

Target Tracking System Using Multiple Cameras and Bayesian Estimation

D. G. Dondo, J. A. Redolfi, M. Griffa, G. M. Steiner and L. R. Canali, *Senior Member, IEEE*

Abstract— In recent years, the increase in amount of camera networks available has led to the rise up of new techniques and applications of signal processing. In this work, an implementation of image sensor network, cameras, for mobile target tracking is shown. More specifically, it is performed an application of particle filter in a distributed way, based on leader agents for state estimation. Finally, simulations are presented, as well as a real experiment made with a camera network to track a mobile platform.

Keywords— Target Tracking, Particle Filter, Camera Network

I. INTRODUCCIÓN

UNA red de sensores consiste en un conjunto de dispositivos generalmente idénticos, llamados nodos, desplegados sobre una región geográfica de interés, que se usa para la medición y monitoreo de diversos fenómenos físicos o para la detección y seguimiento de eventos, en forma cooperativa y coordinada.

El procesamiento de la información sensorial puede realizarse genéricamente de dos maneras: centralizada o distribuida. En una red centralizada los nodos solamente proveen información que es transmitida a una unidad de procesamiento central. Por el contrario, en el procesamiento distribuido, los nodos no sólo proveen la información sensorial sino que también realizan parte del procesamiento de la misma en forma cooperativa, usando para ello el poder computacional de cada nodo. La suposición básica, es que cada nodo en la red puede explotar la información recibida para optimizar las futuras acciones de sensado y manejar los limitados recursos de procesamiento y comunicación eficientemente.

Un ejemplo típico de aplicación, usando procesamiento cooperativo de la información en redes inalámbricas de sensores, puede ser localización y seguimiento de objetos móviles en forma colaborativa. Los nodos desplegados geoespacialmente, colectan señales desde un objetivo en movimiento u otra entidad dentro de su área de cobertura y comunican la información sobre el mismo con otros nodos vecinos para ubicar al objeto. Dependiendo del campo de aplicación, el objeto a seguir puede ser un animal, vehículo o persona. Uno de los sensores más adecuado para tales aplicaciones son las cámaras.

En aplicaciones de seguimiento, el objetivo es la medición y observación del estado del evento, cuya evolución temporal/espacial puede ser modelada como un proceso estocástico, que dependiendo de los sensores a utilizar, puede ser no lineal y no Gaussiano [1], [2], [3]. Un modelo válido para las observaciones, es el empleo de mediciones de las posiciones relativas del objetivo a seguir en el plano imagen de la cámara, respecto de la posición de cada nodo, lo que implica una ecuación no lineal en la observación y una distribución de probabilidad a posteriori no Gaussiana. Bajo estas consideraciones la estimación de estado puede ser realizada mediante técnicas de filtrado Bayesiano, más específicamente implementaciones secuenciales de Monte Carlo, conocidas como filtro de partículas [4].

La utilización de una red de cámaras puede expandir el horizonte tradicional de aplicaciones de monitoreo a través de un aumento del campo de visión [5], permitiendo así, percibir el entorno desde diferentes y dispares puntos de vista. Además el solapamiento de observaciones simultáneas puede ayudar al mejoramiento del monitoreo del entorno.

En este trabajo, consideramos una red descentralizada de cámaras para realizar tareas de estimación a través de procesamiento y comunicaciones locales. La contribución de este trabajo, es la implementación de un filtro de partículas en forma distribuida, siguiendo el esquema de nodos líderes [6] para realizar tareas de estimación de las posiciones de un objeto en movimiento, utilizando sensores de imagen y los resultados preliminares del empleo de este algoritmo sobre la información obtenida de una red de cámaras, para realizar el seguimiento de una plataforma móvil de experimentación.

El trabajo está organizado de la siguiente manera: la formulación del problema de seguimiento de un móvil mediante una red de cámaras, así como su descripción matemática, es el objeto de la sección II. Los resultados de diversas simulaciones numéricas se describen en la sección III. y resultados preliminares del seguimiento de un robot usando una red de cámaras se describen en la sección IV.

Finalmente, las conclusiones y los trabajos a futuro se exponen en la sección V.

II. MÉTODO PROPUESTO

En este trabajo se propone resolver el problema de seguimiento de objetivos usando mediciones de múltiples cámaras y una posterior fusión de éstas realizada por cada uno de los sensores usando estimación Bayesiana.

A continuación se describe el algoritmo de fusión propuesto, los modelos dinámicos y de observación utilizados.

A. Fusión de las mediciones

Para la estimación del estado del objeto se emplea un filtro de partículas, el cual es un método de Monte Carlo que

D. G. Dondo, CIII, Universidad Tecnológica Nacional, Facultad Regional Córdoba, Córdoba, Argentina, dgonzalezdondo@frc.utn.edu.ar

J. A. Redolfi, CONICET, CIII, Universidad Tecnológica Nacional, Facultad Regional Córdoba, Córdoba, Argentina, jredolfi@frc.utn.edu.ar

M. Griffa, CIII, Universidad Tecnológica Nacional, Facultad Regional Córdoba, Córdoba, Argentina, martingriffa@frc.utn.edu.ar

G. M. Steiner, CIII, Universidad Tecnológica Nacional, Facultad Regional Córdoba, Córdoba, Argentina, gsteiner@frc.utn.edu.ar

L. R. Canali, CIII, Universidad Tecnológica Nacional, Facultad Regional Córdoba, Córdoba, Argentina, lcanali@frc.utn.edu.ar

permite la estimación Bayesiana de sistemas no-lineales en forma secuencial.

Comenzamos asumiendo que N nodos están desplegados aleatoriamente sobre un área, con posiciones preestablecidas (x_s^i, y_s^i) , $i \in \{1..N\}$. Cada uno de estos nodos está equipado con un sensor de imágenes de modo tal que pueden captar objetos dentro de su rango de visión, el cual depende del modelo de cámara. Se asume además un procesamiento previo en las imágenes, que nos permite detectar al objeto de interés y determinar su posición dentro de ellas. Suponemos también que existe un objeto en movimiento dentro del área de cobertura que en todo instante está siendo monitoreado por al menos un nodo sensor.

1) *Modelo dinámico* En el contexto de móviles maniobrando lentamente, consideramos un modelo de velocidad casi constante [7], [8].

El vector de estado \mathbf{x} representa las coordenadas (x, y) y las velocidades (\dot{x}, \dot{y}) del móvil en el plano 2-D para un tiempo de muestreo Δ entre observaciones.

$$\mathbf{x} \doteq [x \ y \ \dot{x} \ \dot{y} \Delta]^T \quad (1)$$

Donde el vector de velocidad es típicamente constante por períodos cortos. La ecuación de estado en tiempo discreto asociada en el instante de tiempo k es:

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{\Gamma}\mathbf{v}_k \quad (2)$$

Donde \mathbf{F} es la matriz de movimiento y $\mathbf{\Gamma}$ es la matriz de ruido de transición para un intervalo de muestreo Δ_k .

$\mathbf{v}_k \sim \mathcal{N}(0, Q)$ es el vector 3×1 del ruido de proceso, donde la última componente es el ruido asociado a los errores de sincronización entre las observaciones.

2) *Modelo de medición*: Las observaciones disponibles en el instante k son mediciones provenientes de un conjunto formado por M_k cámaras cercanas al móvil.

$$\mathbf{y}_k = h(\mathbf{x}_k) + \omega_k \quad (3)$$

Donde ω_k es un ruido Gaussiano independiente con media cero y varianzas σ_{θ}^2 , siendo el modelo de observación de cada cámara como sigue:

$$h(\mathbf{x}_k)^{i_k} = \begin{pmatrix} H_{11}^i \mathbf{x}_k + H_{12}^i \mathbf{y}_k + H_{13}^i \\ H_{31}^i \mathbf{x}_k + H_{32}^i \mathbf{y}_k + H_{33}^i \\ H_{21}^i \mathbf{x}_k + H_{22}^i \mathbf{y}_k + H_{23}^i \\ H_{31}^i \mathbf{x}_k + H_{32}^i \mathbf{y}_k + H_{33}^i \end{pmatrix} \quad (4)$$

Donde $h(\mathbf{x}_k)^{i_k}$ es el vector con las coordenadas del objetivo en píxeles (u_k^i, v_k^i) , medido desde el nodo $\{i; i \in M\}$. Los valores H_{mm}^i , son los elementos de la matriz de homografía entre el plano imagen y el plano del piso 2D sobre el cual se mueve el objetivo.

B. Filtro de partículas

El filtro propuesto aquí es una implementación del filtro de partículas del tipo muestreo de importancia y remuestreo, descrito en [4], [7]. El estado que no se puede observar $\{\mathbf{x}_k; k \in \mathbb{N}\}$ es modelado como un proceso de Markov con una distribución de probabilidad inicial $p(\mathbf{x}_0) \sim \mathcal{N}(\mathbf{M}_0, \Sigma_0)$ y una probabilidad de transición $p(\mathbf{x}_k | \mathbf{x}_{k-1})$. Se asume que las

observaciones $\{\mathbf{y}_k; k \in \mathbb{N}\}$ son condicionalmente independientes en el tiempo dado el proceso \mathbf{x}_k y la función de verosimilitud $p(\mathbf{y}_k | \mathbf{x}_k)$ [9]. El objetivo principal es la estimación instantánea del valor de expectación del estado actual. La función básica del filtro es aproximar la densidad a posteriori del estado del móvil, dadas todas las mediciones, a través de un conjunto de P puntos $\{\mathbf{x}_k^p; p \in P\}$, llamados partículas, con sus correspondientes pesos ω_k^p . Esto implica una discretización de $p(\mathbf{x}_k | \mathbf{y}_{1:k})$:

$$p(\mathbf{x}_k | \mathbf{y}_{1:k}) \approx \sum_{i=1}^P \omega_k^i \delta(\mathbf{x}_k - \mathbf{x}_k^i) \quad (5)$$

Donde δ denota la función delta de Dirac. Las partículas y sus pesos son actualizados recursivamente a medida de que se disponen de nuevas mediciones. Las partículas se actualizan usando la ecuación de estado en tiempo discreto (2) distribuida de acuerdo a la densidad de transición:

$$\mathbf{x}_k^i \sim p(\mathbf{x}_k | \mathbf{x}_{k-1}^i, \mathbf{y}_k) \quad (6)$$

Para solucionar el problema de degeneración de las partículas se utiliza un esquema de remuestreo. Esto elimina muestras con bajos pesos de importancia y replica muestras con alto peso. El resultado son muestras independientes idénticamente distribuidas uniformemente

$$\{\mathbf{x}_k^i, \omega_k^p\} \rightarrow \{\hat{\mathbf{x}}_k^i, \hat{\omega}_k^i\}, \hat{\omega}_k^i = 1/P$$

La estimación del estado se obtiene por el cálculo de la esperanza:

$$\hat{\mathbf{x}}_k = E\{\mathbf{x}_k | \mathbf{y}_k\} = \sum_{i=1}^P \hat{\omega}_k^i \hat{\mathbf{x}}_k^i \quad (7)$$

En la Fig. 1 se ilustra 200 muestras (partículas) obtenidas desde la densidad a posteriori de la estimación de la posición en un gráfico de dispersión. En el mismo se puede observar trayectoria estimada (verde) por la red de sensores y la real (azul) para diferentes instantes de tiempo. Las líneas que parten desde los nodos líderes indican las observaciones de posición (línea a trazos) y las posiciones estimadas (las líneas continuas.)

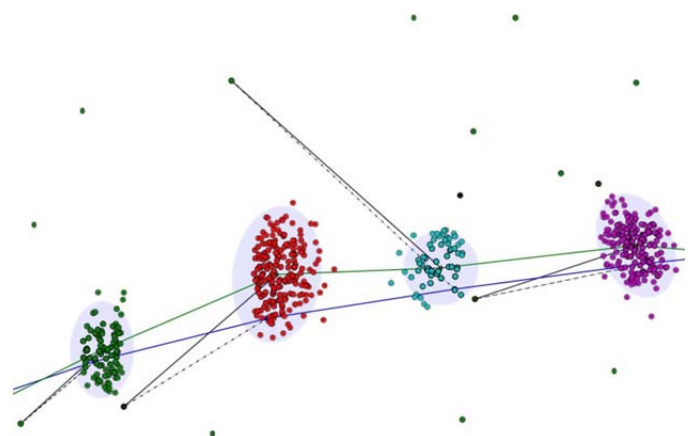


Figura 1. Muestras de la distribución de densidad a posteriori en la estimación de la posición de un objeto para diferentes instantes de tiempo.

C. Transferencia del liderazgo

Se asume que el nodo más cercano al objeto móvil puede medir con mejor precisión que el resto. Como así también que cada nodo posee el conocimiento de su propia ubicación geográfica y la de sus vecinos. Dada la estimación actual de la posición del móvil, se desea incrementalmente actualizar esta información transfiriendo el liderazgo a un nodo más adecuado para llevar a cabo la tarea de filtrado [10].

La decisión de la selección del nuevo líder se realiza sin el conocimiento explícito de las mediciones que éste pueda aportar, sino en base a su ubicación con respecto a la estimación de la posición. De esta manera, el nuevo líder es determinado en base a su posición relativa a la estimación. A causa del movimiento del objeto, el líder más apropiado tiene que ser también una función del tiempo. Por lo tanto, en cada instante k , el líder utiliza su estimación de la distribución a posteriori para elegir un nuevo nodo líder al tiempo $k + 1$ de acuerdo a su ubicación.

Sea $\Lambda(l)$ el conjunto de nodos vecinos al nodo líder l que pueden comunicarse directamente con él en un solo lazo. Sea $(\hat{x}_{k+1}, \hat{y}_{k+1})$ la ubicación predicha del móvil en el instante k . Entonces se elige al nodo más cercano a esta ubicación como el nuevo líder. Ésto es, el nuevo nodo líder l^* es dado por:

$$l^* = \arg \min_{i \in \Lambda(l)} [(x_s^i - \hat{x}_{k+1})^2 + (y_s^i - \hat{y}_{k+1})^2] \quad (8)$$

Donde (x_s^i, y_s^i) son las coordenadas del sensor i . El líder puede elegirse a sí mismo como nuevo líder; en caso contrario, éste debe propagar las partículas de la distribución a posteriori hacia el nuevo líder. En la Fig. 2 se puede observar el mecanismo de transferencia del líder, donde el próximo líder es el más cercano a la estimación de la posición.

III. SIMULACIONES

En esta sección, presentamos los resultados obtenidos de la aplicación del filtro de partículas con la utilización de un modelo de observación de posición, para el seguimiento de un objeto dentro de una red de cámaras. Para ello se llevó a cabo una representación de una red de cámaras distribuidas aleatoriamente sobre una región rectangular, con una densidad de cobertura de 1 cámara cada $10m^2$. Para cada cámara se simuló la proyección del plano imagen sobre el plano 2D

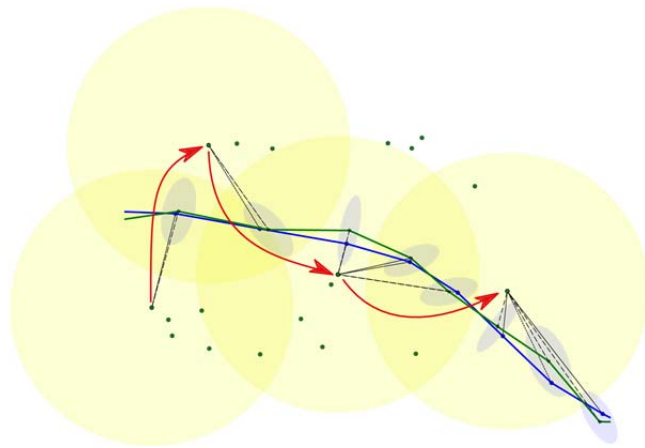
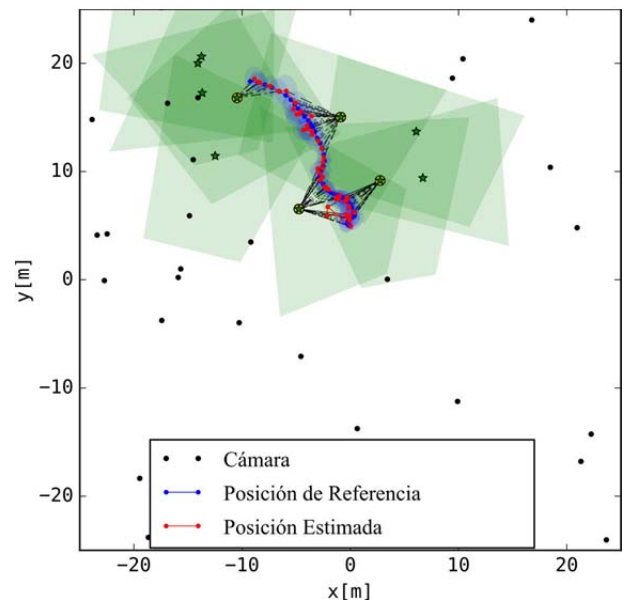


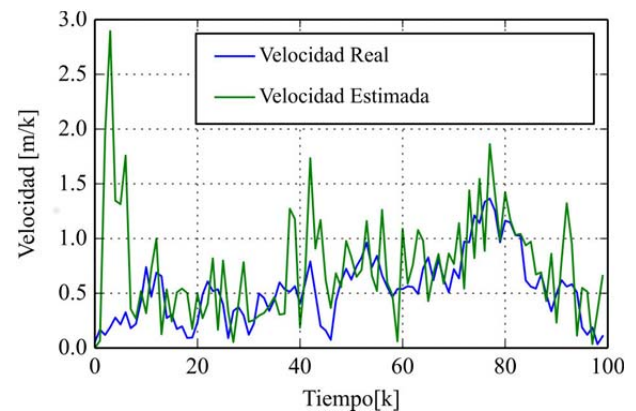
Figura 3. Transferencia del liderazgo hacia el nodo más cercano a la estimación de la posición.

donde se mueve el objetivo. El esquema propuesto es aplicado al seguimiento de un objeto en movimiento. El ensayo se realizó generando en forma artificial y controlada una trayectoria aleatoria con un período de simulación de $T = 100$.

En la Fig. 3 podemos ver una ilustración de la trayectoria realizada por el objeto dentro de la red, como así también la estimación de la velocidad correspondiente. Las áreas verdes representan las áreas de cobertura de cada cámara. Para medir el desempeño se generaron $n = 100$ corridas de Monte Carlo para cada trayectoria con idénticos estados iniciales y parámetros del modelo para medir la performance del filtro de partículas. Se propuso un $N = 200$ como cantidad de



(a) Trayectoria aleatoria



(b) Velocidad aleatoria

Figura 2. Trayectoria aleatoria junto con las estimaciones de posición y velocidad correspondientes.

partículas para el filtro. Para cada corrida de simulación y para cada salto de tiempo k , cuando una estimación fue obtenida, se calculó el error en términos de la distancia Euclídea entre la posición estimada y la posición real del objetivo. En la Fig. 4 se observa las medias del error cuadrático medio en la estimación y su desviación para la estimación de la posición

para la trayectoria aleatoria analizada para 100 corridas de

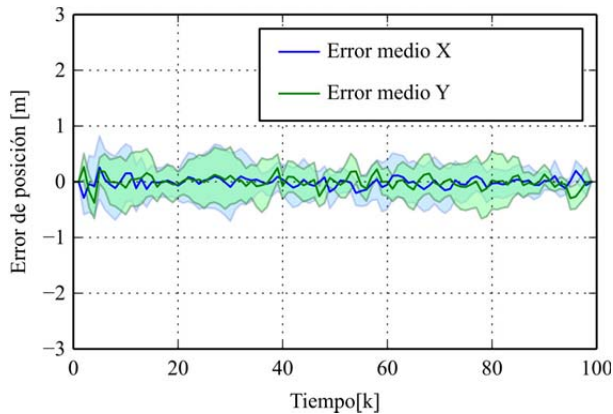


Figura 6. Media y desviación estándar de los errores en la estimación de la posición para diferentes trayectorias con 100 iteraciones independientes cada una.

simulaciones. Analizando dicha figura se puede ver una rápida convergencia del filtro y un bajo error de estimación. Los resultados cualitativos demuestran que la estimación realizada sigue correctamente al objetivo.

IV. RESULTADOS EXPERIMENTALES

En esta sección se describen resultados preliminares del seguimiento de un objetivo usando una red de cámaras.

Después de validar el funcionamiento del método con simulaciones, se decidió montar una red de 3 cámaras del tipo IP en el laboratorio de experimentación de robótica del CIII (Centro de Investigación en Informática para la Ingeniería, FRC, UTN. URL: <http://ciiii.frc.utn.edu.ar>) para realizar el seguimiento de un robot móvil. Específicamente se utilizó la plataforma ROMAA [11], la cual posee una PC de a bordo, una cámara y sensores de odometría en las ruedas. Las cámaras utilizadas como sensores son las TL-SC3230N de la firma TP-LINK, que cuentan con una resolución máxima de 1,3 Mega-píxel (1280×1024), un ángulo de visión horizontal de 47° y conectividad Wi-Fi. Las dimensiones del laboratorio son de aproximadamente $8 \times 4m$, de tal manera que para cubrir la mayor área posible se ubicaron las cámaras a una distancia de 4m entre ellas, colocadas a una altura de 3,3m y apuntando hacia el suelo.

En este experimento se generó una trayectoria con el ROMAA para luego comparar el resultado de la odometría del robot con la estimación de la posición obtenida con el filtro.

A. Obtención de las mediciones

Como se explicó anteriormente, el filtro estima el estado de la plataforma basado en las observaciones de la posición realizadas por cada sensor. Para el caso de cámaras, se necesita conocer la posición del ROMAA en el plano de la imagen de esta. Las cámaras fueron configuradas para obtener una imagen cada 100ms y luego sobre esta secuencia de imágenes se aplicó un algoritmo de seguimiento, el cual nos devuelve las coordenadas de un recuadro en donde es más probable que se encuentre el robot. De este recuadro, se tomó el centro como posición del robot en la imagen. Esto se

muestra en la Fig. 5, en donde se ve un recuadro sobre el robot de experimentación y un círculo que indica el centro del robot.

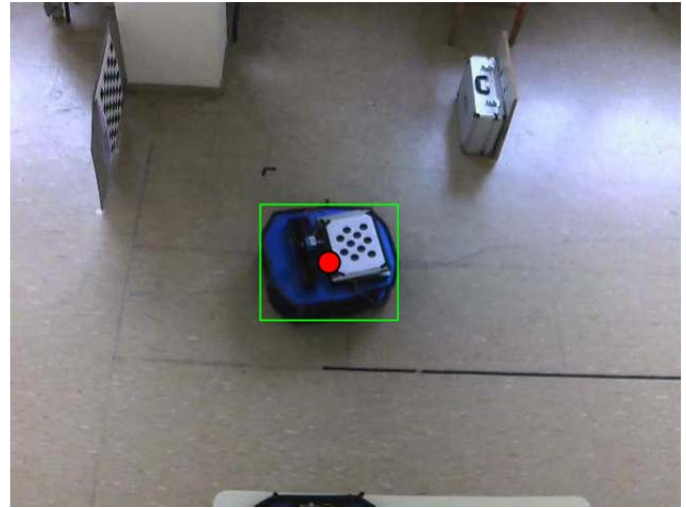


Figura 5. Obtención de la posición del ROMAA en el plano imagen de una

El algoritmo de seguimiento utilizado es CMT [12], el cual permite el seguimiento de objetos en secuencias largas de imágenes. Este algoritmo combina técnicas de tracking basadas en apariencia y búsqueda de correspondencias entre descriptores de características basada en consenso. Para la inicialización del CMT se necesita un recuadro inicial del objeto a seguir, el cual es obtenido con un detector de movimiento, el cual detecta la entrada de algún objeto en el área de visión de cada una de las cámaras y genera un recuadro para iniciar el seguimiento.

Las salidas del CMT que se ejecuta sobre cada sensor son las observaciones utilizadas por el filtro para realizar la estimación de la posición del objetivo.

Para la utilización del algoritmo propuesto con esta configuración de cámaras fue necesario conocer las posiciones de las cámaras, como así también las homografías asociadas a cada una de ellas. Para ello se realizó una calibración extrínseca e intrínseca aplicando una técnica convencional mediante un patrón de calibración [13]. En la Fig. 6 se puede observar la proyección de los planos de imagen sobre el plano del piso de las 3 cámaras presentes en el sistema.

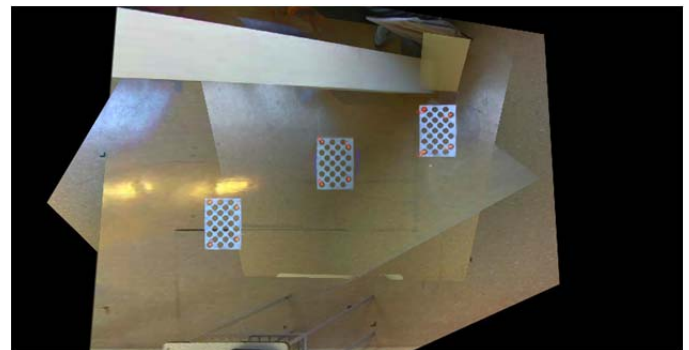


Figura 4. Área de cobertura de las cámaras. Proyecciones sobre el plano del piso.

B. Resultados

Utilizando como observación las coordenadas en píxel de las posiciones del ROMAA en cada cámara y empleando un modelo de velocidad constante para el movimiento del mismo, se realizó la estimación de la posición en el plano del piso.

En la Fig. 7 se muestra la trayectoria estimada por la red de cámaras (en rojo). Las posiciones de las cámaras obtenidas en la etapa de calibración, están representadas por una estrella, las líneas indican desde que cámara se han llevado a cabo las estimaciones. Las elipses verdes alrededor de las posiciones estimadas representan la varianza en la estimación.

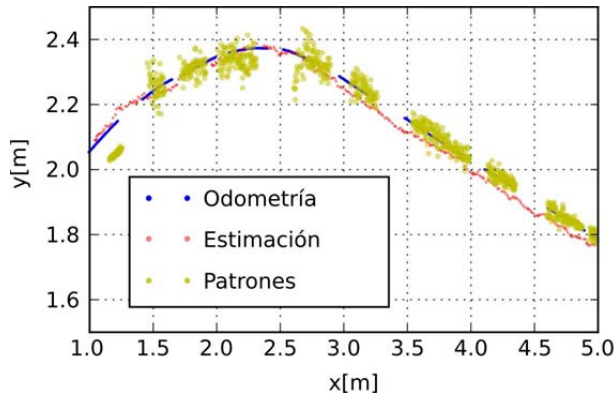


Figura 7. Resultados de la estimación de la posición comparada con la obtenida con la odometría del encoder y con el arreglo de patrones.

Se llevó a cabo una comparación preliminar entre los resultados obtenidos en la estimación de la posición, los obtenidos mediante la odometría propia del ROMAA y las mediciones realizadas utilizando la cámara de a bordo de la plataforma y un arreglo externo de patrones de calibración. Las trayectorias obtenidas con cada uno de los métodos se observa en la Fig. 8.

V. CONCLUSIONES

En el trabajo bosquejado se presentó la aplicación de un esquema de filtro de partículas distribuido basado en agentes líderes para la realización de tareas de seguimiento en una red de cámaras. Para la evaluación del mismo se llevaron a cabo

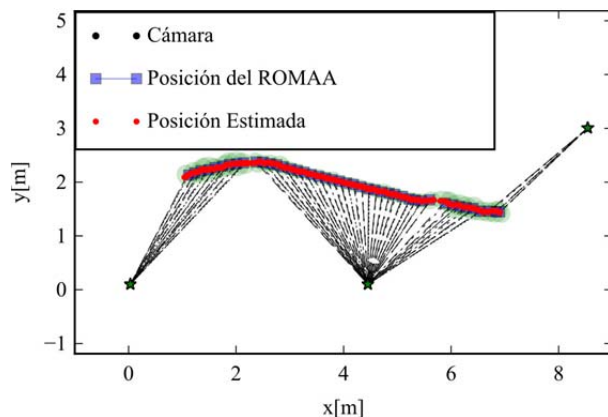


Figura 8. Trayectoria del ROMAA estimada utilizando la red de cámaras.

simulaciones y una experimentación real utilizando una red de cámaras IP para realizar el seguimiento de una plataforma móvil.

Como trabajo a futuro, se planea validar el método propuesto con la comparación de otros sistemas de seguimiento o de estimación de estado, como así también llevar a cabo experimentaciones con un mayor número de cámaras.

REFERENCIAS

- [1] A. Mohammadi and A. Asif, *A consensus/fusion based distributed implementation of the particle filter*, in Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2011 4th IEEE International Workshop on, dec. 2011, pp. 285–288.
- [2] O. Hlinka, O. Sluciak, F. Hlawatsch, P. Djuric, and M. Rupp, *Likelihood Consensus and Its Application to Distributed Particle Filtering*, Signal Processing, IEEE Transactions on, vol. PP, no. 99, p. 1, 2012.
- [3] A. Chiuso, F. Fagnani, L. Schenato, and S. Zampieri, *Gossip Algorithms for Simultaneous Distributed Estimation and Classification in Sensor Networks*, Selected Topics in Signal Processing, IEEE Journal of, vol. 5, no. 4, pp. 691–706, aug. 2011.
- [4] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, *A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking*, Signal Processing, IEEE Transactions on, vol. 50, no. 2, pp. 174–188, feb 2002.
- [5] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, *A survey on wireless multimedia sensor networks*, Computer Networks, vol. 51, no. 4, pp. 921–960, 2007.
- [6] O. Hlinka, F. Hlawatsch, and P. Djuric, *Distributed particle filtering in agent networks: A survey, classification, and comparison*, Signal Processing Magazine, IEEE, vol. 30, no. 1, pp. 61–81, Jan 2013.
- [7] B. Ristic, S. Arulampalam, and N. c. Gordon, *Beyond the Kalman filter: particle filters for tracking applications*. Boston, London: Artech House, 2004.[Online]. Available: <http://opac.inria.fr/record=b1102164>
- [8] Y. Bar-Shalom, T. Kirubarajan, and X.-R. Li, *Estimation with Applications to Tracking and Navigation*. New York, NY, USA: John Wiley & Sons, Inc., 2002.
- [9] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*, ser. Dover Books on Electrical Engineering. Dover Publications, 2013.
- [10] F. Zhao, J. Shin, and J. Reich, *Information-driven dynamic sensor collaboration*, Signal Processing Magazine, IEEE, vol. 19, no. 2, pp. 61–72, Mar 2002.
- [11] G. Perez Paina, R. Araguas, D. Gaydou, G. Steiner, and L. Canali, *Romaa-ii, an open architecture mobile robot*, Latin America Transactions, IEEE (Revista IEEE America Latina), vol. 12, no. 5, pp. 915–921, Aug 2014.
- [12] G. Nebehay and R. Pflugfelder, *Consensus-based matching and tracking of keypoints for object tracking*, in Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on. IEEE, 2014, pp. 862–869.
- [13] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge universitypress, 2003.



Diego Gonzalez Dondo recibió el título de Ingeniero Electrónico otorgado por la Universidad Tecnológica Nacional (UTN) en el año 2011, y en la actualidad es estudiante de posgrado de la carrera de doctorado en la Facultad Regional Córdoba de la Universidad Tecnológica Nacional. Es docente del Departamento de Ingeniería Electrónica de la Facultad Regional Córdoba de la UTN; y se desempeña como investigador en el Centro de Investigación en Informática para la Ingeniería, en el área de fusión sensorial y visión por computadora.



Redolfi Javier Andrés recibió el título de Ingeniero Electrónico en la Universidad Tecnológica Nacional (UTN) en el año 2011 y desde el año 2013 es estudiante de posgrado de la carrera de doctorado en Ciencias de la Ingeniería en la Universidad Nacional de Córdoba (UNC). Actualmente es becario doctoral del CONICET y se desempeña como investigador en el Centro de Investigación en Informática para la Ingeniería (CIII), en el área de visión por computadora.



Martin Griffa, estudiante avanzado de Ingeniería Electrónica en la Universidad Tecnológica Nacional (UTN) Facultad Regional Córdoba. Se desempeña como becario estudiante en el Centro de Investigación en Informática para la Ingeniería, en el área visión por computadora.



Guillermo Max Steiner graduado con el grado de ingeniero en electrónica en la Universidad Tecnológica Nacional - Regional Córdoba, Córdoba, Argentina en el año 2000 y posteriormente con el título de doctor mención electrónica en el año 2013 en la misma institución, actualmente se desempeña como investigador en el Centro de Investigación en Informática para la Ingeniería UTN-FRC, en el área de Visión Artificial y Robótica.



Luis Rafael Canali (M, 1993; SM 2006) Ingeniero en Electrónica, Universidad Tecnológica Nacional (UTN), Argentina, 1977. Doctor en Ingeniería, Mención Electrónica, UTN, 1999. Profesor del Departamento de Electrónica, UTN en el Campus de Córdoba, Argentina. Intereses: Control Numérico, Robótica, Procesamiento de Señales.