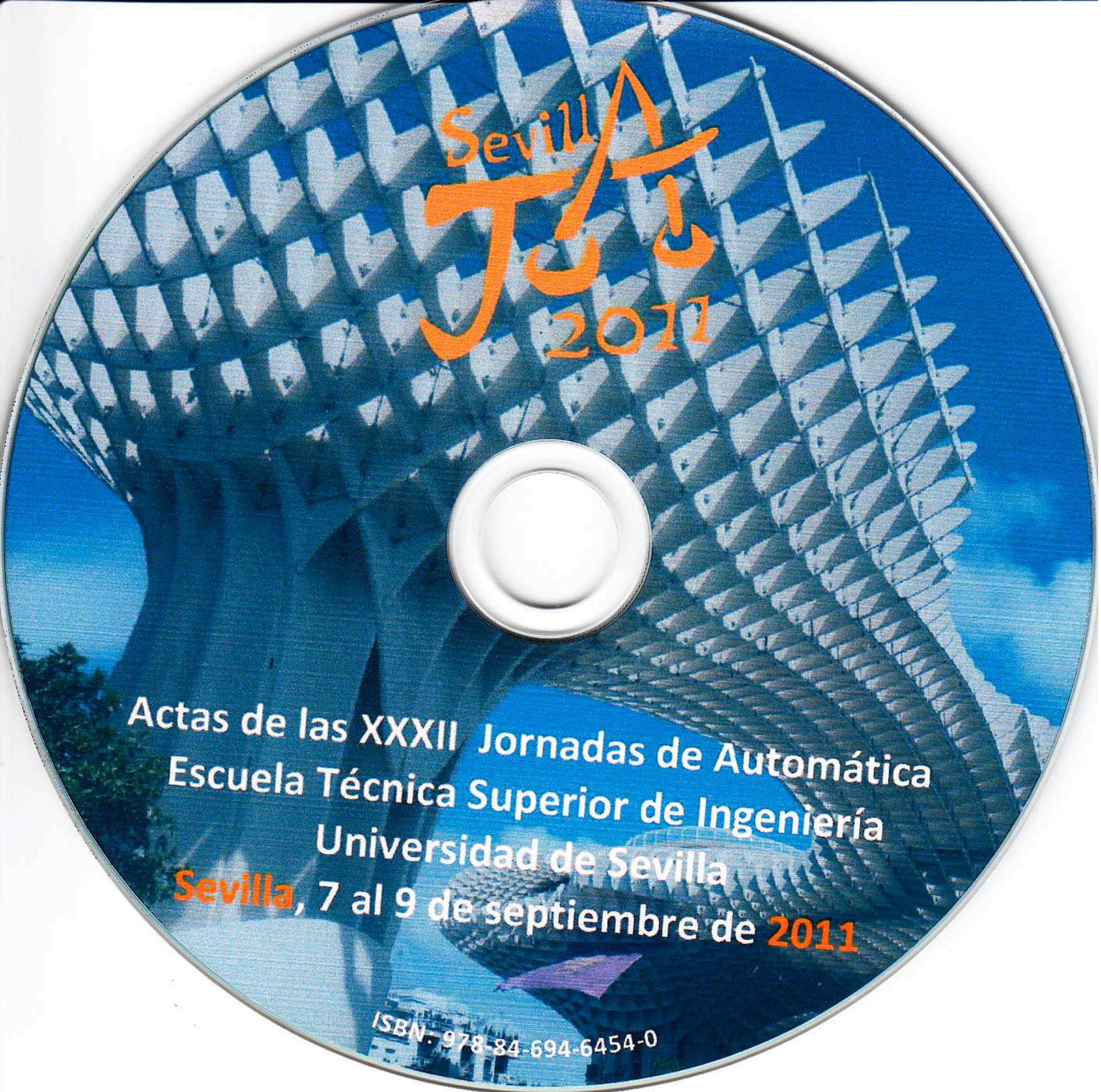
Página principal
Presentación
Comités
Comunicaciones
Programa
Patrocinadores
Contacto

Las Jornadas de Automática constituyen un punto de encuentro para la comunidad de Automática en España tanto en el ámbito de la investigación, la docencia o la empresa. Las Jornadas vienen realizándose anualmente desde el año 1977, promovidas por el Comité Español de Automática (CEA), y organizadas por distintas Universidades o Centros de Investigación de nuestro país. Este año 2011 celebrarán su XXXII edición en la Universidad de Sevilla, durante los días 7, 8 y 9 de septiembre en la Escuela Técnica Superior de Ingeniería, siendo ésta la segunda ocasión en la que se desarrollará en Sevilla.


Sevilla
JA
2011
Actas de las XXXII Jornadas de Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla
Sevilla, 7 al 9 de septiembre de 2011
ISBN: 978-84-694-6454-0

Comunicaciones aceptadas en Visión por Computador

ID.	TÍTULO	AUTOR/ES
P14	SOFTWARE PARA LA DETECCIÓN DE OBJETOS SUMERGIDOS FABRICADOS POR EL HOMBRE BASADO EN EL POST-PROCESADO DIGITAL DE IMÁGENES PROCEDENTES DE UN SONAR DE BARRIDO LATERAL DE SIMPLE HAZ	José María Zamanillo, Isabel Zamanillo, Manuel Salas, Elías Revestido, Francisco-Jesús Velasco
P26	DETECCIÓN Y SEGUIMIENTO DE OBJETOS SÓLIDOS CON CAMARAS TOF	Thomas Kisler, Pablo Gil
P28	CONTROL VISUAL DE UN VEHÍCULO RC	David Sánchez Benítez, Jesús de la Cruz García, Gonzalo Pajares
P34	REPLICACIÓN DE MOVIMIENTOS CON MARCADORES VISUALES PARA LA INTERACCIÓN HOMBRE-MÁQUINA	Lucas Martín-Poggi, Santiago Puente-Méndez, Pablo Gil-Vázquez, Fernando Torres-Medina
P43	DISEÑO DE UN AMBIENTE VIRTUAL MEDIANTE VISIÓN ARTIFICIAL PARA APLICACIONES EN TELEROBÓTICA	Jaime Masiá, Juan Ramón Rufino, Francisco Javier Tormo, José Ferri, Borja Mansilla
P45	CONSTRUCCIÓN DE MAPAS VISUALES CON IMÁGENES OMNIDIRECCIONALES	David Valiente, Arturo Gil, José Mª Marín, Lorenzo Fernández, Óscar Reinoso
P49	DETECTORES DE PUNTOS DE INTERÉS. ¿SON SUFICIENTEMENTE ROBUSTOS?	Joan Climent, Miguel Ángel Cataño Sánchez
P53	CONTROL MEDIANTE VISIÓN ARTIFICIAL DE UN CUADRICÓPTERO	Francisco Vázquez, Salvador Rodríguez, Rafael Ruiz
P55	ADAPTIVE FILTERS EVALUATION FOR SHARPNESS ENHANCEMENT AND NOISE REMOVAL	Oscar García-Olalla, Diego García-Ordás, María Teresa García-Ordás, Laura Fernández-Robles, Enrique Alegre
P56	WORDS RECOGNITION USING METHODS OF WORD SHAPE CODING	María Teresa García-Ordás, Laura Fernández-Robles, Oscar García-Olalla, Diego García-Ordás, Enrique Alegre
P57	VIDEO SEGMENTATION COMBINING DEPTH MAPS AND INTENSITY IMAGES	Diego García-Ordás, Oscar García-Olalla, Laura Fernández-Robles, María Teresa García-Ordás, Enrique Alegre
P65	VISIÓN 3D DE BAJO COSTE PARA LA INTERACCIÓN HUMANO-ROBOT UTILIZANDO FILTRO DE PARTÍCULAS	Enrique Martínez Berti, David Alejandro Hernández Campos, Antonio Sánchez
P75	SELECCIÓN DE CARACTERÍSTICAS MEDIANTE COMPUTACIÓN EVOLUTIVA PARA CLASIFICADORES DE DEFECTOS SUPERFICIALES IDENTIFICADOS CON VISIÓN POR COMPUTADOR	Silvia Satorres Martínez, Alejandro Sánchez García, Juan Gómez Ortega, Juan Pedro Aranda Carmona, Javier Gámez García
P79	CLASIFICACIÓN DE MATERIALES EN FUNCIÓN DE SU RUGOSIDAD UTILIZANDO DESCRIPTORES DE LAWS	Sir Alexci Suarez Castrillon, Enrique Alegre Gutiérrez, Joaquín Barreiro García
P81	FUSIÓN DE IMÁGENES DE PROFUNDIDAD Y ESPECTRO VISIBLE EN ASM FACIAL	Ismael Serrano, Gloria Bueno, Jesús Salido, Oscar Déniz
P84	ACROSOME EVALUATION OF SPERMATOZOA CELLS USING SIFT AND CLASSICAL TEXTURE DESCRIPTORS	Laura Fernández-Robles, María Teresa García-Ordás, Diego García-Ordás, Oscar García-Olalla, Enrique Alegre
P109	CÁMARAS BASADAS EN TIEMPO DE VUELO. USO EN LA MEJORA DE MÉTODOS DE DETECCIÓN DE CARAS	José Raúl Ruiz-Sarmiento, Cipriano Galindo, Javier González-Jiménez
P118	EVALUACIÓN DE LA PLATAFORMA CUDA MEDIANTE UN ALGORITMO DE PUESTA EN CORRESPONDENCIA ESTÉREO	Alberto Castillo-Baquero, Eusebio de la Fuente-López
P120	CLASIFICADOR AUTOMÁTICO DE PLÁSTICO TÉCNICO RECICLADO UTILIZANDO VISIÓN POR COMPUTADOR Y FUSIÓN SENSORIAL	José Manuel López Paniza, Javier Gámez García, Juan Gómez Ortega, Silvia Satorres Martínez, Matías Cobo Ramírez, Juan Pedro Aranda Carmona
P133	CALIBRACIÓN DE UN SISTEMA DE ESTEREOVISIÓN PARA LA MEDICIÓN ONLINE DE PARÁMETROS DE CRECIMIENTO VEGETAL	José Antonio Cantón Claro, Julián García Donaire, Julián Sánchez-Hermosilla López
P164	SISTEMA DE RECONOCIMIENTO FACIAL BASADO EN PCA	Daniel Gómez-Muñoz, Eusebio de la Fuente-López

CONSTRUCCIÓN DE MAPAS VISUALES CON IMÁGENES OMNIDIRECCIONALES

David Valiente, Arturo Gil, J. M. Marín, Lorenzo Fernández, Óscar Reinoso,
Departamento de Ingeniería de Sistemas y Automática.
Universidad Miguel Hernández, Avda. de la Universidad s/n, Elche (Alicante).
dvaliente@umh.es, arturo.gil@umh.es, jmarin@umh.es, l.fernandez@umh.es, o.reinoso@umh.es

Resumen

En este artículo se describe una solución al problema de SLAM Simultaneous Localization and Mapping *basado en una única cámara omnidireccional. Consideramos que el robot está equipado con un único sensor catadióptrico y que es capaz de extraer puntos de interés de las imágenes. En esta solución, el mapa está representado por un conjunto de imágenes omnidireccionales y sus posiciones, teniendo cada imagen omnidireccional un conjunto de puntos característicos y de descriptores visuales asociados. Cuando el robot se mueve por el entorno captura imágenes omnidireccionales y extrae un conjunto de puntos de interés de ellas. A continuación, busca correspondencias con el resto de imágenes omnidireccionales existentes en el mapa. Si se encuentra un número suficiente de correspondencias entre las imágenes, se calcula una rotación y translación (salvo un factor de escala) entre ambas imágenes. A partir de estas medidas podemos deducir la localización del robot con respecto a las imágenes almacenadas en el mapa. Se presentan resultados obtenidos en un entorno simulado que validan la idea presentada. Además, se presentan resultados obtenidos utilizando datos reales que demuestran la validez de la solución presentada.*

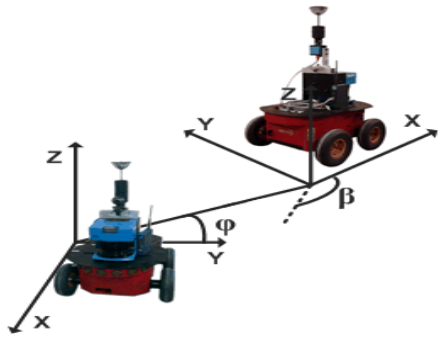
Palabras clave: SLAM, SLAM visual, imagen omnidireccional.

1. INTRODUCCIÓN

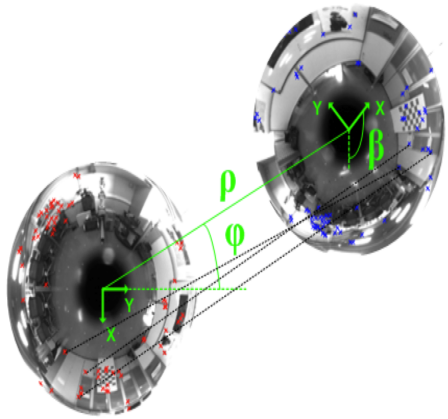
Un gran número de aplicaciones de robótica móvil precisan de la existencia de un mapa para realizar la tarea. Así pues, la construcción del mapa es un requisito indispensable para que un robot móvil sea realmente autónomo. Para construir el mapa el robot debe explorar el entorno adquiriendo información que le permita construir un mapa coherente. Durante este proceso la pose del robot es desconocida, con lo que se genera el problema de construir un mapa mientras, simultáneamente, el robot se localiza dentro de él. Este problema ha recibido el nombre de SLAM *Simultaneous Localization and Mapping* y hasta el momento ha sido abordado por un gran número de investigadores.

Las soluciones de SLAM se pueden agrupar en función del tipo de sensor utilizado para construir el mapa. Por ejemplo, los sensores de distancia láser han sido utilizados con frecuencia para la creación de mapas de ocupación [18] o bien para la construcción de mapas basados en *landmarks* [13]. Durante los últimos años, un gran número de trabajos proponen la utilización de cámaras como sensor principal para la construcción de mapas. Estas aplicaciones se agrupan generalmente bajo el nombre de SLAM visual. En este grupo podemos encontrar diferentes alternativas. Por ejemplo, en [7] se utiliza un par estéreo de cámaras calibradas para obtener medidas relativas de distancia a un conjunto de marcas visuales, estando cada *landmark* acompañada de un descriptor visual. En [4] una única cámara se utiliza para construir un mapa tridimensional del entorno, constituido por un conjunto de puntos de interés extraídos con el detector de esquinas de Harris [8]. La cámara se mueve a mano y permite extraer la posición 3D de los puntos cuando estos son observados desde puntos de vista separados. Según [1] los resultados de SLAM visual utilizando una única cámara son mejores cuando se utiliza una óptica con un gran ángulo de visión, hecho que sugiere la utilización de una cámara omnidireccional para la creación del mapa, ya que el ángulo de visión horizontal es máximo.

Consideramos el caso en el que el robot está equipado con una única cámara omnidireccional, según se muestra en la figura 1(a). Cuando el robot se mueve por el entorno captura imágenes omnidireccionales y extrae un conjunto de puntos de interés de ellas. A continuación, busca correspondencias con el resto de imágenes omnidireccionales existentes en el mapa. Si se encuentra un número suficiente de correspondencias entre las imágenes, se calcula una rotación y translación (salvo un factor de escala) entre ambas imágenes [16]. A partir de estas medidas podemos deducir la localización del robot con respecto a las imágenes almacenadas en el mapa. El cálculo de la rotación y translación está detallado en el apartado 3. En la figura 1(b) se presentan dos imágenes omnidirecciona-



(a)



(b)

Figura 1: La Figura 1(a) muestra la configuración del sensor usado en los experimentos. La Figura 1(b) presenta dos imágenes omnidireccionales reales, con varias correspondencias indicadas.

les donde se han indicado algunas correspondencias. Durante los experimentos se han utilizado las características SURF para la detección y descripción de los puntos. En la arquitectura de SLAM presentada aquí, la imagen omnidireccional I_0 se denominará *vista*, para diferenciarla del concepto de *landmark* visual utilizado comúnmente en este ámbito. Así pues, una *vista* está formada por una imagen capturada desde una pose en el entorno que está asociada a un conjunto de puntos de interés. Es importante notar que una landmark visual corresponde a un punto físico en el entorno, como, por ejemplo, una esquina sobre una pared. Sin embargo, una *vista* representa la información visual obtenida desde una pose en particular del entorno.

En este artículo se propone una representación del entorno que se separa del utilizado hasta el momento en el área del SLAM visual. En vez de estimar la posición tridimensional de un conjunto de landmarks visuales en el entorno, se propone la estimación de la posición y orientación de un conjunto de vistas en el entorno. El proceso de construcción del mapa se resume a continuación: supóngase que el robot parte desde el origen del sistema global de referencia. En ese instante, captura una *vista* inicial. Mientras el robot se mueve en las cercanías de esta vista inicial captura imágenes y encuentra puntos correspondientes entre la imagen actual y la vista inicial, calculando una rotación y translación y localizándose respecto de la vista inicial. Cuando el robot se aleja de la vista inicial, no será capaz de encontrar puntos correspondientes. En este momento iniciará una nueva vista en el mapa. Esta nueva vista permitirá la localización del robot en su cercanía.

El concepto de vista ha sido utilizado con anterioridad en el contexto de localización. Por ejemplo en [5] se utilizan un conjunto de imágenes omnidireccionales adquiridas en ciertos puntos del entorno. La localización se basa en un método Monte-Carlo y la comparación de la apariencia global de las imágenes. En [11] se presenta un mapa basado en vistas. Las conexiones entre las diferentes vistas se realizan gracias a las medidas obtenidas mediante un par estéreo.

La solución presentada en este artículo presenta algunas ventajas si la comparamos con otras soluciones de SLAM visual previsas. La ventaja principal radica en la compacidad de la representación del entorno. Por ejemplo, en [1, 4] se utiliza un Filtro de Kalman Extendido (EKF) para estimar la posición de las landmarks visuales, así como la posición y orientación de la cámara. En [4] se utilizan 6 variables para representar cada landmark, con lo que el vector de estado del EKF crece rápidamente con el número de landmarks que se almacenan en el mapa. Este hecho plantea un problema para la mayoría de algoritmos de SLAM, haciendo que los tiempos de cálculo aumenten de forma cuadrática con el número de landmarks en el mapa. En la solución presentada en este artículo, únicamente se estima la posición de un reducido conjunto de vistas. Cada vista encapsula información de un área del entorno en forma de un conjunto de puntos de interés. Según se observará en los experimentos cada vista permitirá la localización del robot en un entorno.

La principal desventaja de la solución presentada aquí es, no obstante, el coste computacional requerido para calcular la transformación entre dos imágenes omnidireccionales. En el apartado 3 detallamos un algoritmo que puede ser utilizado para calcular la observación con una frecuencia alta y permite la realización de SLAM en tiempo real. En este caso, el cálculo de la transformación entre dos imágenes depende sólo del número de correspondencias encontradas en las imágenes.

Presentamos un conjunto de resultados obtenidos

en simulación y utilizando datos reales que permiten demostrar la validez de la solución de SLAM visual presentada.

El resto del artículo se organiza de la siguiente manera. Primero, en el apartado 2 se describe el proceso de SLAM. A continuación, el algoritmo usado para estimar la transformación entre imágenes omnidireccionales se describe en el apartado 3. Seguidamente, el apartado 4 presenta los principales resultados experimentales. Finalmente, las principales conclusiones se exponen en el apartado 5.

2. SLAM

En este apartado se describe en detalle la representación del entorno que se ha elegido, así como el proceso de creación del mapa. Típicamente, la mayoría de soluciones de SLAM visual se centran en estimar la posición tridimensional de un conjunto de landmarks visuales respecto de un sistema de referencia global. En nuestro caso, abordamos el problema de forma diferente: en vez de estimar la posición de un conjunto de landmarks visuales, proponemos la estimación de la posición y orientación de un conjunto de vistas capturadas del entorno. Así pues, el mapa está formado por un conjunto de imágenes omnidireccionales obtenidas desde diferentes posiciones del entorno. En oposición con otras soluciones los elementos a estimar no se corresponden con ningún elemento físico en el entorno (p.e. una esquina). En nuestro caso, cada landmark (bautizada como *vista*) estará constituida por una imagen omnidireccional capturada desde la pose $x_l = (\mathrm{x}_l, \mathrm{y}_l, \theta_l)$ y un conjunto de puntos de interés extraídos de la imagen.

2.1. REPRESENTACIÓN DEL MAPA

Se propone la estimación de la pose $x_v = (\mathrm{x}_v, \mathrm{y}_v, \theta_v)^T$ de un robot móvil en el instante t así como la pose de N vistas. Cada vista i está constituida por su pose $x_{l_i} = (\mathrm{x}_l, \mathrm{y}_l, \theta_l)_i^T$, su incertidumbre P_{l_i} y un conjunto de M puntos de interés p_j expresados en coordenadas de imagen. Cada punto de interés esta asociado con un descriptor visual d_j, $j = 1, \ldots, M$.

La representación de este tipo de mapa se presenta en la figura 2, donde se indica la posición de un conjunto de vistas. Por ejemplo, la vista A se almacena en una pose particular $x_{l_A} = (\mathrm{x}_{l_A}, \mathrm{y}_{l_A}, \theta_{l_A})^T$ en el mapa y tiene un conjunto de M puntos de interés detectados. La vista A permite al localización del robot en el pasillo. La vista B representa la primera habitación, mientras que la vista C describe la habitación 2 y permite la localización del robot dentro de ella.

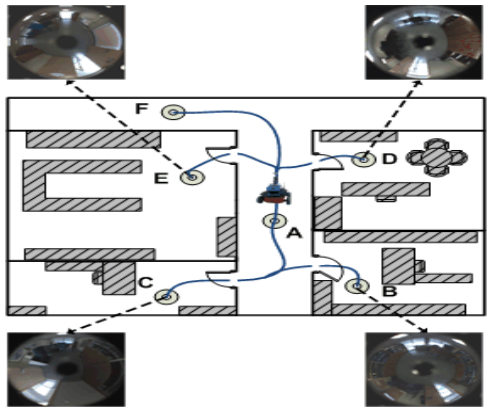


Figura 2: La Figura presenta la idea básica para la construcción del mapa. El robot comienza la exploración en el punto A y almacena una vista I_A en el origen. A continuación se mueve. Cuando no se encuetran correspondencias entre la imagen actual e I_A, una nueva vista es creada en la posición actual del robot, B. El proceso continúa hasta que el entorno queda completamente representado.

El vector de estado aumentado se define como:

$$x = [x_v, x_{l_1}, x_{l_2}, \cdots, x_{l_N}]^T \tag{1}$$

donde N es el número de vistas que existen en el mapa.

2.2. MODELO DE OBSERVACIÓN

A continuación se describe el modelo de observación propuesto. Se asume que existen dos imágenes omnidireccionales obenidas desde dos poses diferentes en el entorno. Una de las imágenes está almacenada en el mapa, mientras que la otra es la imagen actual capturada por el robot. Se asume que hemos sido capaces de encontrar un conjunto de puntos correspondientes en ambas imágenes. Según se describirá en el apartado 3, obtenemos una observación z_t:

$$z_t = \begin{pmatrix} \phi \\ \beta \end{pmatrix} = \begin{pmatrix} \arctan\left(\frac{\mathrm{y}_{l_n} - \mathrm{y}_v}{\mathrm{x}_{l_n} - \mathrm{x}_v}\right) - \theta_v \\ \theta_{l_n} - \theta_v \end{pmatrix} \tag{2}$$

donde el ángulo ϕ es la orientación con la que la vista n es observada y β es la orientación relativa entre ambas imágenes. La vista n está representada por $x_{l_n} = (\mathrm{x}_{l_n}, \mathrm{y}_{l_n}, \theta_{l_n})$, mientras que la pose del robot está descrita por $x_v = (\mathrm{x}_v, \mathrm{y}_v, \theta_v)$. Ambas medidas (ϕ, β) se presentan en la figura 1(a).

2.3. INICIALIZACIÓN DE NUEVAS VISTAS

Una nueva vista se incluye en el mapa cuando el número de correspondencias encontradas con el

resto de imágenes en el mapa es bajo. En concreto, utilizamos la ratio:

$$R = \frac{2m}{n_A + n_B} \tag{3}$$

que calcula el grado de similaridad entre las vistas A y B, siendo m el número total de correspondencias entre A y B, mientras que n_A y n_B son el número de puntos detectados en las imágenes A y B respectivamente. El robot decidirá incluir una nueva vista en el mapa cuando la ratio R cae por debajo de un valor predefinido. En la inicialización de la vista, la pose y la incertidumbre se obtienen de la estimación actual del filtro de SLAM en el instante t.

3. TRANSFORMACIÓN ENTRE IMÁGENES OMNIDIRECCIONALES

En este apartado presentamos un método para obtener los ángulos relativos entre dos imágenes omnidireccionales, β y ϕ. Estos ángulos representan la posición relativa del robot y permiten su localización. Para su obtención deben detectarse puntos característicos en ambas imágenes y encontrar sus correspondencias. Posteriormente se aplica la condición de epipolaridad, de la cual se extrae una cierta rotación y traslación. Los esquemas tradicionales, tales como [10, 15, 19], resuelven el caso general con 6 GOD, mientras que en nuestro caso, atendiendo a la característica del movimiento del robot sobre un plano, podemos reducirlo a 4 variables, reduciendo de este modo el coste computacional. El proceso de obtención de los ángulos relativos entre dos poses del robot tiene una duracción aproximada de $t = 0{,}4ms$, lo cual confirma la capacidad de cómputo en tiempo real.

3.1. DETECCIÓN DE PUNTOS SIGNIFICATIVOS Y CORRESPONDENCIAS

Empleamos SURF [2] con el fin de obtener puntos de interés entre imágenes. Según [6], SURF mejora los resultados de otros detectores y descriptores en términos de robustez de los puntos detectados y de invarianza del descriptor. En [14] se prueba con éxito la detección de puntos SURF en imágenes omnidireccionales. Transformamos la imagen omnidireccional a vista panorámica ya que con esta vista la variación de apariencia es menor, y por ello se consigue aumentar el número de correspondencias válidas entre imágenes. El método para adquirir correspondencias robustas se configura adecuadamente según [12]. Finalmente se realiza el cambio de coordenadas inverso, de vista panorámica a omnidireccional, para volver al sistema de referecia original.

3.2. CÓMPUTO DE LA TRANSFORMACIÓN

Una vez detectados los puntos SURF y determinadas sus correspondencias entre imágenes, ha de establecerse un proceso para recuperar los ángulos relativos β y ϕ.

3.2.1. Geometría Epipolar

La condición de epipolaridad establece la relación entre dos puntos 3D observados desde diferentes vistas.

$$p'^T E p = 0 \tag{4}$$

La matriz E representa la matriz esencial y puede ser calculada a partir de una serie de puntos correspondientes entre imágenes. El mismo punto detectado en dos imágenes se expresa como $p = [x, y, z]^T$ en el sistema de referencia de la primera cámara y $p^{'} = [x^{'}, y^{'}, z^{'}]^T$ en el de la segunda. En nuestro caso, debido al uso de una única cámara, las imágenes se adquieren desde dos posiciones indeterminadas, sin información de la distancia entre ellas. Este hecho implica el desconocimiento de la profundidad y por tanto la solución se obtiene salvo un factor de escala ρ. La matriz esencial E representa una rotación R y una traslación T (salvo un factor de escala) entre los sistemas de referencia de dos imágenes, con $E = R \cdot T_x$. Por tanto los ángulos deseados pueden ser obtenidos de la descomposición de E. Debe señalarse que la Geometría Epipolar puede ser usada en imágenes omnidireccionales ya que reproyectamos el sistema 2D del plano imagen a 3D mediante el modelado del espejo hiperbólico de la cámara, a partir de una calibración previa [17]. A causa de la ambigüedad en la profundidad, denotamos $\vec{p}$ and $\vec{p'}$ en 3D, como los vectores unitarios que indican la dirección de los puntos en los dos sistemas de referencia, ya que la posición 3D no puede ser totalmente definida con una única vista de la escena.

Con el propósito de obtener β y ϕ, hemos considerado [9], donde se sugiere el empleo de la matriz de proyección P, la cual también define la transformación entre imágenes. Se ha adoptado este método por su simplicidad a la hora de calcular las cuatro posibles soluciones del problema. Primero aplicamos la condición de epipolarodad $\vec{p'}^T \cdot E \cdot \vec{p} = 0$ sobre N puntos, y resolvemos la ecuación resultante $D \cdot E = 0$. A continuación descomponemos E mediante SVD:

$$[U|S|V] = SVD(E) \tag{5}$$

que permite calcular:

$$R_1 = [UV^T W],\ R_2 = [UV^T W^T],\ T = [UZU^T] \tag{6}$$

siendo W y Z matrices auxiliares [9] y las posibles rotaciones (R_1, R_2) y traslaciones $(T_{1x}, -T_{1x})$. Para obtener las cuatro posibles P-matrices, computamos:

$$P_1 = [R_1|T_{1x}],\ P_2 = [R_1| - T_{1x}], \tag{7}$$
$$P_3 = [R_2|T_{1x}],\ P_4 = [R_2| - T_{1x}], \tag{8}$$

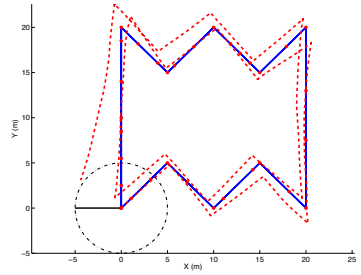
En nuestro caso, las matrices de proyección tienen la forma:

$$P_i = \begin{bmatrix} \cos(\beta) & -\sin(\beta) & 0 & \rho\cos(\phi) \\ \sin(\beta) & \cos(\beta & 0 & \rho\cos(\phi) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{9}$$

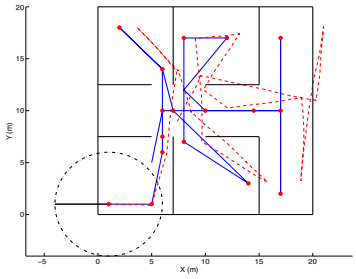
Notar que β, ϕ y ρ pueden tomar diferentes valores que cumplan la condición de epipolaridad (4) debido a la inderminación del factor de escala ρ. Esto plantea un proceso de selección de una de las cuatro posibles soluciones descritas en (7) como la correcta. En nuestro caso, hemos utilizado una solución basada en mínimos cuadrados, según se detalla en [3].

4. RESULTADOS

Presentamos tres conjuntos de experimentos diferenciados. En primer lugar, en el apartado 4.1 presentamos los resultados obtenidos en simulación que validan el esquema de SLAM aquí propuesto. A continuación, en el apartado 4.2 se presentan resultados a partir de datos reales adquiridos con un robot indoor Pioneer P3-AT. El robot está equipado con una cámara firewire 1280x960 y un espejo hiperbólico. El eje óptico de la cámara está instalado aproximadamente perpendicular al plano del suelo como se describe en la Figura 1(a). Como consecuencia, una rotación del robot se corresponde con una rotación de la imagen respecto a su punto central. A su vez, usamos un algoritmo SICK LMS basaso en distancias láser para calcular el camino real según [18].



(a)



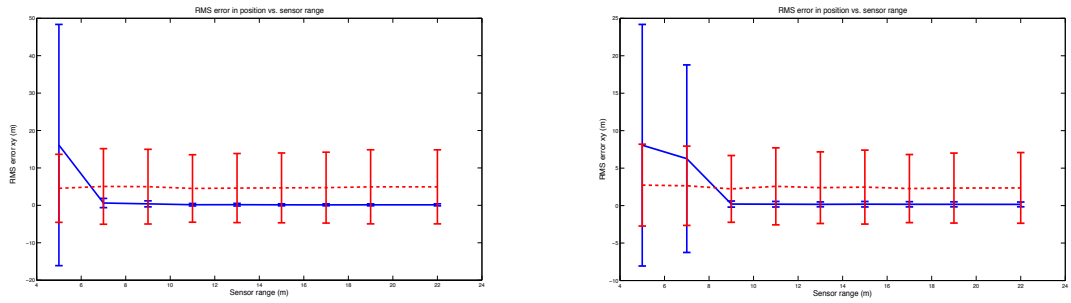
(b)

Figura 3: La Figura 3(a) representa el escenario simulado 1. La localización de las distintas vistas en el mapa se representa con puntos. La Figura 3(b) representa el escenario simulado 2.

4.1. SLAM. RESULTADOS EN SIMULACIÓN

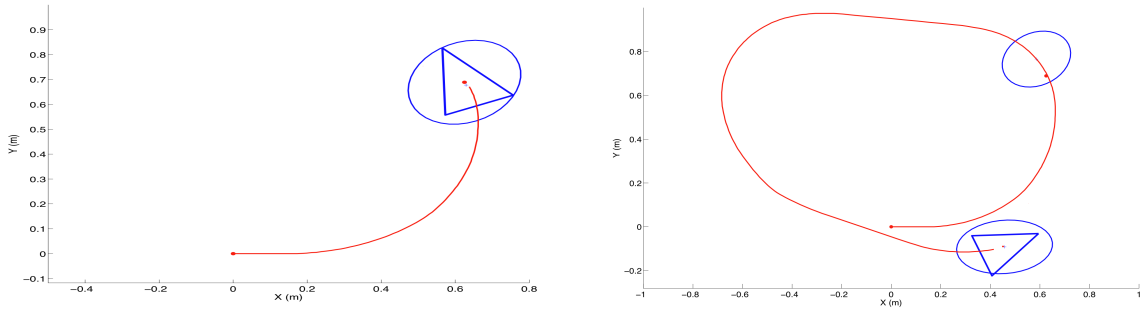
Hemos realizado una serie de experimentos en simulación a modo de validación del esquema propuesto de SLAM. Notar la importancia de asegurar la convergencia de un algoritmo de SLAM basado en EKF, lo cual no es trivial cuando se introduce un nuevo modelo de observación. La Figure 3(a) muestra el escenario simulado 1. La línea continua muestra el camino real mientras que la línea punteada muestra la odometría. Un conjunto de vistas han sido aleatoriamente dispuestas a lo largo de la trayectoria, se muestran con puntos. Notar que el emplazamiento de las vistas depende de la apariencia de las imágenes y del ratio R elegido. La variación R ha sido generada aleatoriamente con valores similares a los de un caso real.

El robot comienza el proceso de SLAM en el origen y realiza dos vueltas a lo largo de la trayectoria. El radio de observación del robot, es decir, la capacidad de procesar una observación $z_t = (\phi, \beta)^T$, viene dado por una distancia representada con línea a trazos. Las observaciones obtenidas por el robot han sido simuladas según el modelo presentado en la ecuación 2 con un AWGN de $\sigma_\phi = \sigma_\beta = 0{,}1 rad$. La Figura 3(b) representa el escenario simulado 2, el cual emula un entorno típico de interior donde existen obstrucciones, tales como paredes, que li-



(a) (b)

Figura 4: La Figura 4(a) presenta los resultados obtenidos en el escenario simulado 1. La Figura 4(b) presenta los resultados obtenidos en el escenario simulado 2.



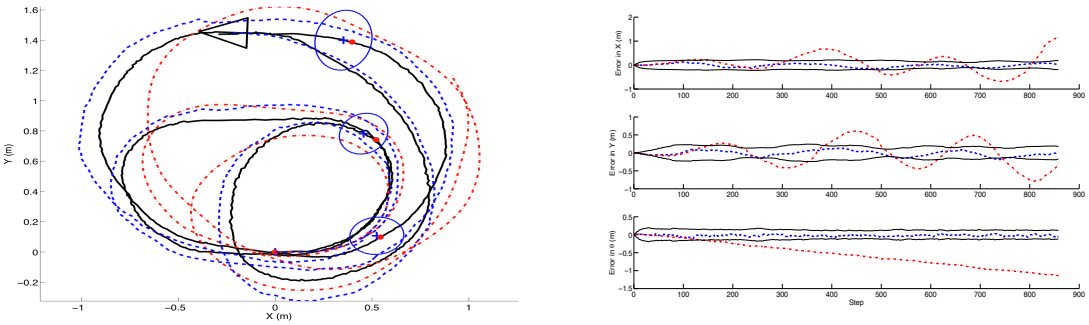
(a) (b)

Figura 5: La Figura 5(a) y 5(b) presentan los dos pasos iniciales en la construcción del mapa con datos reales, mostrado en la Figura 6.

mitan la observación. Hemos llevado a cabo una serie de experimentos donde se varía el radio de observación del robot. Los resultados se presentan en la Figura 4(a) y 4(b), donde se muestra el error RMS en la trayectoria frente al radio de observación. Comparamos la solución del EKF (línea contínua) con la odometría (línea a trazos). Hemos repetido el experimento con 50 simulaciones, generando aleatoriamente 50 series diferentes de odometría. En la Figure 4(a) se observa la media y los intervalos 2σ. Como se muestra en la Figure 4(a), cuando el radio está por debajo de $5m$ la incertidumbre en la posición aumenta y el filtro diverge. Para valores de radio superirores a $5m$ el error RMS disminuye manteniendo la convergencia del filtro. Un resultado similar se presenta en la Figure 4(b), la cual se corresponde con el escenario simulado 2. En este caso se obtienen resultados satisfactorios con valores de radio por encima de $9m$, debido a la mayor dificultad de convergencia del filtro cuando aparecen elementos obstructores en la visibilidad de las vistas, como son las paredes. Notar que los resultados dependen fuertemente del emplazamiento de las vistas, situando más vistas se consigue un cálculo más preciso tanto del mapa como de la trayectoria, sin embargo tiene asociado un mayor coste computacional.

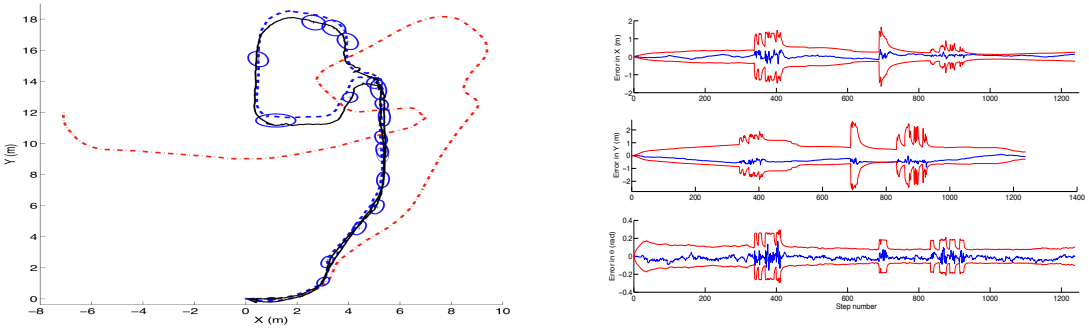
4.2. SLAM. RESULTADOS CON DATOS REALES

En este apartado presentamos resultados que validan el esquema de SLAM propuesto. El robot es guiado a través del entorno mientras captura imágenes omnidireccionales y datos de distancia láser a lo largo de la trayectoria. De nuevo, para poder comparar resultados, hacemos uso de un algoritmo de SLAM basado en distancias láser, descrito en [18], para definir la trayectoria real. El robot comienza inicializando una vista a partir de la adquisición de una imagen omnidireccional en el origen, como se indica en la Figura 5(a). A continuación se mueve a lo largo de la trayectoria mientras continúa adquiriendo imágenes. Instantes después se inicializa una nueva vista, como se indica en la Figura 5(a) con una elipse de error. Mientras se calcula el mapa, se realiza una comparación entre la imagen actual y el resto de vistas del mapa, obteniendo un conjunto de correspondencias. Al mismo tiempo, el ratio de similitud (3) es evaluado, y cuando éste cae por debajo de $\delta_R = 0{,}5$, se crea una nueva vista y se inicializa con la posición actual del robot. En la Figura 5(b) representamos la secuencia del proceso, donde una tercena vista es inicializada. Finalmente el robot



(a) (b)

Figura 6: La Figura 6(a) presenta los resultados de SLAM con datos reales, para trayectoria real (punteada), estimación (continua) y odometría (trazos). La posición de las vistas se presenta con elipses de error. La Figura 6(b) presenta el error en cada paso temporal en X, Y y θ de la estimación (punteada) y la odometría (trazos).



(a) (b)

Figura 7: La Figura 7(a) presenta los resultados de SLAM con datos reales, para trayectoria real (punteada), estimación (continua) y odometría (trazos). La posición de las vistas se presenta con elipses de error. La Figura 7(b) presenta el error en cada paso temporal en X, Y y θ de la estimación (punteada) con intervalos de 2σ.

recorre la trayectoria mostrada en la Figura 6(a), donde mostramos con puntos el resto de posiciones en las que el robot decide inicializar una nueva imagen. La línea punteada muestra la trayectoria real, la línea continua muestra la estimación del EKF, mientras que la línea a trazos muestra la odometría. Cabe señalar que el robot continúa el movimiento dentro de la misma estancia siendo capaz de realizar observaciones de las vistas inicializadas anteriormente. En nuestro caso el umbral δ_R fue determinado experimentalmente con el objetivo de generar un número reducido de vistas y representar el entorno de un modo más compacto. Si se eligiese un valor más bajo de δ_R, menos imágenes serían inicializadas en el mapa. Por el contrario si se eligiese un valor superior, el mapa resultante almacenaría un mayor número de vistas. Puede observarse en la Figura 6(a) como una vez la cuarta vista es inicializada no es necesario inicializar ninguna otra, obteniendo así una representación más compacta. En la Figura 6(b) comparamos la trayectoria estimada con la trayectoria real y con la odometría. Presentamos el error en la estimación de la trayectoria (línea punteada) junto a los intervalos 2σ y al error en la odometría (línea a trazos).

La Figura 7 presenta otro experimento. En este caso, el robot explora una habitación, recorre un pasillo, entra en una habitación diferente y vuelve al punto de origen. La distancia total recorrida es de $45m$. La Figura 7(a) presenta la trayecoria real (punteada), la odometría (a trazos) y la estimación (continua). La localización de las vistas y su incertidumbre asociada se indica mediante puntos y elipses de error. En la Figura 7(b) presentamos el error en la posición para cada paso temporal con intervalos de 2σ.

5. CONCLUSIONES

Hemos presentado un esquema de Localización y Construcción de Mapas Simultáneo (SLAM) empleando una cámara omnidireccional. Proponemos

una representación del entorno distinta. En lugar de estimar posiciones 3D de un conjunto de marcas visuales y sus descriptores, únicamente estimamos la posición y orientación de un conjunto de imágenes omnidireccionales. Cada imagen omnidireccional tiene asociado un conjunto de puntos de interés y sus descriptores visuales que describen el entorno de una forma compacta. Una imagen omnidireccional permite la localización del robot en los alrededores de ésta. Dadas dos imágenes omnidireccionales y un conjunto de correspondencias, somos capaces de obtener una rotación y una traslación (salvo un factor de escala) entre ambas. Esto nos permite proponer un nuevo modelo de observación y construir un mapa y una trayectoria. Presentamos resultados de localización y SLAM empleando un algoritmo de SLAM basado en un EKF, sin embargo, consideramos que pueden usarse diferentes estrategias de SLAM. Presentamos resultados obtenidos en entornos simulados que validan el esquema de SLAM, y a su vez, mostramos la validez del esquema ante datos reales adquiridos con un robot móvil.

Agradecimientos

Este trabajo se ha llevado a cabo gracias en parte al Ministerio de Ciencia e Innovación a través del proyecto DPI2010-15308, con título "Exploración integrada de entornos mediante robots cooperativos para la creación de mapas 3D visuales y topológicos que puedan ser usados en navegación con 6 grados de libertad"

Referencias

[1] Andrew J. Davison, A. J., Gonzalez Cid, Y., and Kita, N. Improving data association in vision-based SLAM. In *Proc. of IFAC/EURON* (Lisboa, Portugal, 2004).

[2] Bay, H., Tuytelaars, T., and Van Gool, L. SURF: Speeded up robust features. In *Proc. of the ECCV* (Graz, Austria, 2006).

[3] Bunschoten, R., and Krose, B. Visual odometry from an omnidirectional vision system. In *Proc. of the ICRA* (2003).

[4] Civera, J., Davison, A. J., and Martínez Montiel, J. M. Inverse depth parametrization for monocular slam. *IEEE Trans. on Robotics* (2008).

[5] Fernandez, L., Gil, A., Paya, L., and Reinoso, O. An evaluation of weighting methods for appearance-based monte carlo localization using omnidirectional images. In *Proc. of the ICRA* (Anchorage, Alaska, 2010).

[6] Gil, A., Martinez-Mozos, O., Ballesta, M., and Reinoso, O. A comparative evaluation of interest point detectors and local descriptors for visual slam. *Machine Vision and Applications* (2010).

[7] Gil, A., Reinoso, O., Martínez-Mozos, O., Stachniss, C., and Burgard, W. Improving data association in vision-based SLAM. In *Proc. of the IROS* (Beijing, China, 2006).

[8] Harris, C. G., and Stephens, M. A combined corner and edge detector. In *Proc. of Alvey Vision Conference* (Manchester, UK, 1988).

[9] Hartley, R., and Zisserman, A. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.

[10] Kawanishi, R., Yamashita, A., and Kaneko, T. Construction of 3D environment model from an omni-directional image sequence. In *Proc. of the Asia International Symposium on Mechatronics 2008* (Sapporo, Japan, 2008).

[11] Konolige, K., Bowman, J., Chen, J., and Mihelich, P. View-based maps. *IJRR* (2010).

[12] Lowe, D. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* (2004).

[13] Montemerlo, M., Thrun, S., Koller, D., and Wegbreit, B. Fastslam: a factored solution to the simultaneous localization and mapping problem. In *Proc. of the 18th national conference on Artificial Intelligence* (Edmonton, Canada, 2002).

[14] Murillo, A. C., Guerrero, J. J., and Sagüés, C. SURF features for efficient robot localization with omnidirectional images. In *Proc. of the ICRA* (San Diego, USA, 2007).

[15] Nister, D. An efficient solution to the five-point relative pose problem. In *Proc. of the IEEE CVPR* (Madison, USA, 2003).

[16] Scaramuzza, D., Fraundorfer, F., and Siegwart, R. Real-time monocular visual odometry for on-road vehicles with 1-point RANSAC. In *Proc. of the ICRA* (Kobe, Japan, 2009).

[17] Scaramuzza, D., Martinelli, A., and Siegwart, R. A toolbox for easily calibrating omnidirectional cameras. In *Proc. of the IROS* (Beijing, China, 2006).

[18] Stachniss, C., Grisetti, G., Haehnel, D., and Burgard, W. Improved Rao-Blackwellized mapping by adaptive sampling and active loop-closure. In *Proc. of the SOAVE* (Ilmenau, Germany, 2004).

[19] Stewenius, H., Engels, C., and Nister, D. Recent developments on direct relative orientation. *ISPRS Journal of Photogrammetry and Remote Sensing* (2006).