

Segmentación robots móviles en espacios inteligentes utilizando técnicas GPCA y minimización de funciones de energía

Cristina Losada, Manuel Mazo, Sira Palazuelos, Francisco Redondo

Departamento de Electrónica. Universidad de Alcalá
Alcalá de Henares, Madrid, España
losada@depeca.uah.es

Abstract — En este trabajo se aborda el tema de la segmentación de objetos en movimiento en espacios inteligentes utilizando técnicas de Análisis de Componentes Principales Generalizado (GPCA), junto con métodos basados en la minimización de una función de energía. Para ello se propone, de forma previa a la segmentación, realizar la inicialización del conjunto de curvas que definirán los contornos de los objetos segmentados mediante GPCA. Posteriormente, se realiza la minimización de una función de energía que permite obtener la segmentación, los parámetros de movimiento 3D, y la posición de cada uno de los objetos en movimiento presentes en la escena. Para aumentar la robustez de los algoritmos se incorpora en la minimización la información disponible en el espacio inteligente. Esta propuesta ha sido implementada en un “espacio inteligente” dotado de múltiples cámaras y se han realizado diversas pruebas que han puesto de manifiesto la validez de la propuesta y la notable mejora que supone, en cuanto a la precisión de la segmentación y el tiempo de procesamiento, si se compara con otras propuestas en las que se utiliza únicamente la minimización de la función de energía para la segmentación de objetos en movimiento.

I. INTRODUCCIÓN

La segmentación de movimiento es una tarea fundamental para el análisis de secuencias de imágenes y consiste en agrupar la información presente en las mismas en conjuntos de píxeles cuyo movimiento en el plano imagen es coherente a lo largo de una secuencia. Todo ello sin tener un conocimiento previo acerca de qué píxeles de la imagen se mueven de acuerdo a un determinado modelo de movimiento. El análisis de movimiento en secuencias de imágenes digitales tiene un gran interés para una amplia gama de aplicaciones de la visión artificial, tales como robótica móvil, vigilancia, monitorización de tráfico, etc.

Existen en la literatura diferentes técnicas que tratan de analizar y extraer información acerca del movimiento de los objetos en una escena. Las primeras aproximaciones son las basadas en las discontinuidades del flujo óptico [1], [2]. Estas técnicas presentan inconvenientes derivados del problema de apertura, así como de la presencia de ruido en las estimaciones del flujo óptico. Por otro lado, la segmentación del movimiento ha estado tradicionalmente ligada con la detección de movimiento, en la que cada región corresponde a un modelo particular de movimiento que

explica los cambios temporales en dicha región de la imagen [3], [4]. También se han realizado diferentes propuestas para segmentación de movimiento basadas en técnicas de *clustering* [5], [6].

Finalmente, en [7] y [8] se propone obtener la segmentación de movimiento rígido 3D a partir de una función de energía obtenida en función de tres grupos de variables directamente relacionadas con el movimiento. Dicha función, tras la inicialización de los parámetros, se minimiza empleando un algoritmo iterativo en tres etapas que se repite hasta la convergencia. Como se muestra en [7] y [8], es posible realizar la segmentación de múltiples movimientos 3D independientes, sin embargo, aunque es posible obtener buenos resultados, este algoritmo presenta diversas limitaciones: en primer lugar, tanto los resultados obtenidos, como el número de iteraciones necesarias hasta la convergencia dependen fuertemente de la inicialización, así como de las constantes que ponderan la contribución de cada uno de los términos a la función de energía. Además, se trata de un algoritmo lento, ya que, según se indica en [8], el número de iteraciones necesarias hasta alcanzar la convergencia oscila entre 700 y 10.000.

En la sección II se describirá el sistema propuesto. A continuación, en los apartados III y IV se explicarán con mayor detalle las etapas de inicialización y minimización de la función de energía y, finalmente, en la sección V se presentarán los resultados obtenidos.

II. SISTEMA PROPUESTO

En este trabajo se propone un algoritmo robusto de segmentación de movimiento rígido 3D en espacios inteligentes, a partir de las imágenes captadas por una cámara fija en el entorno. Las etapas que contempla la solución propuesta se representan en la Fig. 1.

La medida del movimiento en cada par de imágenes consecutivas se realiza de forma indirecta mediante las derivadas parciales (espaciales y temporal).

Respecto a la segmentación y posicionamiento de los objetos móviles, el algoritmo utilizado está basado en la minimización de una función de energía (1) obtenida a partir de la restricción de brillo 3D según se describe en [7]

$$E[\{\gamma_k\}_{k=1}^{N-1}, \{\mathbf{T}_k\}_{k=1}^N, \{\boldsymbol{\omega}_k\}_{k=1}^N, \mathbf{Z}] = \sum_{k=1}^N \left[\int_{\Omega_k} \psi_k^2(\mathbf{x}) d\mathbf{x} + \mu \int_{\Omega_k} g(\|\nabla Z\|) d\mathbf{x} \right] + \sum_{k=1}^{N-1} \lambda \oint_{\gamma_k} ds \quad (1)$$

donde λ y μ constantes reales y positivas que ponderan la contribución de cada uno de los términos de la ecuación y $\nabla = (\partial_x, \partial_y)$ es el operador gradiente espacial.

Dado que esta función depende de tres grupos de parámetros (los contornos de la segmentación $\{\gamma_k\}_{k=1}^{n-1}$, las componentes de velocidad 3D, lineal y angular, en el sistema de referencia de la cámara $\{\mathbf{T}_k, \boldsymbol{\omega}_k\}_{k=1}^n$ y la profundidad \mathbf{Z}) para su minimización se emplea un algoritmo *greedy* formado por tres etapas que, tras la inicialización de las curvas y la profundidad, se repiten hasta la convergencia del algoritmo.

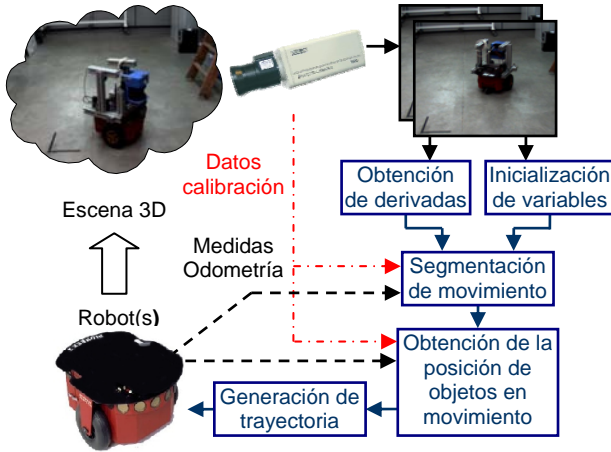


Fig. 1. Diagrama de bloques del sistema propuesto

Para salvar las limitaciones inherentes a esta técnica de segmentación, que han sido comentadas en la introducción, en este trabajo se propone realizar la inicialización de las curvas usando la técnica matemática GPCA, así como incluir la información disponible en el espacio inteligente tanto en la inicialización como en las etapas de segmentación y posicionamiento 3D.

Finalmente, tras el posicionamiento, es posible determinar la trayectoria a seguir por cada uno de los robots y enviar los comandos de guiado.

III. INICIALIZACIÓN

La correcta inicialización de las curvas que definirán los contornos de los diferentes objetos segmentados influye de forma notable tanto en los resultados de la segmentación, como en el tiempo de cómputo del algoritmo ya que cuanto más próximas estén las curvas iniciales a los contornos de los objetos en movimiento, menor será el número de iteraciones necesario hasta alcanzar la convergencia.

En este trabajo en el que la segmentación se realiza a partir de las imágenes tomadas por una cámara fija en un entorno interior se propone realizar un modelado del fondo de la escena a partir de un conjunto de imágenes del mismo y, posteriormente comparar cada par de imágenes con dicho modelo. Tanto para la obtención del modelo de fondo como para la comparación de las imágenes de entrada con el modelo obtenido se emplea la técnica GPCA (*Generalized Principal Components Analysis*) [9].

En los siguientes apartados se describen las diferentes etapas en las que se divide el proceso de inicialización de las curvas y la profundidad.

A. Inicialización de las curvas. Etapa off-line: Obtención del modelo de fondo

En la primera etapa del algoritmo, usando la técnica GPCA [9] se obtienen dos matrices de transformación \mathbf{L} y \mathbf{R} a partir de un conjunto de imágenes del fondo $\{\mathbf{I}_j\}_{j=1}^N$. Estas matrices permiten proyectar cualquier imagen de fondo al espacio transformado, de dimensión inferior a la del espacio de entrada, manteniendo las características principales de dicha imagen.

Las matrices que definen el modelo de fondo se obtienen a partir de un conjunto de imágenes a las que se les ha restado previamente la imagen media (2)

$$\tilde{\mathbf{I}}_j = \mathbf{I}_j - \frac{1}{N} \sum_{i=1}^N \mathbf{I}_i \quad j = 1..N \quad (2)$$

Estas imágenes a las que se les ha sustraído la media son las que se utilizarán para la obtención del modelo de fondo mediante el proceso iterativo descrito en [9] que finaliza cuando se la variación del $RMSE$ (3) entre una iteración y la anterior es inferior a un umbral fijado previamente (η). La convergencia de este proceso está garantizada por el teorema 4.2 definido en [9].

$$RMSE(i) = \sqrt{\frac{1}{n} \sum_{j=1}^N \|\tilde{\mathbf{I}}_j - \mathbf{L}_i \mathbf{L}_i^T \tilde{\mathbf{I}}_j \mathbf{R}_i^T\|_F^2} \quad (3)$$

Un parámetro fundamental es el número de autovectores utilizados para obtener las matrices \mathbf{M}_R y \mathbf{M}_L (4) y (5).

$$\mathbf{M}_R = \sum_{j=1}^N \tilde{\mathbf{I}}_j^T \mathbf{L}_i \mathbf{L}_i^T \tilde{\mathbf{I}}_j \quad (4)$$

$$\mathbf{M}_L = \sum_{j=1}^N \tilde{\mathbf{I}}_j^T \mathbf{R}_i \mathbf{R}_i^T \tilde{\mathbf{I}}_j \quad (5)$$

donde \mathbf{L}_i y \mathbf{R}_i se actualizan en cada iteración del algoritmo, y dependen de los autovectores de las matrices \mathbf{M}_R y \mathbf{M}_L , ($\{\phi_j^R\}_{j=1}^d$ y $\{\phi_j^L\}_{j=1}^d$) obtenidas en la iteración anterior:

$$\mathbf{R}_i = [\phi_1^R, \dots, \phi_d^R] \quad (6)$$

$$\mathbf{L}_i = [\phi_1^L, \dots, \phi_d^L] \quad (7)$$

Tras alcanzar la convergencia se obtienen las matrices de transformación como las matrices $\mathbf{L} = \mathbf{L}_i$ y $\mathbf{R} = \mathbf{R}_i$ siendo i la iteración en que el algoritmo converge.

B. Inicialización de las curvas. Etapa on-line. Obtención de las curvas iniciales

En esta segunda etapa (Fig. 2) se compara el modelo de fondo obtenido con cada par de imágenes que se quiere segmentar para, de esta forma, determinar qué píxeles son candidatos a pertenecer a objetos que han entrado en la escena después de haber tomado las imágenes del fondo. Estos elementos pueden ser tanto robots móviles controlados por el espacio donde se mueven, como objetos con movimiento no controlado, o incluso nuevos elementos fijos, que se han introducido en la escena después de captar las imágenes utilizadas para obtener el modelo de fondo y brillos o reflejos debidos a cambios en la iluminación.

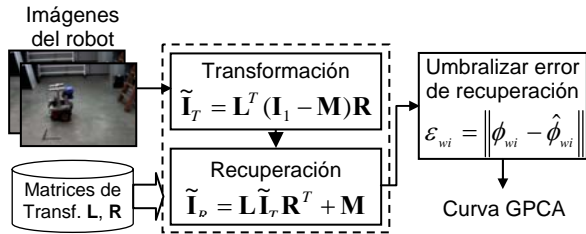


Fig. 2. Etapa on-line. Inicialización de curvas usando GPCA

Para la comparación de las nuevas imágenes con el modelo de fondo en primer lugar se proyecta la imagen al espacio transformado definido por las matrices \mathbf{L} y \mathbf{R} obtenidas anteriormente (8) y a continuación se realiza la recuperación de la imagen (9):

$$\mathbf{I}_T = \mathbf{L}^T (\mathbf{I} - \mathbf{M}) \mathbf{R} \quad (8)$$

$$\mathbf{I}_R = \mathbf{L} \mathbf{I}_T \mathbf{R}^T + \mathbf{M} \quad (9)$$

El error de recuperación se define como la diferencia entre la imagen original y la recuperada, y puede obtenerse de forma directa restando la imagen recuperada de la original de forma que el error de recuperación en el píxel de coordenadas (w, i) se expresa empleando la expresión (10) donde \mathbf{I}_{wi} es el valor del píxel en la imagen original e $\hat{\mathbf{I}}_{wi}$ su valor en la imagen recuperada.

$$\varepsilon_{wi} = \|\mathbf{I}_{wi} - \hat{\mathbf{I}}_{wi}\| \quad (10)$$

Sin embargo, esto es poco robusto al ruido, por ello se define una ventana en torno a cada píxel de la imagen y se obtiene el error de recuperación para esa ventana. Se definen ventanas cuadradas de $q \times q$ píxeles y se identifican por Φ_{wi} y $\hat{\Phi}_{wi}$ para la imagen original y

recuperada respectivamente. El error de recuperación asociado al píxel central de la ventana y cuyas coordenadas son (w, i) se determina como:

$$\varepsilon_{wi} = \|\Phi_{wi} - \hat{\Phi}_{wi}\| \quad (11)$$

Los píxeles candidatos a pertenecer a objetos en movimiento (alrededor de los que se inicializarán las curvas) serán aquellos en los que el valor de recuperación obtenido supere un determinado umbral, ya que esto indicará que, en esos píxeles, existe una diferencia importante con respecto a las imágenes del fondo utilizadas para la obtención del modelo.

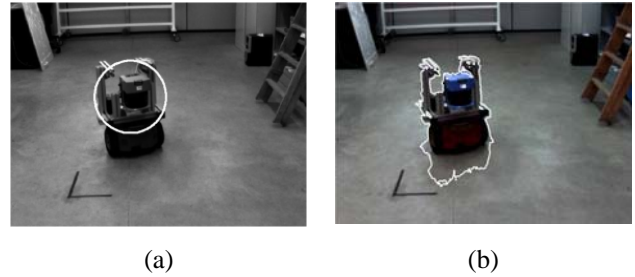


Fig. 3. Curvas iniciales (a) circunferencia (b) curva obtenida usando GPCA

En la Fig. 3 se muestra la diferencia entre la inicialización con circunferencias [7] y la curva obtenida usando el algoritmo basado en GPCA propuesto en este trabajo. Se puede observar cómo en la figura de la derecha, la curva se aproxima al contorno del robot. Esto hace posible obtener la segmentación y las velocidades en un número de iteraciones mucho menor. Sin embargo, en la figura también se puede observar cómo la curva obtenida con GPCA incluye la sombra que proyecta el robot sobre el suelo. Esto puede provocar problemas en la segmentación, debido a que la sombra se desplaza a la misma velocidad que el robot, de forma que no es sencillo eliminarla en la etapa de segmentación de movimiento. Por este motivo, se ha incluido en la inicialización un proceso de eliminación de sombras que se describe a continuación.

C. Eliminación de sombras

El proceso de eliminación de sombras implementado está basado en la técnica propuesta en [10]. Sin embargo, en lugar de dividir dos de las componentes de color, entre la tercera, para evitar que los resultados aparezcan sesgados hacia alguno de los canales de color, la imagen se normaliza dividiendo cada una de las componentes R, G, B, entre la media geométrica de las tres para, a continuación, tomar logaritmos (12) y proyectar la imagen a un espacio 2D, utilizando PCA para obtener la matriz \mathbf{U} (13) que permite la transformación del espacio original 3D ρ a un espacio 2D χ .

$$\rho_k = \log\left(\frac{R_k}{\sqrt[3]{\prod_{i=1}^3 R_i}}\right) \quad k = R, G, B \quad (12)$$

$$\boldsymbol{\chi} = \mathbf{U} \cdot \boldsymbol{\rho} \quad (13)$$

Finalmente, mediante la ecuación (14) es posible proyectar $\boldsymbol{\chi}$ a una imagen invariante en escala de grises. Debido a que no se ha realizado la calibración en color de las cámaras antes de captar las imágenes, el ángulo θ de proyección al espacio invariante a la iluminación se obtiene como aquel que minimiza la entropía del histograma de la imagen proyectada [11].

$$I_{inv} = \chi_1 \cos \theta + \chi_2 \sin \theta \quad (14)$$

Una vez obtenida la imagen invariante se obtiene la norma del gradiente tanto de la imagen en entrada en escala de grises, como de la imagen invariante (14). Se consideran bordes de sombras los píxeles en los que la norma del gradiente de la imagen original supera un umbral dado, mientras que la norma del gradiente de la imagen invariante es inferior a un segundo umbral. De esta forma se define la función de umbral (15). Ambos umbrales (U_g y U_{inv}) se han obtenido de forma experimental.

$$S = \begin{cases} 1 & \|\nabla \mathbf{I}_g(x, y)\| > U_g, \|\nabla \mathbf{I}_{inv}(x, y)\| < U_{inv} \\ 0 & \text{resto} \end{cases} \quad (15)$$

En la Fig. 4 la curva resultante tras aplicar el proceso de eliminación de sombras a la curva inicial obtenida con GPCA (Fig. 3(b))

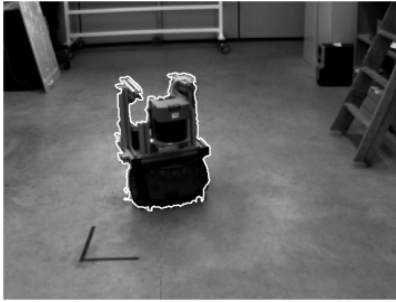


Fig. 4. Curva inicial tras la eliminación de la sombra

D. Inicialización de la profundidad

En la etapa de inicialización, también es necesario asignar un valor a la profundidad (coordenada Z en el sistema de referencia de la cámara). En nuestro caso se ha decidido emplear las medidas de odometría del robot en caso de que estén disponibles con objeto de que el valor inicial de la profundidad sea lo más próximo posible al valor final de la misma. De esta forma, por cada par de imágenes de la secuencia, se obtiene un valor de la profundidad a partir de las medidas disponibles en el espacio inteligente inicializándose la

profundidad como un plano a la distancia a la que se encuentra el fondo en la parte exterior a las curvas obtenidas en la inicialización, y un segundo plano a la distancia calculada para los puntos interiores de la curva. En la Fig. 5 se muestra la Z inicial para una imagen de la secuencia.

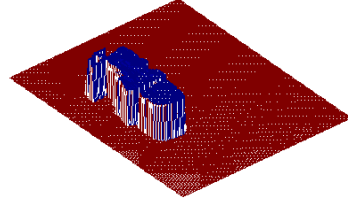


Fig. 5. Profundidad inicial

IV. MINIMIZACIÓN DE LA FUNCIÓN DE ENERGÍA

Tras la inicialización de las curvas y la profundidad, la minimización de la función de energía (1) se realiza en tres pasos que se repiten hasta alcanzar la convergencia. Estas etapas se describen brevemente a continuación:

En la primera etapa, se suponen conocidas las curvas y la profundidad, de forma que la función de energía a minimizar se reduce a (16):

$$E(\{\mathbf{T}_k\}_{k=1}^N, \{\boldsymbol{\omega}_k\}_{k=1}^N) = \sum_{k=1}^N \int_{\Omega_k} \psi_k^2(\mathbf{x}) d\mathbf{x} \quad (16)$$

donde $\boldsymbol{\Psi}_k = \mathbf{I}_t + \mathbf{s} \cdot \frac{\mathbf{T}_k}{Z} + \mathbf{q} \cdot \boldsymbol{\omega}_k$ y \mathbf{s} y \mathbf{q} son los siguientes vectores que dependen de la distancia focal de la cámara f y las derivadas parciales ($\mathbf{I}_x, \mathbf{I}_y$)

$$\mathbf{s} = \begin{pmatrix} f \mathbf{I}_x \\ f \mathbf{I}_y \\ -x \mathbf{I}_x - y \mathbf{I}_y \end{pmatrix} \quad \mathbf{q} = \begin{pmatrix} -f \mathbf{I}_y - \frac{y}{f} (x \mathbf{I}_x + y \mathbf{I}_y) \\ -f \mathbf{I}_x - \frac{x}{f} (x \mathbf{I}_x + y \mathbf{I}_y) \\ -y \mathbf{I}_x + x \mathbf{I}_y \end{pmatrix}$$

En [7], teniendo en cuenta que ψ_k depende linealmente de \mathbf{T}_k y $\boldsymbol{\omega}_k$, la minimización se realiza resolviendo un sistema de ecuaciones lineales mediante mínimos cuadrados, sin embargo, las pruebas realizadas han puesto de manifiesto que, dado que el número de muestras dentro de cada región es muy elevado para las imágenes utilizadas, los resultados obtenidos con esta técnica distan en gran medida de los esperados, introduciendo un error que afecta al resto de etapas del algoritmo. Por este motivo, en este trabajo se propone obtener los parámetros de movimiento en cada región, como aquellos que minimicen el error cuadrático (17) para cada una de las N regiones en las que las $N-1$ curvas dividen la imagen.

$$\varepsilon_k^2(\mathbf{T}_k, \boldsymbol{\omega}_k) = \sum_{\mathbf{x} \in \Omega_k} \left(I_t + \mathbf{s}(\mathbf{x}) \cdot \frac{\mathbf{T}_k}{Z(\mathbf{x})} + \mathbf{q}(\mathbf{x}) \cdot \boldsymbol{\omega}_k(\mathbf{x}) \right)^2 \quad (17)$$

En la segunda etapa la función a minimizar para recuperar la profundidad es la mostrada en (18) donde χ_k es la función característica de la región Ω_k , a partir de la que se actualiza el valor de la profundidad mediante una iteración de descenso por el gradiente.

$$E(Z) = \int_{\Omega} \sum_{k=1}^N [\chi_k(\mathbf{x})(\psi_k^2(\mathbf{x}) + \mu g(\|\nabla Z\|))] d\mathbf{x} \quad (18)$$

En la tercera y última etapa, dada la profundidad y los parámetros del movimiento rígido 3D, la función de energía a minimizar con respecto a las curvas es:

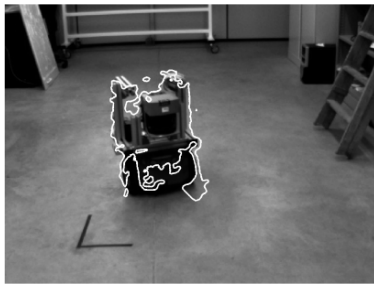
$$E[\{\gamma_k\}_{k=1}^{N-1}] = \sum_{k=1}^N \int_{\Omega_k} \xi_k(\mathbf{x}) d\mathbf{x} + \lambda \sum_{k=1}^{N-1} \oint_{\gamma_k} ds \quad (19)$$

donde $\xi_k(\mathbf{x}) = \psi_k^2(\mathbf{x}) + \mu g(\|\nabla Z\|)$. A partir de esta ecuación es posible actualizar las curvas mediante una iteración del descenso por curvas de nivel.

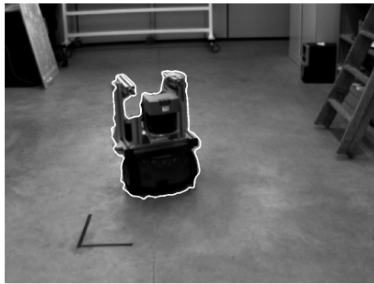
El proceso de minimización finaliza cuando se alcanza la convergencia, obteniéndose tanto los parámetros de movimiento 3D, como las curvas que definen la segmentación, y la posición 3D del objeto. Ya que, conocida la profundidad, es posible proyectar los puntos del plano imagen, al sistema de referencia 3D de la cámara.

V. RESULTADOS

Para comprobar las prestaciones del algoritmo propuesto se han realizado diversos experimentos. Todas las pruebas han sido realizadas utilizando un equipo Intel Core 2 CPU 6600 a 2.40GHz con 3.50 Gb de memoria RAM.



(a)



(b)

Fig. 6. Resultado de la segmentación (a) 100 iteraciones, inicialización con circunferencias (b) 3 iteraciones, inicialización usando GPCA

En las figuras 3 y 4 se han presentado dos posibles curvas para inicializar el algoritmo. A continuación se muestra el resultado de la segmentación en cada uno de los casos. En la Fig. 6(a) se puede observar cómo tras 100 iteraciones, si la inicialización se hace con una circunferencia, aún no se ha conseguido la segmentación del robot completo mientras que, en la Fig. 6(b) se muestra cómo tres iteraciones son suficientes para obtener la segmentación al inicializar usando la técnica propuesta.

Por otro lado, en la Fig. 7 se ha representado el tiempo de ejecución para cada una de las dos alternativas de inicialización, al segmentar diferentes pares de imágenes de entrada, representándose en el eje X la posición que ocupa la primera de las dos imágenes utilizadas en la segmentación, dentro de la secuencia de imágenes, mientras que en el eje Y se muestra el valor del tiempo de ejecución en segundos. En esta figura se puede observar cómo el tiempo de cómputo necesario para la segmentación se reduce de forma significativa al realizar la inicialización utilizando la técnica basada en GPCA que se ha propuesto en este trabajo, frente a la inicialización con circunferencias propuesta en [7].

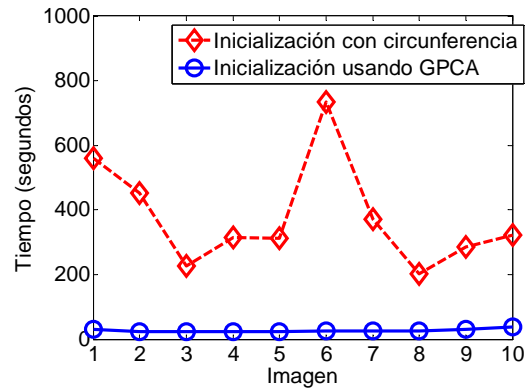


Fig. 7. Tiempo de procesamiento para la segmentación de 10 pares de imágenes.

A la vista de las figuras anteriores se puede decir que el hecho de realizar la inicialización de las curvas utilizando la técnica GPCA provoca una mejora, tanto en la precisión de la segmentación, como en el tiempo de cómputo necesario.

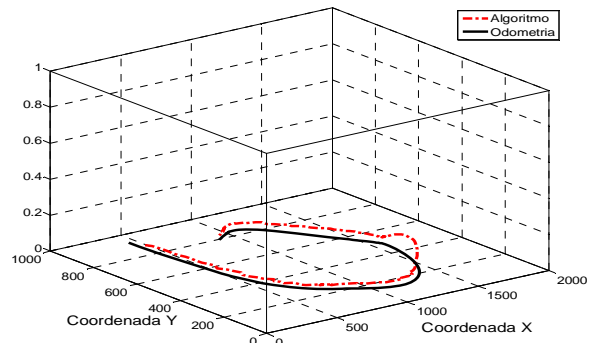


Fig. 8. Trayectoria seguida por el robot.

El algoritmo propuesto también permite recuperar la posición 3D de los objetos en movimiento. En la Fig. 8 se presenta la trayectoria obtenida a partir de las medidas de odometría (en línea continua negra) y mediante el algoritmo propuesto (línea discontinua roja). Tanto la coordenada X como la Y se muestran en milímetros.

Finalmente, la Fig. 9 muestra la diferencia en milímetros de la coordenada X e Y entre las medidas de la odometría del robot, y la estimación realizada usando el algoritmo propuesto para una secuencia de 50 pares de imágenes consecutivas capturadas en el espacio inteligente. Se puede observar cómo dicha diferencia no supera en ningún caso los 10 centímetros.

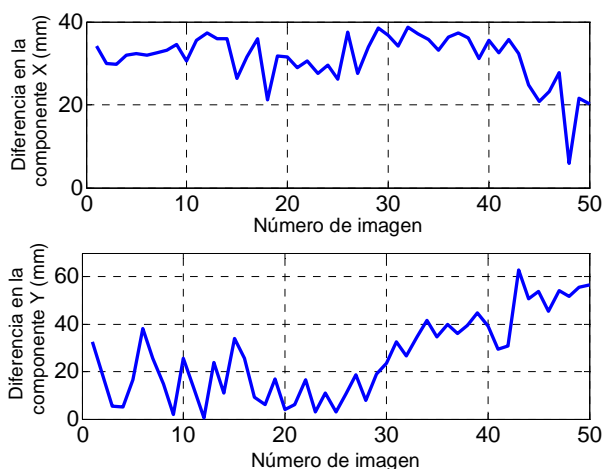


Fig. 9. Diferencia entre la posición 3D medida por los sensores odométricos del robot y la estimada por el algoritmo propuesto

VI. CONCLUSIONES

En este trabajo se ha presentado un sistema que permite la segmentación de los diferentes objetos en movimiento presentes en una secuencia de imágenes utilizando un algoritmo iterativo en tres etapas que se repiten hasta la convergencia.

Tanto el tiempo de cómputo del algoritmo, como los resultados finales del mismo dependen fuertemente de la inicialización realizada. En este trabajo se ha propuesto realizar la inicialización de las curvas mediante la comparación de cada par de imágenes de la secuencia con un modelo del fondo utilizando la técnica matemática GPCA, así como una técnica de eliminación de sombras. Por otro lado, también se ha propuesto utilizar la información disponible en el espacio inteligente (medidas de los sensores odométricos de los

robots) para inicializar la profundidad y los parámetros de movimiento 3D.

En las pruebas realizadas se ha puesto de manifiesto que la correcta inicialización de las curvas y la profundidad mejora notablemente el tiempo de cómputo, así como los resultados finales de la segmentación.

AGRADECIMIENTOS

Este trabajo ha sido posible gracias a la financiación del Ministerio de Educación y Ciencia (MEC) a través del proyecto RESELAI (REF-TIN2006-14896-C02-01).

REFERENCIAS

- [1] Black, M. and Anandan, P. "Robust dynamic motion estimation over time". Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition. June 1991. Page(s): 296-302.
- [2] Yan, H.; Tjahjadi, T.; "Multiple motion segmentation through a highly robust estimator". IEEE International Conference on Systems, Man and Cybernetics Oct. 2004. Page(s): 3082-3087.
- [3] Darrell, T.; Pentland, A. "Robust estimation of a multi-layered motion representation". Proc. of the IEEE Workshop on Visual Motion. 1991. Page(s): 173-178.
- [4] Weiss, Y. "A unified mixture framework for motion segmentation: incorporating spatial coherence and estimating the number of models". Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition. 1996. Page(s): 321-326
- [5] Costeira, J. and Kanade, T. "A multibody factorization method for independently moving objects". International Journal of Computer Vision. Volume 29, Number 3. September 1998. Page(s):159-179.
- [6] Kanatani, K. "Motion Segmentation by Subspace Separation and Model Selection". Proceedings of the 8th IEEE International Conference on Computer Vision 2001. Volume 2, Page(s): 586-591.
- [7] H. Sekkati and A. Mitiche. "Concurrent 3D Motion Segmentation and 3D Interpretation of Temporal Sequences of Monocular Images". IEEE Transactions on Image Processing, Vol. 15, No. 3, Page(s): 641-653, 2006.
- [8] H. Sekkati and A. Mitiche. "Joint Optical Flow Estimation, Segmentation, and Interpretation with Level Sets". Computer Vision and Image Understanding, Vol. 103, No. 2, Page(s): 89-100, 2006.
- [9] Ye, Jieping; Janardan, Ravi; Li, Qi. "GPCA: an efficient dimension reduction scheme for image compression and retrieval". Proceedings of the 10th ACM SIGKDD international conference on Knowledge discovery and data mining. 2004. Page(s): 354-363
- [10] G.D. Finlayson, S.D. Hordley, and M.S. Drew. "Removing shadows from images". Proceedings of the 7th European Conference on Computer Vision-Part IV Page(s): 823-836. 2002
- [11] G.D.Finlayson, M.S.Drew and C.Lu. "Intrinsic Images by Entropy Minimization". Proc. 8th European Conf. on Computer Vision. Page(s): 582-595. 2004