre ángulos consecutivos  $p_1$  y el número de columnas  $K_5$ .

Una vez obtenidos los resultados se observa que el error mínimo de localización puede estar en torno al metro (exceptuando el segundo método de la transformada de Radon), este error no es muy alto si se tiene en cuenta que las escenas sufren muchos cambios, el más importante el cambio de iluminación y también otros como cambio de muebles, oclusiones producidas por personas... En general las distancias que mejor han respondido en este caso son  $d1$  y  $d3$  y como en los otros dos casos el descriptor que mejores resultados ofrece es HOG ya que da como solución aquella con el error más bajo pero se debe tener en cuenta que con unos parámetros no adecuados el error aumenta situándose en 4-6 metros mientras que con los descriptores Fourier, gist y el primer método de Radon ofrecen errores algo más altos que los de HOG pero su error no supera los 2,5m en el caso de la firma de Fourier, los 1,4m en el caso del descriptor gist ni los 2.2m en el caso de Radon.

Pese a lo que se podía pensar *a priori*, es reseñable que los resultados de localización obtenidos con luminosidad de un día soleado son peores que los obtenidos en un día de noche. Este resultado puede ser debido a que las imágenes empleadas en la base de datos fueron tomadas en una estancia en las que la mayoría de las paredes son de cristal, esto puede provocar alguna anomalía a la hora de trabajar con la luz. Pero teniendo en cuenta el reto que suponía el poder localizar un robot empleando únicamente una cámara omnidireccional cuando hay cambios de iluminación entre el día que se creó la base de datos y el día en el que se desea localizar el robot, se ha llegado a unos resultados remarcablemente buenos.

### 5.3. Coste computacional

Aparte de tener un error reducido es importante también que el tiempo de ejecución de esa localización no sea elevado. Por ello, también es importante estudiar el coste computacional del proceso y evaluar si la tarea de localización puede realizarse en tiempo real. Las figuras 5-25, 5-26, 5-27, 5-28 y 5-29 muestran el tiempo de procesamiento por imagen necesario para localizar el robot en una posición. Los resultados está expresados en segundos.

Para el cálculo del tiempo de procesamiento por imagen se ha realizado la suma de tiempo en pasarle el descriptor, tiempo que tarda en encontrar el vecino más cercano, y tiempo en pasar a panorámica (este último sumando no se tiene en cuenta en la Transformada Radon ya que no es necesario pasar la imagen a panorámica). Para calcular el tiempo para pasar a panorámica una imagen se ha realizado un experimento en el que se calcula la media de pasar 50 imágenes a panorámica y se repitió el experimento 8 veces. Finalmente se concluyó que el tiempo para pasar una imagen a panorámica son 2.44s, el resto de tiempos se han ido calculando recogiendo los tiempos conforme se realizaban los experimentos..

Si se utiliza la firma de Fourier se consiguen los resultados más rápidos. Contando que el

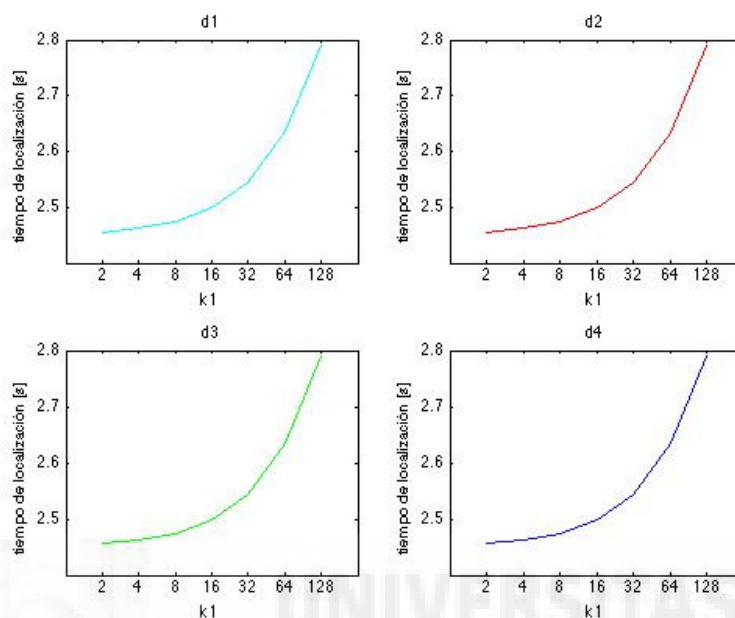


Figura 5-25.: FS. Tiempo de localización por imagen al variar el número de columnas  $k_1$ .

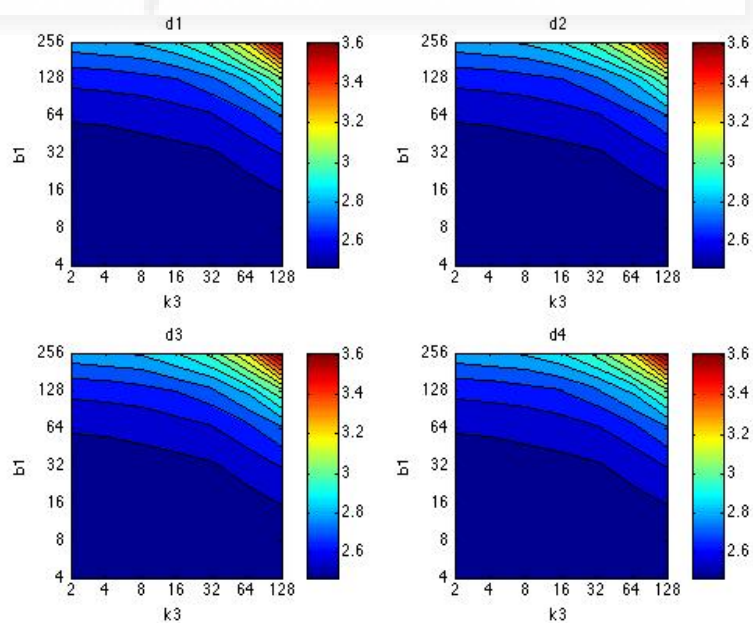
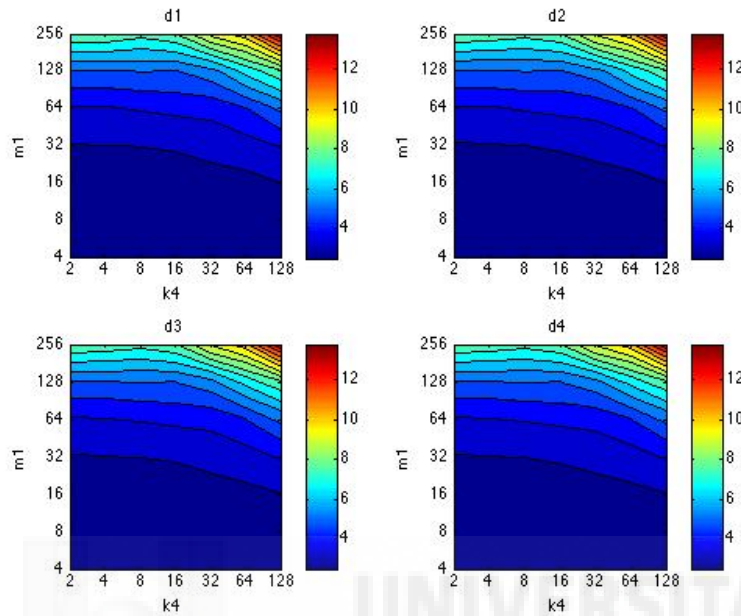
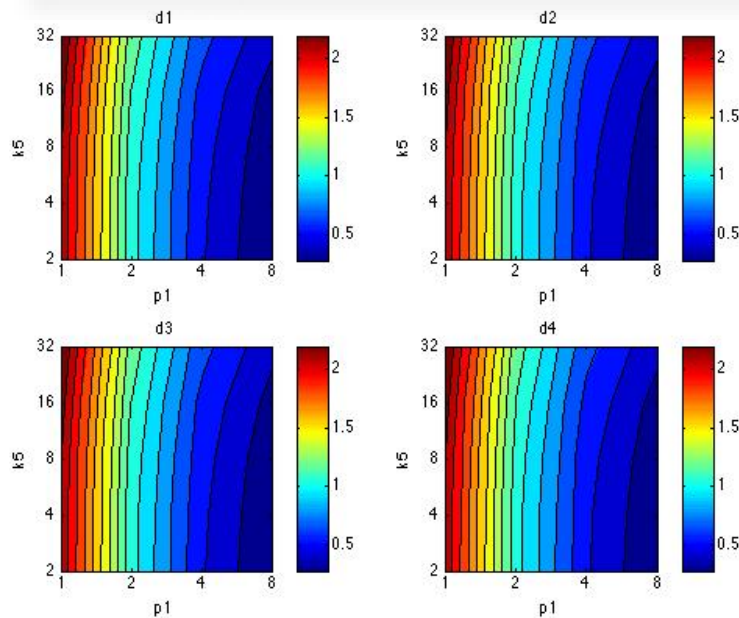


Figura 5-26.: HOG. Tiempo de localización por imagen al variar el número de celdas horizontales  $k_3$  y el número de bins por histograma  $b_3$ .

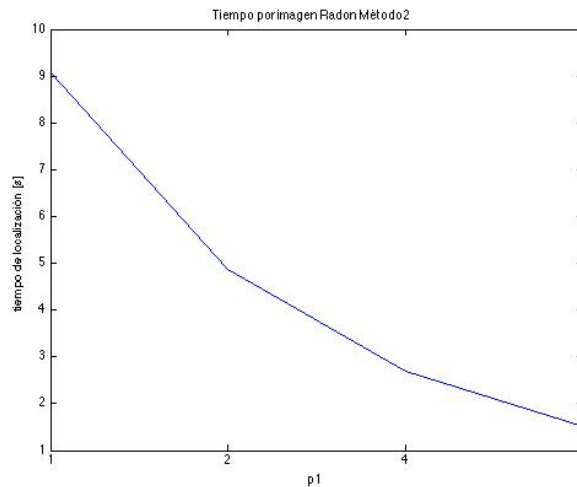


**Figura 5-27.:** gist. Tiempo de localización por imagen al variar el número de bloques horizontales  $k_4$  y el número de filtros de Gabor  $m_1$ .



**Figura 5-28.:** RT Método 1. Tiempo de localización por imagen al variar el número de grados entre ángulos consecutivos  $p_1$  y el número de columnas  $K_5$ .





**Figura 5-29.:** RT Método 2. Tiempo de localización por imagen al variar el número de grados entre ángulos consecutivos  $p_1$ .

tiempo para pasar la imagen a panorámica son 2.44s, observa que el resto de procesamiento está por debajo de 0.2s para la mayoría de casos y únicamente cuando aumentamos mucho el número de columnas  $k_1$  es cuando tenemos un procesamiento algo más lento. Se observa también un crecimiento exponencial en el tiempo al aumentar  $k_1$  lo que hace pensar que si aumentamos todavía más el número de columnas crecerá mucho más el tiempo empleado en la localización.

El tiempo de procesamiento al utilizar el descriptor global HOG es significativamente superior al de la firma de Fourier. En la mayoría de experimentos el resultado está por debajo de los 3s pero con valores de parámetros altos el resultado puede llegar a los 3.6s con valores altos de  $k_3$  y  $b_1$ . Al aumentar cualquiera de los parámetros el coste computacional sube,  $b_1$  es el parámetro que más influencia tiene en los resultados por tanto es preferible no aumentar mucho este valor. Sin embargo, aunque influye en el proceso, la dependencia del tiempo con el parámetro  $k_3$  es menor.

Si utilizamos el descriptor gist el coste computacional se eleva mucho y los resultados muestran unos tiempos de procesamiento por imagen de 4s a 10s obteniendo incluso tiempos de 13s cuando los parámetros son muy elevados. El descriptor gist no ofrece una respuesta muy rápida, además está muy influenciado por el valor de la variable  $textit{m}_1$  por ello, es muy recomendable mantener configuraciones con un valor bajo de  $textit{m}_1$  para tener un coste computacional aceptable.

Por último se estudia los resultados obtenidos con el descriptor Transformada de Radon. Con el método 1 los tiempos no son altos y se obtienen valores que varían entre los 0.3 y 2

segundos aproximadamente. Se observa una clara dependencia del paso  $p_1$  y una dependencia menor del número de columnas  $k_5$ . Cuando los grados entre ángulos consecutivos es  $p_1=1$  los valores de tiempo son mayores a 2.1s obteniendo el mayor tiempo cuando  $k_5=32$  y el tiempo de localización por imagen 2.312s. Cuando el paso entre ángulo consecutivos es  $p_1=8$  los resultados están por debajo de los 0.45s y el mejor resultado se obtiene cuando  $k_5=2$  y la distancia  $d1$  que el tiempo de localización es 0.2875s.

Cuando se emplea el método 2 y POC el coste computacional solo depende de  $p_1$ . El resultado es máximo con  $p_1=1$  donde el tiempo de localización por imagen es 9.094s y mínimo cuando  $p_1=8$  cuando es 1.4988s.

El método de la transformada de Radon puede ser más rápido que los otros métodos ya que un parámetro como el de pasar las imágenes a panorámica no se suma. Sin embargo tiene mucha dependencia de el paso entre ángulos y un valor bajo de  $p_1$  tiene como consecuencia tener valores de tiempo altos.

El coste computacional no depende de las condiciones de iluminación ya que las operaciones a realizar son las mismas independientemente de las condiciones de iluminación. Por ello solo se va a realizar la comparativa de tiempos en uno de los casos.

Los resultados de estos tiempos no son absolutos ya que dependerán de la máquina con la que se procese los datos. Los resultados se pueden utilizar comparativamente entre los diferentes métodos ya que se han realizado con el mismo ordenador.

## 6 Estimación de la orientación del robot

El objetivo del trabajo no es solo evaluar la facilidad del robot para localizar su posición al variar los parámetros de los descriptores, también se considera que para realizar una navegación adecuada es necesario conocer la orientación del robot y por ello se ha realizado un estudio del comportamiento de los descriptores de apariencia global para estimar la orientación de móvil. Para estudiar este comportamiento se ha elegido zonas en las que hay un cambio grande de ángulo en pocos metros como explicaremos en la siguiente sección. El capítulo se va a dividir en dos secciones; en la primera se va explicar el método empleado para elegir las zonas en las que se ha calculado el error de orientación y como se ha calculado dicho error. En el siguiente apartado se expone los resultados obtenido con los cuatro métodos empleados y las tres condiciones de ambiente (Nublado, soleado y de noche).

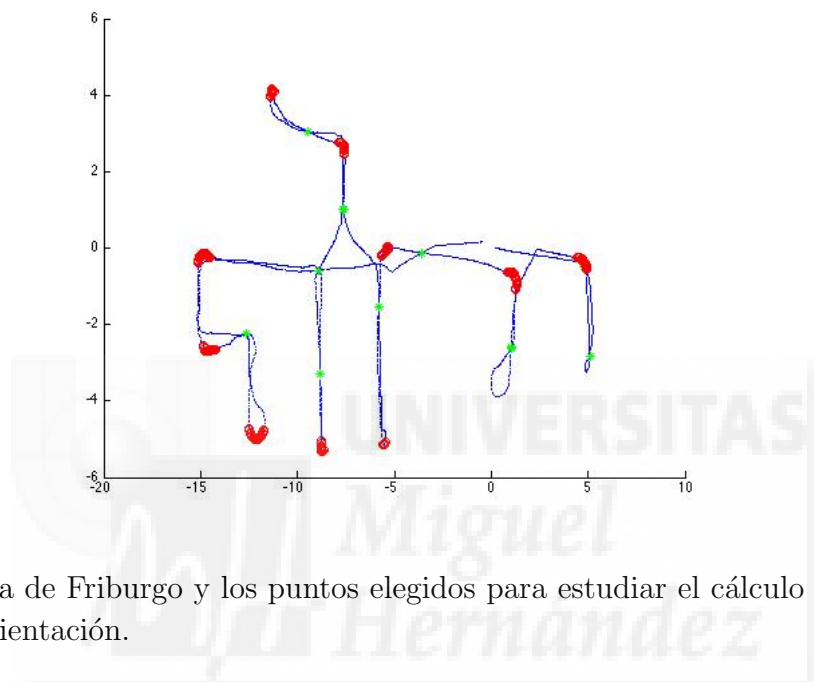
### 6.1. Método

La idea principal para calcular el error de orientación en una aplicación real podría ser la siguiente: primero se debería buscar en una base de datos la imagen más cercana por posición, después calcular el ángulo de giro de cada una de las imágenes y calcular el error de orientación cometido.

Para calcular como se comporta el error en la orientación del robot no tiene sentido que se haga un estudio con toda la batería de imágenes que se dispone ya que con imágenes consecutivas que se detectarían como cercanas la diferencia de ángulo resultante será casi nula y no será posible detectar si es debido a que el descriptor no ha detectado el ángulo o es debido a que realmente el error de orientación entre ambas imágenes es muy bajo.

Para solucionar el problema expuesto se decidió elegir 20 zonas por las que el robot pasa con diferente orientación. Estas zonas se dividen en dos grupos. 10 parejas de imágenes son de lugares en las que el robot realiza un giro. Por otro lado las otras 10 parejas de imágenes son zonas por las que el robot móvil pasa con diferente sentido, es decir en un principio ha pasado por dicha zona y al tiempo vuelve a pasar dirigiendo se a otro lugar con otra orientación. En la figura **6-1** podemos observar el mapa del laboratorio de Friburgo. De color azul encontramos todos los puntos en los que el robot ha realizado una captura de imagen. En color verde encontramos los momentos elegidos como aquellos en los que el

robot ha pasado por el mismo sitio con otra dirección o sentido mientras que de color rojo encontramos parejas de imágenes que han sufrido un giro y están en una posición similar con otra orientación.

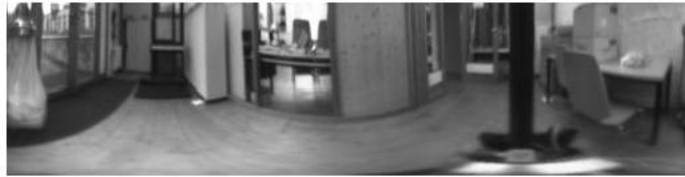


**Figura 6-1:** Mapa de Friburgo y los puntos elegidos para estudiar el cálculo del cambio en la orientación.

En las figuras **6-2**, **6-3**, **6-4** y **6-5** podemos observar cuatro ejemplos de imágenes en los que se ha calculado el error de orientación. Como se puede comprobar se trata del mismo punto o misma zona y el robot tiene una orientación distinta. En todas las figuras la imagen (a) es la imagen del modelo, la imagen (b) una imagen capturada un día nublado, la imagen (c) una imagen capturada un día soleado y por último la imagen (d) una imagen capturada de noche. Como se puede comprobar la iluminación es diferente en las imágenes, factor que afectará a que el cálculo de la orientación. También es posible observar como otros cambios producidos por la actividad humana pueden afectar al resultado, por ejemplo en la figura **6-4** hay cambio en el mobiliario.

El objetivo de este capítulo es doble. Por un lado se desea comprobar si los descriptores son capaces de detectar que entre las dos imágenes ha habido un cambio de orientación. Por otro lado se desea saber si, habiendo detectado un cambio de orientación, el ángulo de rotación detectado es correcto o existe algún error de cálculo.

Como se puede comprobar en las imágenes, la estructura del robot afecta en la captación de las imágenes debido a que el soporte de la cámara se ve en ellas. Por ello, se han realizado unas pruebas en las que se detecta la cantidad de veces que los detectores han



(a) Imagen Cloudy de referencia



(b) Imagen Cloudy



(c) Imagen Sunny



(d) Imagen Night

**Figura 6-2:** Ejemplo de un cruce en el que se ha calculado cambio de orientación



(a) Imagen Cloudy de referencia



(b) Imagen Cloudy



(c) Imagen Sunny



(d) Imagen Night

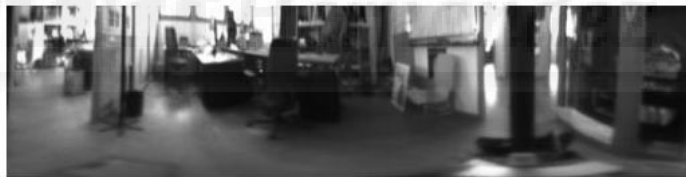
**Figura 6-3:** Ejemplo de un cruce en el que se ha calculado cambio de orientación



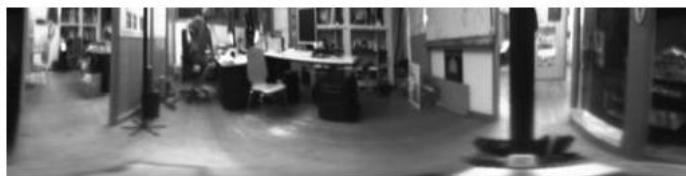
(a) Imagen Cloudy de referencia



(b) Imagen Cloudy



(c) Imagen Sunny



(d) Imagen Night

**Figura 6-4:** Ejemplo de un giro en el que se ha calculado cambio de orientación



(a) Imagen Cloudy de referencia



(b) Imagen Cloudy



(c) Imagen Sunny



(d) Imagen Night

**Figura 6-5:** Ejemplo de un giro en el que se ha calculado cambio de orientación



captado una diferencia de orientación entre imágenes ya que el soporte puede ser una estructura muy predominante y como resultado el descriptor optiene que no hay un cambio de orientación. Estos resultados se muestran en una grafica en la que se expresa el porcentaje de acierto, es decir el porcentaje de veces en las que se detecta un cambio de orientación.

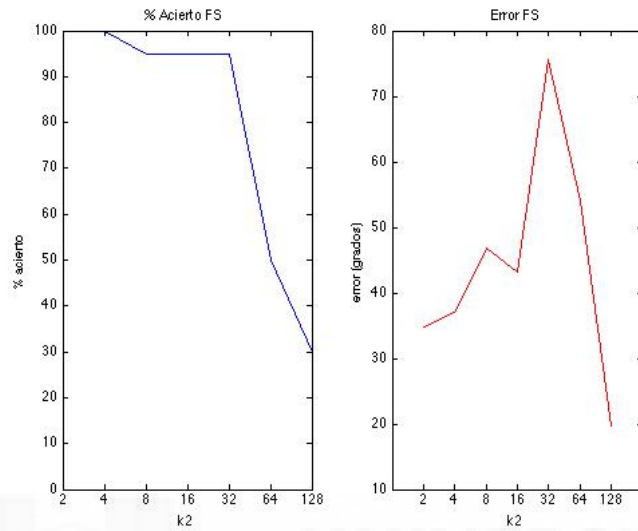
Una vez comprobado si se ha detectado que hay o no cambio de orientación se calcula el ángulo con la información que ofrecen los detectores y con ello calculamos el error entre la medida real y esta. Los resultados de este cálculo de error se muestran en otra gráfica.

## 6.2. Resultado y discusión

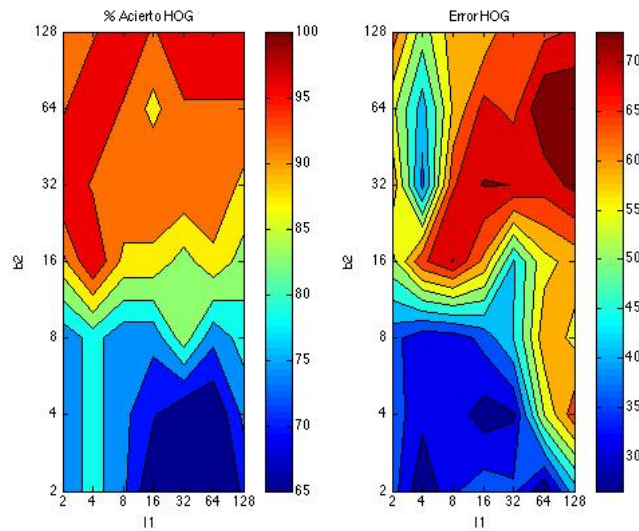
Las figuras 6-6, 6-7, 6-8, 6-9 y 6-10 muestran por un lado una gráfica con el porcentaje de acierto o detección de cambio de orientación entre dos imágenes y por otro lado una gráfica con el error entre el ángulo calculado con los descriptores y el real. Estas imágenes muestran los resultados de el cálculo de orientación con imágenes tomadas un día nublado, mismo escenario que las imágenes con las que se creó el modelo. Los resultados son variantes, encontramos que los descriptores Firma de Fourier, HOG y gist ofrecen un error de cálculo no muy bajo que puede variar entre los 5 y 70 grados dependiendo del descriptor y los parámetros sin embargo la tasa de detección de que ha habido un cambio de orientación no es siempre del 100 %, sino que está entre el 80 % y 90 % de detecciones de ángulo. Por otro lado, el descriptor Radon tiene una porcentaje de detección del 100 % pero el error medio es muy alto.

El comportamiento con el descriptor Firma de Fourier no es del todo el deseado. Con valores altos de  $k_2$  el porcentaje de detección de ángulo baja teniendo un porcentaje del 50 % con  $k_2=64$  y del 30 % con  $k_2=128$ , sin embargo con valores bajos de  $k_2$  el porcentaje de detección de ángulo si que es del 95-100 %. Por parte del error en el cálculo del ángulo se obtiene un error medio alto con valores intermedios de  $k_2$  pero con  $k_2=2$  se obtiene un error medio de 35 grado aproximadamente y con el la variable  $k_2=128$  un error de orientación de 20 grados.

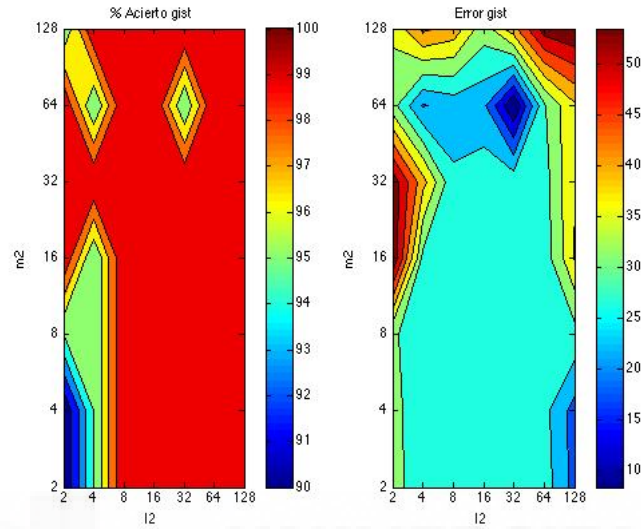
Utilizando el descriptor HOG no se obtiene porcentajes de detección de ángulos tan bajos como con la Firma de Fourier. En este caso el parámetro  $b_2$  influye más a la hora de que se detecte ángulo teniendo que con valores altos de  $b_2$  el porcentaje de detección es del 90-100 % y con valores de  $b_2$  bajos como podría ser 2, 4 u 8 se obtiene un porcentaje de 65-80 %. Por parte del error en la detección de los ángulos, HOG ofrece los mejores resultados con valores bajos de ambos parámetros obteniendo el mejor resultado con  $b_2= 2$  y  $l_1=4$  donde el error de orientación es 30 grados. Sin embargo con valores altos de los parámetros ( $b_2$  con valores de 32, 64 o 128 y  $l_1$  64 o 128) el error medio en el cálculo de orientación es de 60-70 grados.



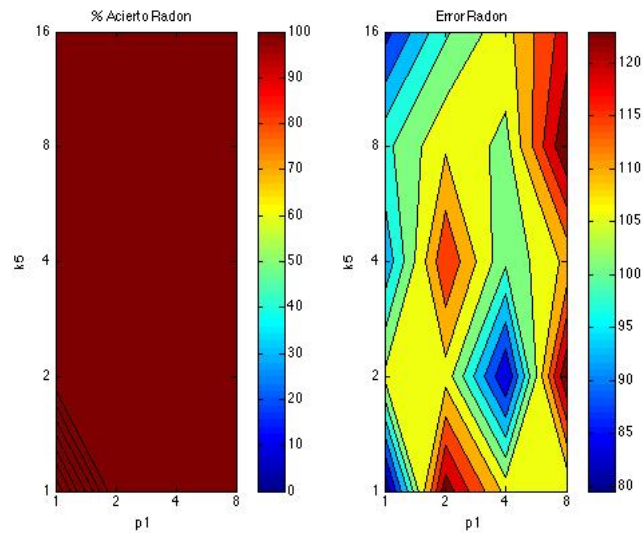
**Figura 6-6:** FS. Error de orientación con el **día nublado** al variar el número de columnas  $k_2$ .



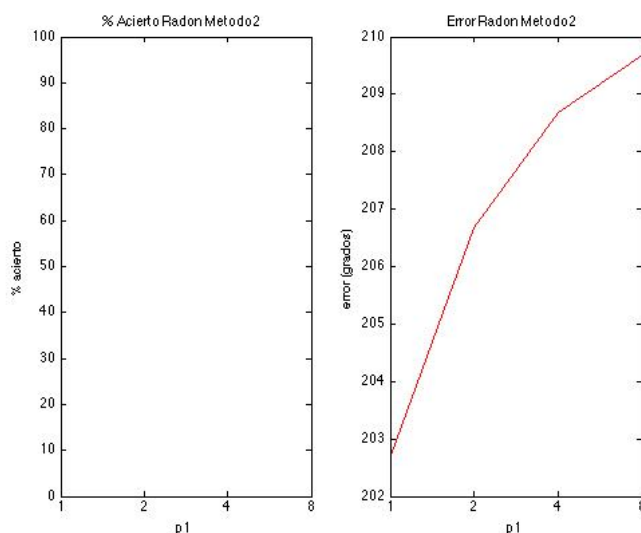
**Figura 6-7:** HOG. Error de orientación con el **día nublado** al variar la anchura de las celdas verticales  $l_1$  y el número de bins por histograma  $b_2$ .



**Figura 6-8:** gist. Error de orientación con el **día nublado** al variar la anchura de las celdas verticales  $l_2$  y el número de filtros de Gabor  $m_2$ .



**Figura 6-9:** RT Método 1. Error de orientación con el **día nublado** al variar el número de grados entre ángulos consecutivos  $p_2$  y el número de columnas  $K_6$ .

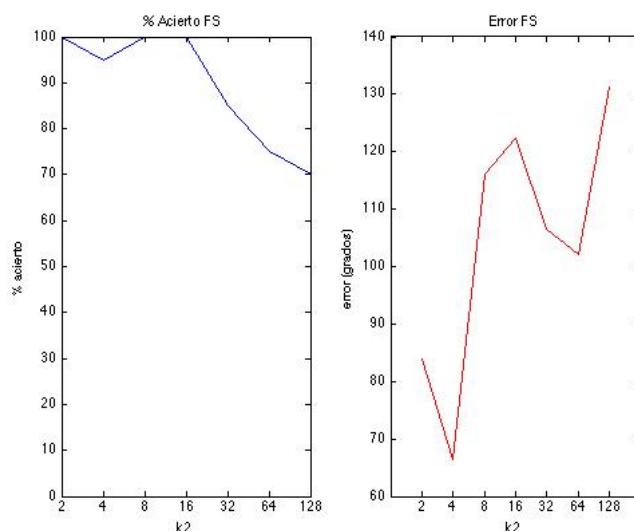


**Figura 6-10:** RT Método 2. Error de orientación con el **día nublado** al variar el número de grados entre ángulos consecutivos  $p_2$ .

Con el descriptor gist obtenemos un porcentaje de detección de ángulo muy alto están con cualquier valor de los parámetros entre el 90 y el 1000 % de detecciones. Por otro lado en el cálculo del error de orientación se obtienen los mejores resultados con valores medios del parámetro  $l_2$  con los que se obtiene un error medio de 10 a 25 grados, destacando el error calculado con  $l_2=32$  y  $m_2=64$  en el que el error medio calculado es menor a 10 grados.

Por último al utilizar el descriptor Transformada de Radon siempre se tiene un porcentaje de detección del ángulo del 100 %. Sin embargo los resultados a la hora de calcular el error de orientación son malos comparados con los obtenidos por los otros tres descriptores. Con el método 1 el error aumenta cuando el parámetro  $p_2$  llegando a tener errores medios mayores a 120 grados con un paso de  $p_2=8$ , el error disminuye cuando se tiene un paso  $p_2$  sin que el valor de  $k_6$  influya demasiado en el resultado. El menor error se obtiene con  $p_2=1$  y  $k_6=16$  donde el error es de 80 grados, un error bastante alta para ser mínimo. Por su parte el método 2 ofrece resultados nefastos no bajando ningún valor de error medio de 202 grados aumentando este error conforme aumenta el valor de  $p_2$ .

Analizando los resultados, si elegimos los valores adecuados el descriptor gist es el que mejor resultados ofrece cuando se juega con el error en la orientación. El resultados obtenidos con el descriptor HOG también podrían considerarse adecuados al igual que los que ofrece el descriptor Firma de Fourier pero con este habría que tener cuidado de no realizar la tarea con un valor alto de  $k_2$  ya que en ese caso el ángulo detectado por el descriptor es 0 y por tanto se tendrá un error en la medida. Por otro lado, el descriptor Radon, en ambos métodos,



**Figura 6-11:** FS. Error de orientación con el día soleado al variar el número de columnas  $k_2$ .

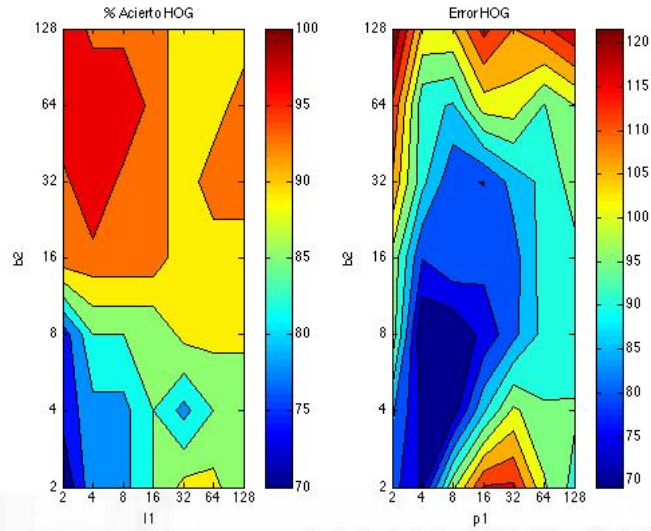
no es el adecuado para detectar un cambio de orientación en la navegación de un robot móvil.

A continuación se realiza el mismo estudio comparando imágenes tomadas un día soleado con las imágenes de la base de datos las cuales estaban sacadas un día nublado. Se puede ver los resultados en las figuras 6-11, 6-12, 6-13, 6-14 y 6-15. Así podemos comprobar si los descriptores de apariencia global son un método suficientemente robusto como para dar buenos resultados cuando hay un cambio de iluminación.

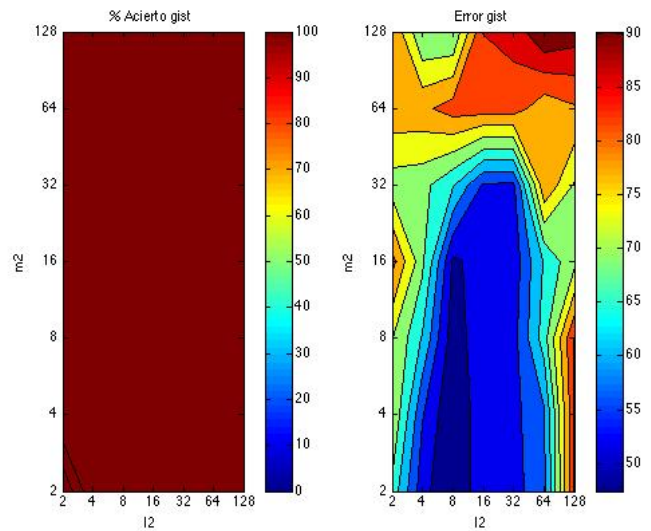
Con el descriptor Firma de Fourier obtenemos unos porcentajes de detección altos (95-100 %) cuando el valor de la variable  $k_2$  es 32 o menor, sin embargo conforme el valor de ese parámetro aumenta el porcentaje de detección de ángulo baja llegando a un porcentaje del 70 % con  $k_2=128$ . Por su parte, el error medio con este descriptor es de variante. El mejor resultado se obtiene con  $k_2=4$  en el que el error medio de orientación es menor a 70 grados, al aumentar el valor de  $k_2$  el error medio aumenta y se tiene el error más alto en este estudio con  $k_2=128$  donde se tiene un fallo medio en el cálculo de ángulo de aproximadamente 130 grados.

Al utilizar el descriptor global HOG se obtiene resultado entre un 70 y un 100 % de las veces. El parámetro que más influye en este porcentaje de detección es  $b_2$ , siendo directamente proporcional, generalmente cuanto mayor es  $b_2$  mayor es el porcentaje. Con respecto al error de orientación se obtienen los mejores resultados con valores medio-bajos del parámetro  $b_2$  y valores bajos de  $l_2$  donde se encuentran errores medios en torno a 70 y 80 grados.

El comportamiento ante e descriptor gist es que con en este caso se obtiene resultado de



**Figura 6-12:** HOG. Error de orientación con el **día soleado** al variar la anchura de las celdas verticales  $l_1$  y el número de bins por histograma  $b_2$ .



**Figura 6-13:** gist. Error de orientación con el **día soleado** al variar la anchura de las celdas verticales  $l_2$  y el número de filtros de Gabor  $m_2$ .

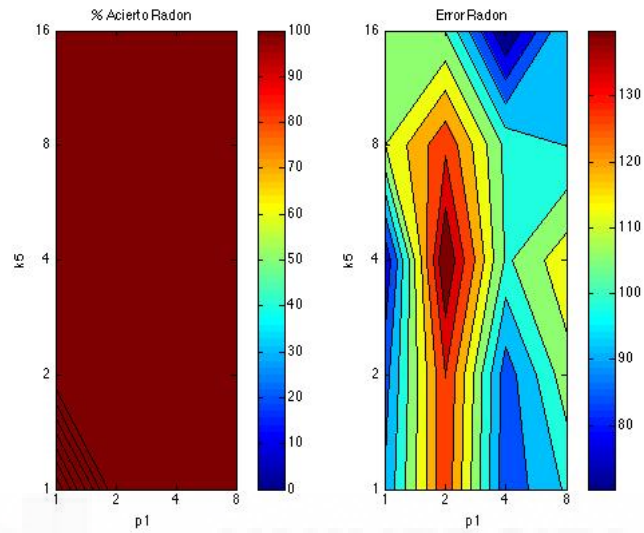


Figura 6-14: RT Método 1. Error de orientación con el **día soleado** al variar el número de grados entre ángulos consecutivos  $p_2$  y el número de columnas  $K_6$ .

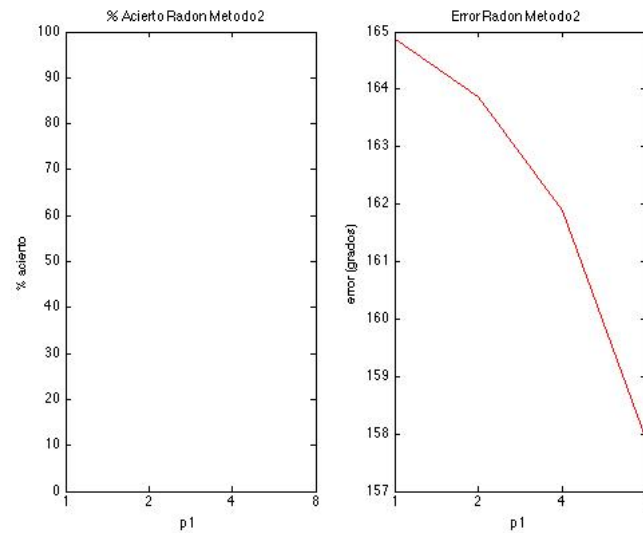


Figura 6-15: RT Método 2. Error de orientación con el **día soleado** al variar el número de grados entre ángulos consecutivos  $p_2$ .

ángulo siempre, es por ello que el porcentaje de detección es siempre del 100%. Respecto a al error de orientación, se obtienen errores más bajos con valores medios de  $l_2$  y valores medio-bajos del parámetro  $m_2$  en los que el error es de 50 o 60 grados. El error medio más bajo se obtiene con  $l_2=8$  y  $m_2=2$  en el que el error medio en el cálculo de la diferencia de ángulo es menor a 50 grados.

Como en el caso de las pruebas con las imágenes tomadas en día soleado, los resultados obtenidos con el descriptor Transformada de Radon son bastante malos. Como antes, detectan un cambio de ángulo el 100% de las veces pero el error medio es con el método 1 mayor a 80 grados llegando a resultados con error medio de 130 grados cuando el valor del paso entre ángulos  $p_2$  es 2 y  $k_5=4$ . Con el método 2 el error baja al aumentar  $p_2$  pero todos los errores medios calculados son mayores a 160 grados.

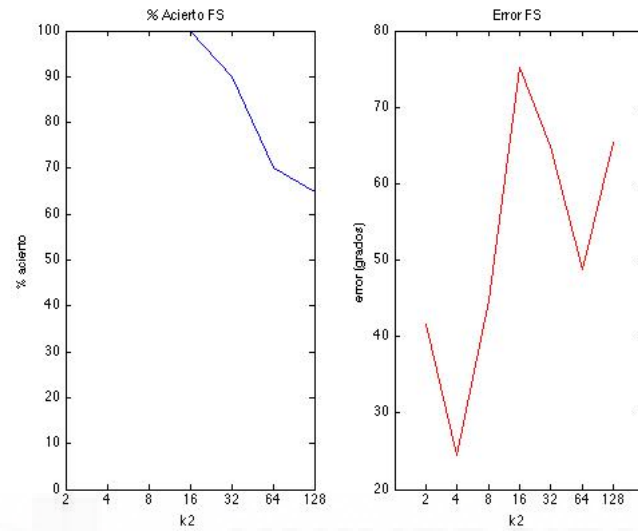
Como conclusión se obtiene que al haber un cambio de iluminación entre el momento que se realiza la base de datos y el momento en el que se realiza el cálculo de la orientación se tiene un error de cálculo grande. Como en el primer caso, el descriptor que mejor resultados ofrece es gist ya que llega a calcular un error menor a 50 grados. Por su parte el descriptor Firma de Fourier, HOG y el primer método de Radon ofrecen uno errores medios entre 70 y 120 grados. El segundo método de la transformada de Radon no es adecuado para esta serie de tareas ya que el error calculado es mayor a 160 grados. por otro lado, destacar que el porcentaje de detección de que ha habido un cambio de ángulo ha aumentado respecto a los resultados que se habían calculado con el día nublado la cual puede ser una característica a tener en cuenta.

Por último se realiza la prueba cuando la iluminación natural es escasa y manteniendo la base de datos que se obtuvo con día nublado. Los resultados pueden observarse en las figuras **6-16**, **6-17**, **6-18**, **6-19** y **6-20** donde se muestran los resultados al ir variando los parámetros que se desea estudiar.

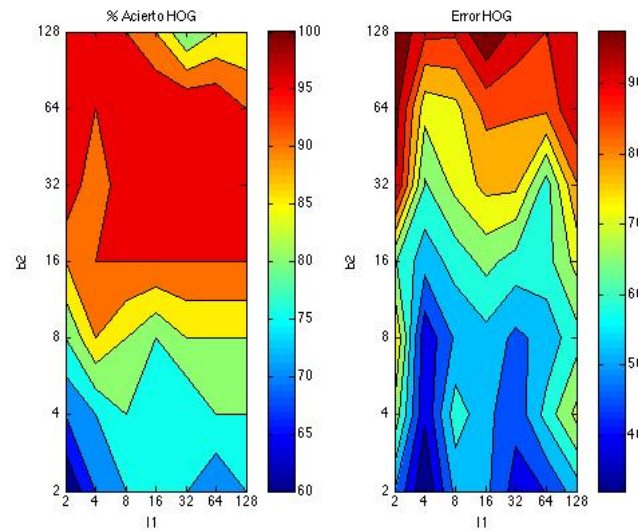
El resultado al utilizar la Firma de Fourier es un porcentaje de detección de cambio de ángulo del 100% hasta que el valor de  $k_2$  es 16 donde este valor de acierto empieza a ser más pequeño llegando a ser del 65% cuando  $k_2=128$ . El resultado en el error de orientación es variable pero se obtiene una gráfica similar a la obtenida en las pruebas de día soleado (figura **6-11**), mínimos locales con  $k_2=4$  y  $k_2=64$  y máximos locales con  $k_2=16$  y  $k_2=128$ . El menor error se obtiene con  $k_2=4$  y es de 25 grados.

Al utilizar como descriptor el método HOG el porcentaje de acierto está entre 60 y 100% y depende sobre todo del parámetro  $b_2$ , cuando  $b_2 \geq 16$  el porcentaje de existencia de un cambio de ángulo es siempre mayor dl 90%. Por parte del error de orientación, cuanto menor es el valor de  $b_2$  mejores resultados se obtienen afectando de manera desigual el valor de  $l_1$ .

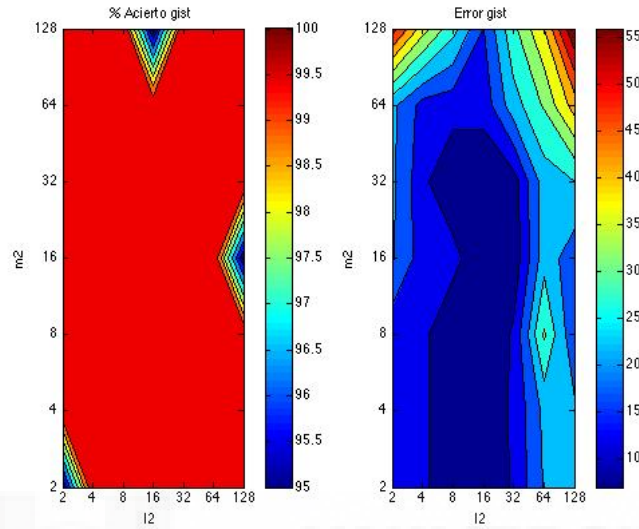




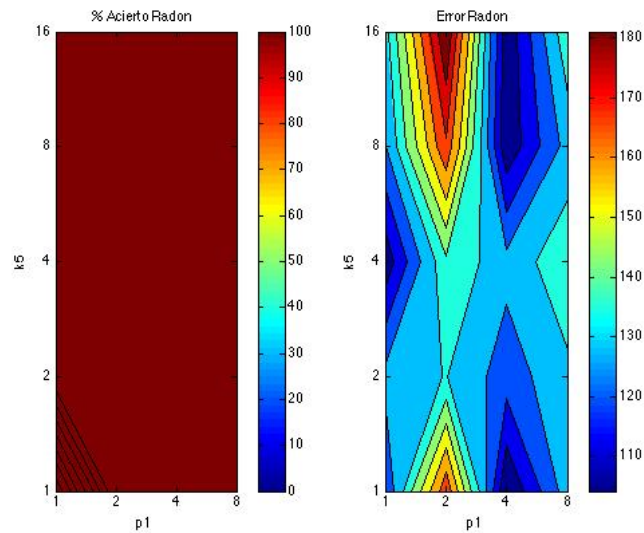
**Figura 6-16:** FS. Error de orientación cuando ha **anochecido** al variar el número de columnas  $k_2$ .



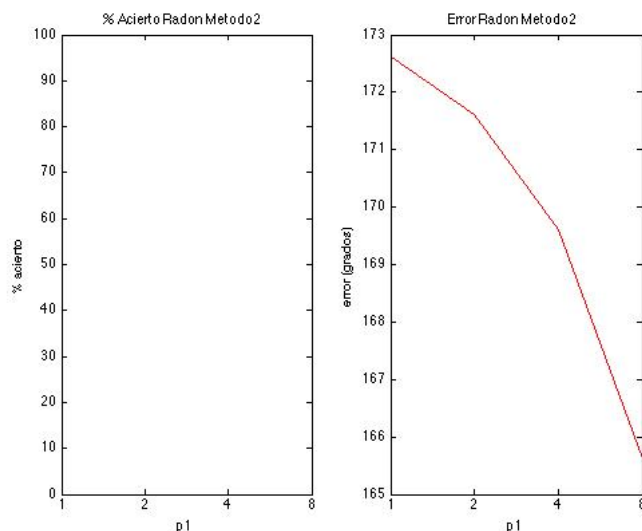
**Figura 6-17:** HOG. Error de orientación cuando ha **anochecido** al variar la anchura de las celdas verticales  $l_1$  y el número de bins por histograma  $b_2$ .



**Figura 6-18:** gist. Error de orientación cuando ha **anochecido** al variar la anchura de las celdas verticales  $l_2$  y el número de filtros de Gabor  $m_2$ .



**Figura 6-19:** RT Método 1. Error de orientación cuando ha **anochecido** al variar el número de grados entre ángulos consecutivos  $p_2$  y el número de columnas  $K_6$ .



**Figura 6-20:** RT Método 2. Error de orientación cuando ha **anochecido** al variar el número de grados entre ángulos consecutivos  $p_2$ .

El mínimo error medio se obtiene con  $b_2=2$  y  $l_1$ , con estos parámetros el error obtenido es menor de 40 grados.

Cuando se utiliza el descriptor global gist el porcentaje de detección es aproximadamente 100 % en todos los caso, exceptuando ciertas ocasiones en las que el porcentaje es del 95 %. Por tanto se trata de un descriptor con un porcentaje de acierto bastante bueno. Si se observa los resultados obtenidos en el error medio se observa que es bastante bajo, en torno a 10 o 20 grados de error en la mayoría de los caso; exceptuando cuando  $m_2=128$  y el valor de  $l_2$  es extremo, cuando  $l_2=2$  el error medio es de 50 grados y cuando  $l_2=128$  el error es de 55 grados.

Por último, con la Transformada de Radon se obtiene siempre un 100 % en la detección de existencia de un cambio de orientación entre la imagen de la base de datos y la imagen que se está estudiando. Pero al observar los resultados en el cálculo del error de orientación nuevamente se encuentra un error bastante alto. En el método 1 el error medio mínimo es de 110-120 grados llegando a tener un fallo medio de 180 grados cuando  $p_2=2$  y  $k_5=16$ . Con el método 2 el error medio en todos los casos es aproximadamente 170 grados. Errores grandes para realizar tareas de navegación.

Observando los resultados se detecta que la transformada de Radon no es un descriptor bueno para realizar tareas en las que se precise una buena detección de la orientación. Los descriptores de Fourier y HOG ofrecen unos resultados decentes ya que Fourier puede realizar la tarea con un error de 20 grados y el descriptor HOG con un error menor a 40 grados. Pero el descriptor que mejor se adecua a esta labor es el descriptor global gist que puede realizar

la tarea con un error medio menor a 10 grados. Como en el caso de los errores calculados con el día soleado, cuando se calcula el error de orientación cuando ha anochecido se obtiene un porcentaje de detección de cambio de ángulo mayor que cuando se realiza la misma prueba con el día soleado.

Realizar la tarea de detectar un cambio de orientación del robot utilizando únicamente cámaras omnidireccionales supone un reto bastante grande. A este reto se le debe añadir que se ha realizado cuando hay cambio de iluminación entre el día que se creó la base de datos y el día que se quiere calcular ese cambio de orientación sumado a los cambios que el ser humano haya podido añadir a la escena. Por tanto no se trata de una tarea trivial de la que se ha obtenido unos resultados que según para que tarea podrían ser tomados como buenos.



## 7 Conclusión

El trabajo está enfocado al estudio de los problemas de las tareas de navegación cuando hay cambios de iluminación en el ambiente y solo se dispone de cámara omnidireccionales como sensor de medida. El problema se ha abordado realizando una serie de pruebas en las que se ha resuelto tareas de localización y detección de cambios de orientación cuando existe dicho cambio de iluminación. El robot móvil que realizaba esta tarea estaba equipado por un sistema de visión catadióptrico montado sobre el y únicamente con las imágenes que capturaba se han resuelto los problemas de localización y orientación.

Para la extracción de información de la imagen se han utilizado descriptores de apariencia global como son la Firma de Fourier, HOG, gist y la transformada de Radon. Tras describir las imágenes y observar los resultados se ha comparado los datos obtenidos con los cuatro descriptores. Con los algoritmos nombrados se han realizado las tareas de localización y orientación por separado de manera separada y secuencial. Primero se realiza la tarea de localización y a continuación la tarea de estimación de la orientación.

Además, también se ha realizado una evaluación del coste computacional que supone realizar estas tareas. Por un lado FS y Radon ofrecen los valores de tiempo más razonables para realizar tareas en tiempo real. Por otro lado los resultados obtenidos en estas tareas muestran que los descriptores HOG y gist ofrecen un mejor resultado ante estas labores, ambos tienen resultados remarcables en las tareas de localización pero en la tarea de estimación de la orientación el descriptor gist destaca por encima del resto de descriptores.

Los resultados muestran que los descriptores de apariencia global son un método bastante viable para realizar tareas de navegación. Con ellos un robot puede construir un modelo y utilizarlo para estimar la posición y orientación del robot en instantes futuros. Este hecho puede suponer interesantes aplicaciones en el mundo de la robótica móvil. Gracias a la flexibilidad que ofrecen los descriptores pueden utilizarse para crear mapas híbridos donde conviven diferentes mapas con diferente nivel de detalle en distintos niveles. Un nivel alto permitirá una localización más basta y rápida mientras que un nivel más bajo ofrecerá una localización más lenta pero de mayor precisión.

Estos hechos animan a profundizar en este campo. Un hecho importante podría ser la construcción de un sistema de navegación completamente autónomo. Para ello, primero se

debe automatizar un proceso en el que la captura de imágenes obtenga una representación óptima del entorno. En segundo lugar, las tareas de mapping y localización se deben integrar en una tarea de SLAM que realice a la vez ambas actividades creando el mapa y localizando desde el principio. Por último, otra línea de trabajo también podría ser optimizar los algoritmos para que el coste computacional se vea reducido.



# Bibliografía

- [1] G. Giralt, R. Sobek, and R. Chatila, “A multi-level planning and navigation system for a mobile robot: A first approach to hilare,” in *Proceedings of the 6th International Joint Conference on Artificial Intelligence - Volume 1, IJCAI'79*, (San Francisco, CA, USA), pp. 335–337, Morgan Kaufmann Publishers Inc., 1979.
- [2] L. Payá, A. Gil, and O. Reinoso, “A state-of-the-art review on mapping and localization of mobile robots using omnidirectional vision sensors,” *Journal of Sensors*, vol. 2017, 2017.
- [3] O. Wijk and H. I. Christensen, “Localization and navigation of a mobile robot using natural point landmarks extracted from sonar data,” *Robotics and Autonomous Systems*, vol. 31, no. 1, pp. 31–42, 2000.
- [4] Y. Cha and D. Kim, “Omni-directional image matching for homing navigation based on optical flow algorithm,” pp. 1446–1451, 2012. cited By 1.
- [5] A. Hata and D. Wolf, “Outdoor mapping using mobile robots and laser range finders,” pp. 209–214, 2009. cited By 6.
- [6] J. Choi, S. Ahn, M. Choi, and W. Chung, “Metric slam in home environment with visual objects and sonar features,” pp. 4048–4053, 2006. cited By 14.
- [7] K. Hara, S. Maeyama, and A. Gofuku, “Navigation path scanning system for mobile robot by laser beam,” pp. 2817–2821, 2008. cited By 2.
- [8] G. N. DeSouza and A. C. Kak, “Vision for mobile robot navigation: A survey,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 24, no. 2, pp. 237–267, 2002.
- [9] F. Bonin-Font, A. Ortiz, and G. Oliver, “Visual navigation for mobile robots: A survey,” *Journal of intelligent and robotic systems*, vol. 53, no. 3, p. 263, 2008.
- [10] P. Sturm, S. Ramalingam, J.-P. Tardif, S. Gasparini, and J. a. Barreto, “Camera models and fundamental concepts used in geometric computer vision,” *Found. Trends. Comput. Graph. Vis.*, vol. 6, pp. 1–183, Jan. 2011.

- [11] D. Scaramuzza, R. Siegwart, and A. Martinelli, “A robust descriptor for tracking vertical lines in omnidirectional images and its use in mobile robotics,” *The International Journal of Robotics Research*, vol. 28, no. 2, pp. 149–171, 2009.
- [12] V. Grassi Junior and J. Okamoto Junior, “Development of an omnidirectional vision system,” *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 28, pp. 58 – 68, 03 2006.
- [13] N. Pears and B. Liang, “Ground plane segmentation for mobile robot visual navigation,” in *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180)*, vol. 3, pp. 1513–1518 vol.3, 2001.
- [14] C. Harris and M. Stephens, “A combined corner and edge detector.,” in *Alvey vision conference*, vol. 15, pp. 10–5244, Manchester, UK, 1988.
- [15] D. G. Lowe, “Object recognition from local scale-invariant features,” in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2, pp. 1150–1157, Ieee, 1999.
- [16] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [17] H. Bay, T. Tuytelaars, and L. Van Gool, “Surf: Speeded up robust features,” *Computer vision–ECCV 2006*, pp. 404–417, 2006.
- [18] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, “Speeded-up robust features (surf),” *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.
- [19] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, “Brief: Binary robust independent elementary features,” *Computer Vision–ECCV 2010*, pp. 778–792, 2010.
- [20] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “Orb: An efficient alternative to sift or surf,” in *Computer Vision (ICCV), 2011 IEEE international conference on*, pp. 2564–2571, IEEE, 2011.
- [21] S. Leutenegger, M. Chli, and R. Y. Siegwart, “Brisk: Binary robust invariant scalable keypoints,” in *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 2548–2555, IEEE, 2011.
- [22] A. Alahi, R. Ortiz, and P. Vandergheynst, “Freak: Fast retina keypoint,” in *Computer vision and pattern recognition (CVPR), 2012 IEEE conference on*, pp. 510–517, Ieee, 2012.



- [23] S. Se, D. G. Lowe, and J. J. Little, "Vision-based global localization and mapping for mobile robots," *IEEE Transactions on robotics*, vol. 21, no. 3, pp. 364–375, 2005.
- [24] A. C. Murillo, J. J. Guerrero, and C. Sagues, "Surf features for efficient robot localization with omnidirectional images," in *Robotics and Automation, 2007 IEEE International Conference on*, pp. 3901–3907, IEEE, 2007.
- [25] C. Valgren and A. J. Lilienthal, "Sift, surf & seasons: Appearance-based long-term localization in outdoor environments," *Robotics and Autonomous Systems*, vol. 58, no. 2, pp. 149–156, 2010.
- [26] M. Kirby, *Geometric Data Analysis: An Empirical Approach to Dimensionality Reduction and the Study of Patterns*. New York, NY, USA: John Wiley & Sons, Inc., 2000.
- [27] L. Payá, L. Fernández, Ó. Reinoso, A. Gil, and D. Úbeda, "Appearance-based dense maps creation-comparison of compression techniques with panoramic images.," in *ICINCO-RA*, pp. 250–255, 2009.
- [28] F. Rossi, A. Ranganathan, F. Dellaert, and E. Menegatti, "Toward topological localization with spherical fourier transform and uncalibrated camera," in *Proceedings of the International Conferences on Simulation, Modeling and Programming for Autonomous Robots*, pp. 319–330, 2008.
- [29] 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. 3, pp. 145–175, 2001.
- [30] H. Ishiguro and S. Tsuji, "Image-based memory of environment," in *Intelligent Robots and Systems' 96, IROS 96, Proceedings of the 1996 IEEE/RSJ International Conference on*, vol. 2, pp. 634–639, IEEE, 1996.
- [31] W. Stürzl and H. A. Mallot, "Efficient visual homing based on fourier transformed panoramic images," *Robotics and Autonomous Systems*, vol. 54, no. 4, pp. 300–313, 2006.
- [32] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 886–893, IEEE, 2005.
- [33] Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan, "Fast human detection using a cascade of histograms of oriented gradients," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 2, pp. 1491–1498, IEEE, 2006.

- [34] M. Hofmeister, M. Liebsch, and A. Zell, “Visual self-localization for small mobile robots with weighted gradient orientation histograms,” in *40th International Symposium on Robotics (ISR)*, pp. 87–91, Barcelona, 2009.
- [35] L. Payá, L. Fernández, A. Gil, and O. Reinoso, “Map building and monte carlo localization using global appearance of omnidirectional images,” *Sensors*, vol. 10, no. 12, pp. 11468–11497, 2010.
- [36] A. Oliva and A. Torralba, “Building the gist of a scene: The role of global image features in recognition,” *Progress in brain research*, vol. 155, pp. 23–36, 2006.
- [37] A. Torralba, “Contextual priming for object detection,” *International journal of computer vision*, vol. 53, no. 2, pp. 169–191, 2003.
- [38] C. Siagian and L. Itti, “Rapid biologically-inspired scene classification using features shared with visual attention,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 2, pp. 300–312, 2007.
- [39] A. Pronobis and B. Caputo, “COLD: COsy Localization Database,” *The International Journal of Robotics Research (IJRR)*, vol. 28, pp. 588–594, May 2009.