

# FUSIÓN DE MAPAS DE CARACTERÍSTICAS VISUALES EN SLAM MULTIROBOT

Mónica Ballesta, Óscar Reinoso, Miguel Juliá, Arturo Gil, Luis Payá  
Dept. Ingeniería de Sistemas Industriales  
Universidad Miguel Hernández  
Av. Universidad s/n, Edificio Quorum V, 03202 Elche  
{m.ballesta||o.reinoso||m.julia||arturo.gil||lpaya}@umh.es

## Resumen

*Este artículo se centra en el estudio de la fusión de mapas de características en SLAM visual. Concretamente, se plantea el caso de una plataforma multirobot en la que los robots realizan distintas trayectorias en un plano 2D, partiendo desde posiciones diferentes y sin tener conocimiento alguno de la posición y observaciones realizadas por el resto de robots. Cada robot extrae información visual 3D del entorno mediante un par estéreo y construye su mapa local con el algoritmo FastSLAM. En cierto instante del proceso de SLAM, los robots pueden fusionar sus mapas y construir un mapa global. Para ello, el primer paso es encontrar la transformación entre los sistemas de referencia de los robots (alineamiento 2D). A continuación, la información de los mapas se fusiona en un único mapa global (fusión). Las aportaciones principales de este artículo son la extensión del alineamiento al caso multirobot y la fusión de los mapas teniendo en cuenta la incertidumbre de las marcas.*

**Palabras clave:** SLAM multirobot, características visuales, FastSLAM, alineamiento, fusión de mapas.

## 1 INTRODUCCIÓN

Una de las habilidades fundamentales que ha de tener un robot autónomo es la capacidad para construir un mapa del entorno y navegar por él. Este concepto se conoce como *Simultaneous Localization and Mapping* (SLAM) y ha recibido gran interés en los últimos años [18, 26]. En la literatura existen difentes propuestas a la solución del problema de SLAM [23, 16]. En este artículo se utiliza el algoritmo FastSLAM [22]. La idea principal de este algoritmo, es la existencia de un conjunto de partículas que representan la incertidumbre de la pose del robot. A su vez, cada partícula tiene asociada una estimación del mapa del entorno. Separa, por tanto, el problema de SLAM en dos partes: la estimación de la pose del robot y la estimación del mapa. El algoritmo FastSLAM ha demostrado ser robusto ante errores en la aso-

ciación de datos y es capaz de representar modelos de movimientos no lineales [15].

En cuanto a los sensores utilizados, existen propuestas en las que se utilizan sensores láser [24, 25] o sonar [17, 20]. En este artículo se ha optado por el uso de cámaras estéreo para la construcción de mapas. Este enfoque se conoce como SLAM visual. El uso de este tipo de sensores viene siendo cada vez más frecuente dado que son capaces de obtener mayor cantidad de información del entorno y, además, son más económicas que el láser. Además, el hecho de utilizar visión estéreo, permite obtener información 3D del entorno.

En este caso, los robots construyen mapas basados en características, dado que es la solución más típica en un enfoque de SLAM visual [8]. Los mapas están formados por un conjunto de puntos distintivos del entorno. Estos puntos son *landmarks* referidas a un sistema de coordenadas global [12, 19] y caracterizados por un descriptor visual. Los mapas construidos en este artículo constan de las coordenadas 3D de puntos de Harris [14], caracterizados por el descriptor U-SURF [4] y una matriz de covarianza  $\Sigma$  asociada a cada punto, que representa la incertidumbre en la observación realizada por el robot. Este tipo de *landmarks* han demostrado tener un buen comportamiento en SLAM visual [21, 1, 11].

El problema de SLAM se puede llevar a cabo con un único robot. Sin embargo, se realizará de forma más eficiente si se dispone de un equipo de robots que cooperen en la consecución de esta tarea [15]. Este enfoque se denomina SLAM multirobot. Hasta la fecha, podemos encontrar diferentes propuestas que se dividen en dos grupos. Por un lado, existen soluciones en las que los robots construyen un mapa común. En este caso, la estimación de las trayectorias y la construcción del mapa se realizan de forma conjunta [13, 9, 24]. Esta opción tiene la ventaja de tener una noción global del entorno de modo que la exploración se puede llevar a cabo de forma eficiente. Sin embargo, requiere que la posición inicial de los robots sea conocida, lo cual no siempre es posible en la práctica. Por otro lado, otras soluciones proponen la construcción de mapas de forma independiente [27]. En este caso las

nuevas observaciones realizadas por los robots se comparan con un conjunto más reducido de *landmarks* en el mapa local. Además, no es necesario conocer la posición inicial de los robots.

Este artículo se centra en el último caso, es decir, la construcción de mapas de forma independiente. Inicialmente, los robots comienzan a construir sus mapas locales sin tener conocimiento de las observaciones y posiciones del resto de robots. En cierto instante, los mapas locales se pueden fusionar dando lugar a un único mapa global. Esto se conoce como el problema de la fusión de mapas [10]. En nuestro caso, se aborda el problema de fusión como el estudio de dos etapas: el alineamiento y la fusión de mapas. La primera etapa, consiste en obtener la transformación existente entre los mapas locales de forma que se puedan relacionar los diferentes sistemas de coordenadas. Esta parte se ha estudiado previamente en [2]. En este artículo, se ha ampliado el estudio del alineamiento considerando el caso en el que hayan más de dos robots. Para este caso se propone un método de optimización. Para la segunda etapa, es decir, la fusión de mapas, se propone un método que tenga en cuenta la incertidumbre de las marcas de los mapas locales. Estas son las dos principales aportaciones de este artículo.

El resto del artículo se estructura de la siguiente forma. En el apartado 2, se describe de forma más detallada los equipos y la metodología utilizada en este trabajo. En el apartado 3, se realiza un recordatorio del trabajo realizado en cuestión de alineamiento de mapas y se presenta la nueva aportación que considera el caso multirobot. A continuación, en el apartado 4, se detalla el método propuesto para realizar la fusión de los mapas locales. En el apartado 5 se presentan experimentos realizados con datos reales capturados por los robots y, finalmente, en 6, se presentan las principales conclusiones.

## 2 DETALLES DE IMPLEMENTACIÓN

Los robots utilizados en estos experimentos son Pioneer-P3AT provistos con un par estéreo STH-MDCS2 de *Videre Design* y un sensor láser SICK LMS 200. Los mapas de características se construyen únicamente a partir de la información capturada por el par estéreo. Las medidas del sensor láser se utilizan para obtener un *ground truth* de la trayectoria seguida por el robot.

Concretamente se utiliza un equipo de cuatro robots: R1, R2, R3 y R4. En la figura 1 se observan las trayectorias seguidas por los robots y una vista 2D del mapa de características global.

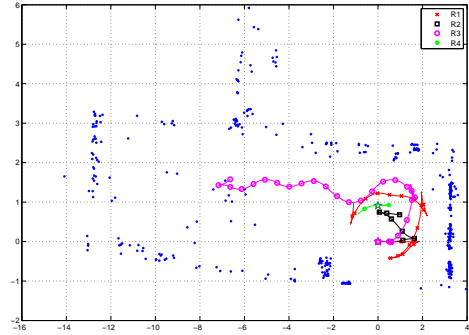


Figura 1: Trayectorias de los robots y vista 2D del mapa global.

Los robots parten de diferentes posiciones y realizan distintas trayectorias en un plano 2D. Concretamente en un edificio típico de oficinas. La información que captura cada robot es posteriormente procesada off-line con el algoritmo FastSLAM. Como resultado se obtiene un conjunto de mapas locales que pueden fusionarse en un único mapa. Los mapas construidos son basados en características o *landmarks*. Se considera como *landmark* las coordenadas 3D de los puntos de Harris [14] detectados en el entorno y un descriptor local U-SURF [5] asociado a cada punto. Los mapas están formados por estas *landmarks* y además, asociada a cada una de ellas, una matriz de covarianzas que representa la incertidumbre en las observaciones del robot. El hecho de utilizar este tipo de *landmarks* es el resultado de un estudio previo para encontrar el detector y el descriptor de características que fuesen más adecuados para SLAM visual en términos de repetibilidad y distinción [21, 1, 11].

Los experimentos realizados en este artículo han sido llevados a cabo entre pares de mapas, es decir, considerando dos robots. A excepción de la sección 3.1 donde se propone un método de optimización para alinear mapas de más de dos robots y se utilizan los cuatro robots de la figura 1.

## 3 ALINEAMIENTO

Se trata de la primera etapa del proceso de fusión. Calcular el alineamiento consiste en obtener la transformación que relaciona los sistemas de coordenadas de los robots. En nuestro caso, el sistema de referencia del robot tiene su origen en la posición inicial de dicho robot. De este modo, las marcas de los mapas visuales estarán referenciadas al mismo sistema de referencia y, en consecuencia, se pueden fusionar las marcas correspondientes.

En un trabajo previo, se realizó una evaluación de

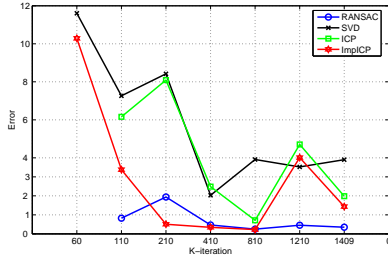


Figura 2: Evaluación de los métodos de alineamiento.

métodos de alineamiento con el fin de seleccionar aquel que obtuviese el alineamiento más preciso de mapas [3]. El método general seguido por los algoritmos evaluados es encontrar correspondencias entre las marcas en base a la similitud de los descriptores. Una vez obtenida esta lista de correspondencias, emplean diferentes técnicas para hallar los parámetros de alineamiento, consistentes en una traslación en  $x$  ( $t_x$ ), una traslación en  $y$  ( $t_y$ ) y una rotación ( $\theta$ ). La matriz de transformación sería entonces:

$$T = \begin{pmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ t_x & t_y & 0 & 1 \end{pmatrix} \quad (1)$$

Cabe destacar, que aunque los mapas visuales tengan características 3D, el alineamiento se produce en dos dimensiones, ya que el movimiento de los robots se produce en un plano  $xy$ . Como resultado de este estudio, se obtuvo que RANSAC es el método de alineamiento que obtiene un error más bajo en la estimación del alineamiento. En la figura 2, se observa el resultado obtenido en [3], donde se aprecia que RANSAC obtiene los resultados más precisos. Para obtener estas curvas, se realizó, con cada método, el alineamiento en distintas etapas del proceso de SLAM (eje  $x$ ), obteniendo en cada caso el error cometido en la obtención de los parámetros de alineamiento (eje  $y$ ). En las primeras iteraciones del algoritmo FastSLAM, los mapas tienen pocas marcas, con lo que la estimación del alineamiento es más inexacta. Sin embargo, a medida que el número de marcas es mayor, aumenta la probabilidad de encontrar correspondencias entre los mapas y se puede calcular el alineamiento con menos dificultad. Esto se observa en la tendencia decreciente de las curvas en la figura 2.

Todos estos resultados, se han obtenido comparando pares de mapas. Sin embargo, en un sistema multirobot, el número de robots existentes suele ser superior a dos. En este caso, no bastaría con obtener el alineamiento entre pares de robots,

sino que es necesario que el alineamiento entre los mapas locales sea consistente de forma global. En el siguiente apartado, se propone un método de optimización para este caso.

### 3.1 ALINEAMIENTO MULTIROBOT

Este apartado aborda la situación en la que existen más de dos robots ( $n > 2$ ,  $n$ : número de mapas) cuyos mapas queremos alinear. En este caso el alineamiento entre pares de mapas no es suficiente, ya que el alineamiento global ha de ser consistente. Para ello planteamos un sistema de ecuaciones que debemos minimizar para obtener la solución óptima.

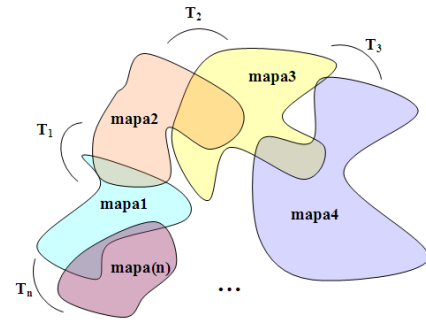


Figura 3: Alineamiento de múltiples mapas.  $T_i$  corresponde a la matriz de transformación entre los mapas  $i$  e  $i + 1$ .

En la figura 3, se muestra esta situación. Como se puede observar, existen  $n$  mapas alineados. La transformación entre cada par de mapas viene dada por la matriz de transformación definida en 1, obtenida con el algoritmo RANSAC. Cada par consecutivo de mapas tiene una parte en común y el conjunto de mapas forman un bucle. En esta situación, se debe de cumplir que:

$$T_1 \cdot T_2 \cdot \dots \cdot T_n = I \quad (2)$$

donde  $I$  es la matriz identidad. Desarrollando la ecuación 2, obtenemos tres expresiones que debemos minimizar:

- E1.  $\sin(\theta_1 + \dots + \theta_n)$
- E2.  $t_{x1} + t_{x2}\cos(\theta_1) + t_{y2}\sin(\theta_1) + t_{x3}\cos(\theta_1 + \theta_2) + t_{y3}\sin(\theta_1 + \theta_2) + \dots + t_{xn}\cos(\theta_1 + \dots + \theta_{n-1}) + t_{yn}\sin(\theta_1 + \dots + \theta_{n-1})$
- E3.  $t_{y1} + t_{x2}\sin(\theta_1) + t_{y2}\cos(\theta_1) - t_{x3}\sin(\theta_1 + \theta_2) + t_{y3}\cos(\theta_1 + \theta_2) + \dots - t_{xn}\sin(\theta_1 + \dots + \theta_{n-1}) + t_{yn}\cos(\theta_1 + \dots + \theta_{n-1})$

Por otro lado, para cada par de mapas que tienen correspondencias comunes, si se diese el caso ideal, tras el alineamiento se debería cumplir que:

$$L_{Aj\{m(k)\}} - L_{i\{m(k)\}} = 0 \quad (3)$$

donde el subíndice  $A$  denota las *landmarks* que han sido alineadas.  $m(k)$  es el número total de correspondencias entre el par de mapas  $k$  ( $k \in \{1, n\}$ ). En la práctica, la ecuación 3 no se cumple, pero la expresión de la izquierda se puede minimizar. Teniendo en cuenta todos los mapas, obtendremos  $2m(1) + 2m(2) + \dots + 2m(n)$  expresiones a minimizar:

$$\text{E}\delta. \quad x_2 \cos(\theta_1) + y_2 \sin(\theta_1) + t_{x1} - x_1 \quad \text{with } \delta \in \{4, X + 4\}$$

$$\text{E}\lambda. \quad y_2 \cos(\theta_1) - x_2 \sin(\theta_1) + t_{y1} - y_1 \quad \text{with } \lambda \in \{X + 5, 3X + 5\}$$

donde  $X = m(1) + m(2) + \dots + m(n)$ .

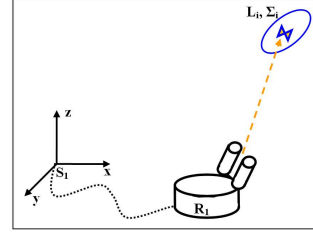
Por tanto, tendríamos un sistema no lineal de  $S = 3 + 2m(1) + 2m(2) + \dots + 2m(n)$  expresiones que debemos minimizar para obtener los nuevos parámetros de alineamiento. Para ello se ha utilizado la función **fsolve** de MATLAB. Se utiliza un algoritmo iterativo que usa un subespacio de la región de confianza basado en el método Newton descrito en [6, 7]. El punto inicial de este algoritmo es la estimación de los parámetros de alineamiento obtenida por RANSAC para cada par de mapas.

Tabla 1: Optimización de los parámetros de alineamiento en el caso multirobot.

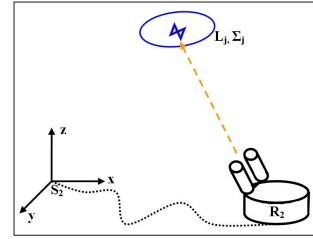
	$t_x$	$t_y$	$\theta$
$A_2^1$	-0.0676	-0.0636	-0.0144
$A_3^2$	0.1174	0.0423	-0.0063
$A_4^3$	-0.0386	0.8602	0.0286
$A_1^4$	0.0547	-0.8713	-0.0248
	$t'_x$	$t'_y$	$\theta'$
$A_2^1$	-0.0388	0.0363	0.0079
$A_3^2$	0.0677	-0.1209	-0.0375
$A_4^3$	-0.0408	0.9521	0.0534
$A_1^4$	0.0774	-0.9220	-0.0436

En la tabla 1 se presenta un ejemplo de los resultados obtenidos tras el proceso de optimización. Estos datos corresponden a los cuatro robots de la figura 1.  $A_j^i$  representa el alineamiento entre el robot  $i$  y el robot  $j$ . En la parte de arriba de la tabla, se observa los parámetros de alineamiento ( $t_x, t_y$  and  $\theta$ ) obtenidos por RANSAC entre cada par de mapas. Esta solución puede ser inconsistente desde el punto de vista global. En la parte

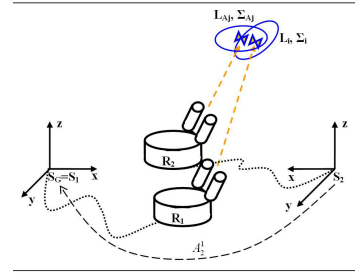
de abajo, se presenta la solución optimizada utilizando las expresiones  $E1, E2, \dots, E\lambda$  de este apartado.



(a)



(b)



(c)

Figura 4: (a) El robot  $R1$  observa la *landmark*  $L_i$  con matriz de covarianza  $\Sigma_i$ . El sistema de referencia es  $S_1$ . (b) EL robot  $R2$  observa la *landmark*  $L_j$  con matriz de covarianza  $\Sigma_j$ . El sistema de referencia es  $S_2$ . (c) Situación tras el alineamiento.  $L_i$  y  $L_j$  son *landmarks* correspondientes y el sistema de coordenadas  $S_2$  se ha transformado a  $S_{A2}$ .

## 4 FUSION

Tras el alineamiento de los mapas locales, las características de dichos mapas estarán expresadas en el mismo sistema de referencia. Llegados a este punto, es posible realizar la fusión de estos mapas locales y obtener un mapa global. En la figura 4, observamos el ejemplo de dos robots cuyos mapas se quieren fusionar. Por simplicidad, se muestra sólo una *landmark* del entorno. En las figuras,

4(a) y 4(b), vemos que los robots observan una característica de forma independiente. Esta observación es añadida a sus respectivos mapas como la *landmark*  $L_i$ ,  $L_j$  con sus respectivas incertidumbres representadas por las matrices de covarianzas  $\Sigma_i$ ,  $\Sigma_j$ . En la figura 4(c), observamos que las *landmark*  $L_i$  y  $L_j$  son correspondientes y el sistema de referencia del robot 2 se ha alineado con el del robot 1. En este momento, se puede realizar la fusión de las *landmarks* correspondientes entre ambos mapas. Es importante destacar que para la fusión se ha de tener en cuenta la incertidumbre asociada a cada *landmark*. Para ello en este artículo, se propone un filtro de Kalman Estacionario Multivariable, cuya formulación se presenta a continuación:

$$K_{\{i\}} = \Sigma_{1\{i\}} \cdot (\Sigma_{1\{i\}} + \Sigma_{2\{i\}})^{-1} \quad (4)$$

$$C_{G\{i\}} = C_{1\{i\}} + K_{\{i\}} \cdot (C_{1\{i\}} - C_{2\{i\}}) \quad (5)$$

$$\Sigma_{G\{i\}} = (I - K_{\{i\}}) \cdot \Sigma_{1\{i\}} \quad (6)$$

donde  $i$  es un índice ( $i \in \{1, M\}$ ,  $M$ : número total de correspondencias) que denota cada par de correspondencias entre los mapas (1 y 2). El subíndice  $G$  hace referencia a los datos del mapa global ( $mapa_G$ ) y los índices 1 y 2 al  $mapa_1$  y  $mapa_2$  respectivamente. En consecuencia,  $C_{G\{i\}}$  son las coordenadas 3D de la *landmark*  $i$  en el mapa global.  $C_2$  son las coordenadas 3D del  $mapa_2$  en el sistema de referencia del  $mapa_1$ . Finalmente,  $\Sigma_{G/1\{i\}}$  son las matrices de covarianza  $3 \times 3$ , que representan la incertidumbre de las marcas en  $mapa_G$ ,  $mapa_1$  y  $mapa_2$  respectivamente. Las matrices de covarianza del  $mapa_2$  ( $\Sigma_2$ ) se transforman también al sistema de referencia del  $mapa_1$ . En la figura 4, la incertidumbre de las marcas se representa con una elipse. Se observa, que en el alineamiento, no solo se transforma la posición de la *landmark*, sino también la elipse de error. Esto se realiza por medio de la matriz de rotación tal y como se muestra a continuación:

$$\Sigma_2 = R^T \cdot \Sigma_{20} \cdot R \quad (7)$$

donde  $\Sigma_{20}$  es la matriz de covarianza del  $mapa_2$  antes del alineamiento y  $R$  es la siguiente matriz de rotación:

$$R = \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (8)$$

En cuanto al descriptor asociado a cada *landmark*, tras el proceso de fusión, el nuevo descriptor de la *landmark* global será la media de los descriptores de las *landmarks* locales.

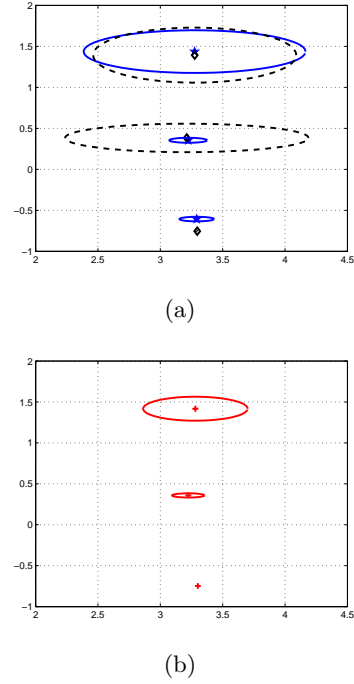


Figura 5: (a) Posición e incertidumbre de 3 *landmarks* alineadas pertenecientes a los mapas 1 y 2 de la figura 6. Las *landmarks* del  $mapa_1$  se representan con un punto en forma de estrella y una elipse de línea continua. Las del  $mapa_2$  se representan con un rombo y una elipse discontinua. (b) Las mismas *landmarks* después de la fusión. La posición de la nueva marca se representa con una cruz. La incertidumbre se representa con una elipse.

La figura 5 presenta un ejemplo la fusión teniendo en cuenta la incertidumbre de las *landmarks*. Concretamente, se muestra la posición e incertidumbre de tres *landmarks* correspondientes entre el  $mapa_1$  y el  $mapa_2$ , expresadas en el mismo sistema de referencia. En la figura 5(b), las *landmarks* se han fusionado mediante el Filtro de Kalman Estacionario Multivariable de las ecuaciones (4), (5) y (6). Es destacable que la incertidumbre de las marcas resultantes es menor que la de las marcas de los mapas locales. En algun caso la elipse de error no se aprecia dado su reducido tamaño.

## 5 EXPERIMENTOS

En los apartados 3 y 4 de este artículo se ha detallado el procedimiento para llevar a cabo el alineamiento y fusión de mapas de características visuales construidos mediante el algoritmo FastSLAM. En la figura 6, se muestra un ejemplo que resume todo el procedimiento. Concretamente, en la figura 6(a) se muestran dos mapas 3D construidos por los robots 1 y 2 respectivamente, a partir de

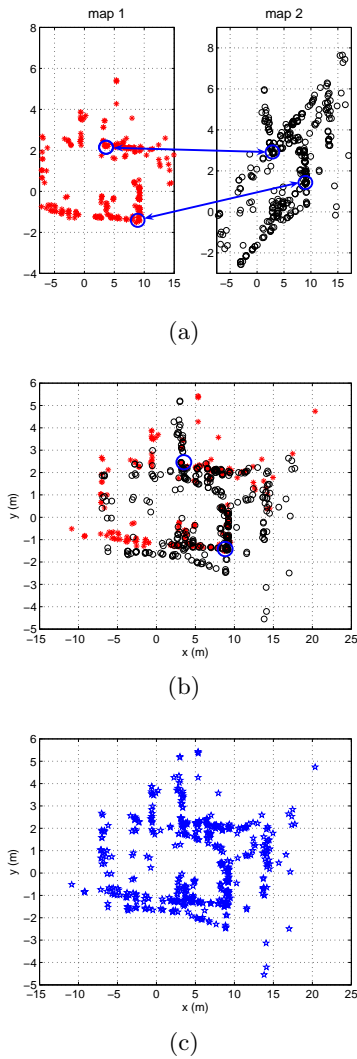


Figura 6: (a) Mapas locales  $mapa_1$  y  $mapa_2$ . (b) Se identifican marcas comunes y se alinean los mapas. Con círculos se destacan algunas marcas coincidentes entre los mapas. (c) Mapa fusionado resultante.

observaciones realizadas con sus cámaras estéreo. Para cada robot, se han utilizado 100 partículas en el algoritmo de FastSLAM. En la figura se muestra una vista 2D de dichos mapas. Inicialmente, estos mapas tienen sistemas de referencia diferentes. A continuación, según se aprecia en las figuras 6(a) y 6(b), se establecen correspondencias entre las *landmarks* en base a la similitud del descriptor y se alinean los mapas con el algoritmo de RANSAC [3]. En la 6(b) se muestra a los mapas alineados. Sin embargo, llegados a este punto, todavía no se ha conseguido un único mapa global. Esto se lleva a cabo con el Filtro de Kalman Estacionario Multivariable (ecs. (4), (5) y (6)). El resultado es el presentado en la figura 6(c). Los mapas que se muestran en la figura 6 corresponden en cada caso, al mapa de la partícula más

probable en el algoritmo de FastSLAM.

Sin embargo, sabemos que en un filtro de partículas, no tenemos un único mapa, sino  $B$  estimaciones de mapas correspondientes a las  $B$  partículas del filtro. Por este motivo, a continuación, se presentan unos experimentos en los que el procedimiento de alineamiento y fusión se engloban dentro del algoritmo FastSLAM. En este caso, no se realiza la fusión con las partículas más probables de cada robot sino que se lleva a cabo un proceso "partícula a partícula", teniendo en cuenta todas las partículas del filtro. De forma más detallada, se procede del siguiente modo. Sabemos que el algoritmo FastSLAM es un proceso iterativo. Supongamos que consta de  $k$  iteraciones y que en la  $k$ -ésima iteración se pretende realizar el alineamiento y fusión de los mapas de dos robots. En este instante, se fusionaría cada partícula del robot 1 con todas las partículas del robot 2. De modo que si cada robot tiene  $B$  partículas, al acabar el proceso, obtendríamos  $B \times B$  partículas fusionadas con sus mapas asociados. En este caso, se producen nuevas observaciones, y en base a ellas, se realiza un muestreo de partículas (*Importance Resampling*), de modo que nos volvamos a quedar con  $B$  partículas. A partir de ahí, el algoritmo FastSLAM seguiría con las sucesivas iteraciones.

Con el fin de evaluar este procedimiento, se plantea el siguiente experimento que consiste en comparar los resultados obtenidos cuando la frecuencia con la que se realiza la fusión varía. Concretamente, disponemos de los dos robots: robot 1 y robot 2, y en consecuencia de dos filtros de partículas F1 y F2, con  $B=50$  partículas cada uno. El algoritmo de FastSLAM realiza  $k=700$  iteraciones. El experimento consiste en realizar el alineamiento y fusión de los dos filtros de partículas en los siguientes casos:

1. Cada 700 iteraciones (al final del proceso).
2. Cada 300 iteraciones.
3. Cada 100 iteraciones.
4. Cada 50 iteraciones.

En cada caso, en la iteración  $k=700$ , se obtiene un mapa global (el mapa de la partícula más probable). Pues bien, lo que se ha hecho es identificar las marcas comunes entre los mapas obtenidos y comparar la incertidumbre de dichas marcas. Lo que se pretende es ver cómo evoluciona la incertidumbre del mapa obtenido como resultado de haber realizado el proceso de fusión con mayor o menor frecuencia durante el proceso de SLAM.

En la figura 7, se muestra el resultado obtenido con este experimento. En el eje de abscisas se pre-

senta las comparaciones realizadas, es decir, el hecho de realizar la fusión al final del proceso y realizarla cada  $k=300$  iteraciones (700 vs. 300), o realizarla cada  $k=100$  iteraciones (700 vs. 100) o cada  $k=50$  iteraciones (700 vs. 50). En cada caso, se han identificado las marcas comunes y se ha comparado la incertidumbre (matriz de covarianza) de dichas marcas. Las barras rojas indican el porcentaje de marcas cuya incertidumbre disminuye y las barras azules, las marcas cuya incertidumbre aumenta. Es decir, si comparamos el caso en el que la fusión se ha hecho en  $k=700$ , con el caso en el que se ha realizado cada  $k=300$ , obtenemos que un 31% de las marcas comunes entre ambos mapas tiene una incertidumbre mayor, mientras que el 69% de dichas marcas tiene una incertidumbre menor. De forma general, vemos que conforme aumenta el número de veces que se fusionan los mapas en el proceso de SLAM, el número de marcas en las que la incertidumbre disminuye, es mayor. Además, en las barras de la figura 7, se muestra en unos recuadros blancos, la media de la diferencia entre la incertidumbre de las marcas comparadas. Por tanto, además de que el número de marcas con menos incertidumbre es mayor al aumentar la frecuencia de fusión, se observa dicha incertidumbre disminuye en mayor medida que el caso en el que aumenta. Como conclusión, se puede deducir que el hecho de realizar la fusión de los mapas, da lugar a una estimación más exacta de los mismos. Sin embargo, se trata de una situación de compromiso, ya que realizar la fusión con demasiada frecuencia ralentizaría el proceso de SLAM.

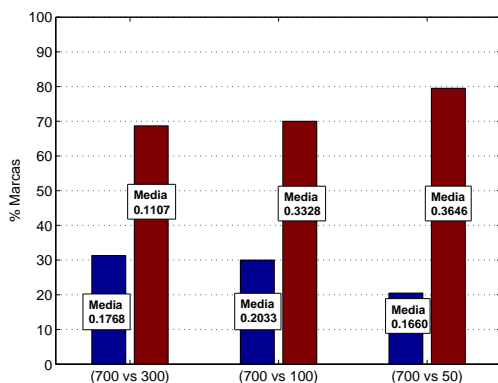


Figura 7: Evolución de la incertidumbre en las marcas variando la frecuencia con la que se fusionan los mapas.

## 6 CONCLUSIONES

El objetivo principal de este artículo ha sido el estudio de la fusión de mapas. Concretamente, el trabajo se ha llevado a cabo en un entorno

multirobot en el que los robots construyen sus mapas locales de forma independiente. La información es capturada del entorno mediante visión estéreo y los mapas construidos son mapas de características visuales 3D. El algoritmo de SLAM utilizado es FastSLAM.

El estudio de la fusión de mapas se ha llevado a cabo separando este problema en dos etapas: la fase de alineamiento y la fase de fusión. La primera fase fue tratada con detenimiento en trabajos anteriores. En este caso, se trata como novedad el caso de alineamiento de múltiples mapas. Para ello se ha presentado un método de optimización. La siguiente aportación, que es el objetivo principal de este artículo, ha sido la propuesta de un método de fusión basado en el filtro de Kalman. Este método tiene en cuenta la incertidumbre con la que cada robot ha incorporado las marcas en su mapa. Los resultados demuestran que las propuestas realizadas en este artículo son válidas y dan lugar a estimaciones precisas de las marcas del mapa fusionado.

## Agradecimientos

Esta investigación está patrocinada por el Ministerio de Ciencia e Innovación (referencia del proyecto: CICYT DPI2007-61107, título 'Sistemas de Percepción Visual Móvil y Cooperativo como Soporte para la Realización de Tareas con Redes de Robots') y por la Generalitat Valenciana con la beca BFPI/2007/096.

## Referencias

- [1] M. Ballesta, A. Gil, O. Martínez Mozos, and O. Reinoso. Local descriptors for visual SLAM. In *Workshop on Robotics and Mathematics (ROBOMAT07), Portugal*, 2007.
- [2] M. Ballesta, Ó. Reinoso, A. Gil, M. Juliá, and L. Payá. Métodos para el alineamiento de mapas de características visuales. 2008.
- [3] M. Ballesta, O. Reinoso, A. Gil, Juliá M., and L. Payá. Aligning methods for visual landmark-based maps. In *Recent Advances in Control Systems, Robotics and Automation 3rd Edition*, 2008.
- [4] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded-up robust features. In *Proc. of the 9th European Conference on Computer Vision*, 2006.
- [5] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: Speeded up robust features. In *European Conference on Computer Vision*, 2006.

- [6] T.F. Coleman and Y. Li. On the convergence of reflective newton methods for large-scale nonlinear minimization subject to bounds. In *Mathematical Programming, Vol.67, Number 2*, pp.189-224, 1994.
- [7] T.F. Coleman and Y. Li. An interior, trust region approach for nonlinear minimization subject to bounds. In *SIAM Journal on Optimization*, 1996.
- [8] Andrew J. Davison and David W. Murray. Simultaneous localisation and map-building using active vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002.
- [9] John W. Fenwick, Paul N. Newman, and John J. Leonard. Cooperative concurrent mapping and localization. In *Proc. of the 2002 IEEE International Conference on Intelligent Robotics and Automation*, pp.1810-1817, 2002.
- [10] D. Fox. Distributed multi-robot exploration and mapping. In *Proc. of the 2nd Canadian conference on Computer and Robot Vision*, 2005.
- [11] A. Gil, O. Martínez Mozos, M. Ballesta, and O. Reinoso. A comparative evaluation of interest point detectors and local descriptors for visual slam. *Accepted for publication in Machine Vision and Applications*, 2009.
- [12] A. Gil, O. Reinoso, W. Burgard, C. Stachniss, and O. Martínez Mozos. Improving data association in rao-blackwellized visual SLAM. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2006.
- [13] A. Gil, O. Reinoso, L. Payá, and M. Ballesta. Influencia de los parámetros de un filtro de partículas en la solución al problema de SLAM. In *IEEE Latin America*, 2007.
- [14] C. G. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, 1998.
- [15] A. Howard. Multi-robot simultaneous localization and mapping using particle filters. In *The International Journal of Robotics Research*, Vol. 25, No. 12, 1243-1256, 2006.
- [16] Patric Jensfelt, Danica Kragic, John Folkesson, and Márten Björkman. A framework for vision based bearing only 3D SLAM. In *IEEE Int. Conf. on Robotics & Automation*, 2006.
- [17] N. Kwak, G-W. Kim, S-H. Ji, and B-H. Lee. A mobile robot exploration strategy with low cost sonar and tungsten-halogen structural lighth. *Journal of Intelligent and Robotic Systems*, 51(1):89-111, 2008.
- [18] J.J. Leonard and H.F. Durrant-Whyte. Mobile robot localization by tracking geometric beacons. *IEEE Transactions on Robotics and Automation*, 7(4), 1991.
- [19] J. Little, S. Se, and D.G. Lowe. Global localization using distinctive visual features. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2002.
- [20] A. Mancini, A. Cesetti, A. Iuale, E. Frontoni, P. Zingaretti, and S. Longui. A framework for simulation and testing of uavs in cooperative scenarios. *Journal of Intelligent and Robotic Systems*, 54(1-3):307-329, 2009.
- [21] O. Martínez Mozos, A. Gil, M. Ballesta, and O. Reinoso. Interest point detectors for visual slam. In *Proc. of the XII Conference of the Spanish Association for Artificial Intelligence (CAEPIA), Salamanca, Spain*, 2007.
- [22] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. Fastslam: A factored solution to simultaneous localization and mapping. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*, pp. 593-598. *Edmonton, Canada*, 2002.
- [23] R. Sim, P. Elinas, M. Griffin, and J. Little. Vision-based slam using the rao-blackwellised particle filter. In *IJCAI Workshop on Reasoning with Uncertainty in Robotics (RUR)*, 2005.
- [24] S. Thrun. A probabilistic online mapping algorithm for teams of mobile robots. In *Int. Journal of Robotics Research*, 20(5), pp. 335-363, 2001.
- [25] R. Triebel and W. Burgard. Improving simultaneous mapping and localization. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*, 2005.
- [26] J. Valls Miro, W. Zhou, and G. Dissanayake. Towards vision based navigation in large indoor environments. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2006.
- [27] Xun S. Zhou and Sergios I. Roumeliotis. Multi-robot slam with unknown initial correspondence: The robot rendezvous case. In *Proc. of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China*, pp. 1785-1792, 2006.