

---

# A Bayesian Solution to Robustly Track Multiple Objects from Visual Data

M. Marrón, J.C. García, M.A. Sotelo, D. Pizarro, I. Bravo, and J.L. Martín

Department of Electronics, University of Alcalá, Edificio Politécnico, Campus Universitario, Alcalá de Henares, 28871 (Madrid), Spain  
marta@depeca.uah.es, jcarlos@depeca.uah.es, sotelo@depeca.uah.es,  
pizarro@depeca.uah.es, ibravo@depeca.uah.es, jlmartin@depeca.uah.es

**Summary.** Different solutions have been proposed for multiple objects tracking based on probabilistic algorithms. In this chapter, the authors propose the use of a single particle filter to track a variable number of objects in a complex environment. Estimator robustness and adaptability are both increased by the use of a clustering algorithm. Measurements used in the tracking process are extracted from a stereo-vision system, and thus, the 3D position of the tracked objects is obtained at each time step. As a proof of concept, real results are obtained in a long sequence with a mobile robot moving in a cluttered scene.

## 1 Introduction

Probabilistic algorithms in their different implementations (Multi-Hypothesis Techniques – MHT – [1], Particle Filters – PF – [2,3] and their diversifications [4,5]) have fully shown their reliability in estimation tasks. Nowadays these methods are widely applied to solve positioning problems in robot autonomous navigation [6,7].

The idea of tracking multiple objects appeared with the first autonomous navigation robot to overcome the obstacle avoidance problem, and soon probabilistic algorithms, such as PFs [8,9] and Kalman Filters (KFs) [10,11], were applied to achieve this aim. The objective is, in any case, to calculate the posterior probability ( $p(\vec{x}_t|\vec{y}_{1:t})$ <sup>1</sup>) of the state vector  $\vec{x}_t$ , that informs about the position of the objects to track, in the recursive two steps standard estimation process (prediction-correction), in which, at least, some of the involved variables are stochastic, and by means of the Bayes rule.

To solve the multiplicity problem, the use of an expansion of the state vector ( $\vec{\chi}_t = \{\vec{x}_t^1, \vec{x}_t^2, \dots, \vec{x}_t^k\}$ ) that includes the model of all objects to track was the first solution proposed in [12].

---

<sup>1</sup> Definition of all variables is included in Table 1.

**Table 1.** Variables definition

Variables	Definition
$\vec{x}_t$	State vector. In the tracking application this vector contains the 3D position and the 2D ground speed in Cartesian coordinates
$\vec{x}_{t t-1}$	State vector prediction
$\vec{y}_t$	Measurements vector. In the tracking application this vector contains the 3D position in Cartesian coordinates
$f(\vec{x}_t, \vec{u}_t, \vec{o}_t)$	Transition model. $\vec{u}_t$ is the input vector and $\vec{o}_t$ is the noise vector related with the states
$p(\vec{x}_t \vec{x}_{t-1})$	Transition model in the model Markovian definition
$p(\vec{y}_{1:t})$	Measurements distribution
$h(\vec{x}_t, \vec{r}_t)$	Observation model. $\vec{r}_t$ is the noise vector related with the measurements
$p(\vec{y}_t \vec{x}_t)$	Observation model in the model Markovian definition. This density informs about measurements likelihood
$p(\vec{x}_t \vec{y}_{1:t})$	Belief or posterior distribution. Result of the state vector probabilistic estimation
$p(\vec{x}_t \vec{y}_{1:t-1})$	Prior distribution. Probabilistic prediction of the state vector
$S_t = \left\{ \vec{x}_t^{(i)}, \tilde{w}_t^{(i)} \right\}_{i=1}^n$	Particle set. Discrete representation of the belief used in the PF. Defined by $n$ normal weighed $\tilde{w}_t^{(1:n)}$ evaluations of the state vector $x_t^{-(1:n)}$
$S_{t t-1}$	Prediction of the particle set
$n$	Total number of particles
$n_{m,t} = \gamma_t \cdot n$	Number of particles to be inserted at the re-initialization step
$w(\vec{x}_{0:t}) \equiv \vec{w}_t = \left\{ w_t^{(i)} \right\}_{i=1}^n$	Importance sampling function. Continuous representation of the weights array $\vec{w}_t$
$q(\vec{x}_t \vec{x}_{0:t-1}\vec{y}_{1:t})$	Best approximation to the belief
$m$	Number of measurements in the set
$Y_t = \{\vec{y}\}_{i=1}^m$	Measurements set
$k$	Number of clusters
$G_{1:k,t} \equiv \{\vec{g}_{1:k,t}, L_{1:k,t}\}$	Clusters set. Each cluster is defined by its centroid $\vec{g}_{1:k,t}$ in the clustering characteristics space, and its member set $L_{1:k,t}$
$\{d_{i,j}\}_{i=1,j=1}^{m,k}$	Distance defined in the clustering characteristic space between the centroides $\vec{g}_{1:k,t}$ and the data set $Y_t = \{\vec{y}\}_{i=1}^m$ in the tracking application

The computational load of the resultant estimator does not allow achieving a real time execution of the algorithm for more than four or five objects [13].

Another solution for the multiple objects tracker is to use a standard estimator to track each object but, apart from the inefficiency of the final algorithm [14], it cannot deal easily with a dynamic number of objects [15].

In any case, in order to achieve a robust multi-tracking system, it is necessary to include an association algorithm to correctly insert the information included in the observation model to the estimation process. Most of the association solutions are based on the Probabilistic Data Association (PDA) theory [16], such as the Joint Probabilistic Particle Filter (JPDAF) like in [17] or in [18]. Again, the problem related to these techniques is the execution time.

In this context the authors propose in [19] another solution to the multi-tracking problem based on a PF. In this case, the multi-modality of the filter is exploited to perform the estimation task for various models with a single PF, and a clustering algorithm is used as association process in the multi-modal estimation, whose deterministic behavior is also exploited in order to increase the multi-tracker robustness.

The algorithm obtained is called Extended Particle Filter with Clustering Process (XPFCP). This solution has been tested in complex indoor environments with sonar [19] and vision data [20] with good results.

The choice of vision sensors to implement the observation system of the tracking application guarantees a rich amount of information from the objects in the world. For this reason, the final development described here is based on visual information.

In this chapter, a general revision of the global tracking system is included and a complete analysis of the results obtained with the multi-tracking proposal is exposed.

## 2 System Description

The complete obstacle detection and tracking system proposed is described in Fig. 1. The objective is to design a tracker that detects and predicts the movement and position of dynamic and static objects in complex environments, so two main constraints are taken into account in the development:

- *Indoor environment is unknown*, because no map information is available, and complex, because hard dynamic and crowded situations are frequent.
- *A real time application in a modular organization has to be achieved*, in order to attach it to any robotic autonomous navigator.

As it can be noticed in Fig. 1, three main processes are included in the global tracking system:

1. *A stereovision system* is used to obtain 3D position information from the elements in the environment.
2. *The extracted 3D position data is then classified in two types*: measurements related with the objects to track; and information from the environmental structure that can be used in a partial-reconstruction process.

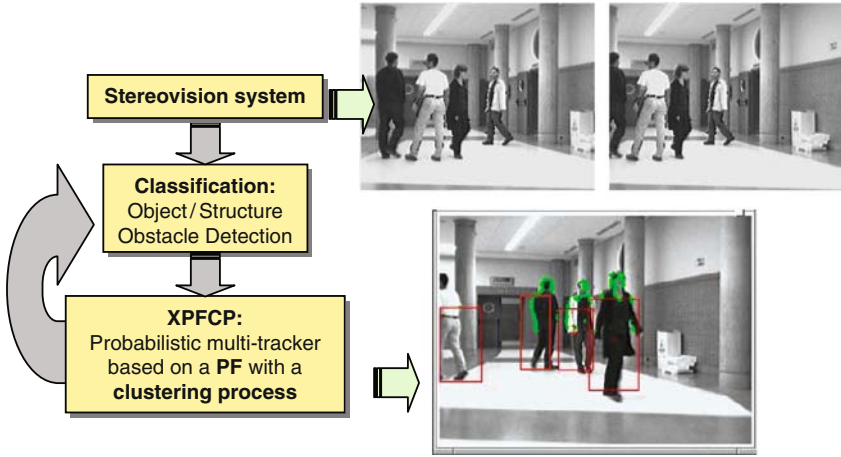


Fig. 1. Block diagram of the global tracking system

3. A probabilistic algorithm, the XPFCP, with two main components:
  - *An extended PF* is used to implement the multi-modal tracker. Using this kind of algorithm it is possible to estimate a variable number of probabilistic non-linear and non-Gaussian models with a single density function.
  - *A clustering algorithm* is inserted in the PF to develop the association process task and to increase the robustness and adaptability of the multi-modal estimator.

Descriptions of each one of the modules presented are completed in the following sections.

### 3 The Estimation Model

The main objective of XPFCP is to estimate the movement of objects around an autonomous navigation platform. In order to develop the tracking process a position estimation model has to be defined.

State vector encoding the objects position and speed in Cartesian coordinates at time  $t$  is represented by  $\vec{x}_t$ .

From a probabilistic point of view, this state vector can be expressed by a density function  $p(\vec{x}_t|\vec{y}_{1:t})$ , also called belief.

The evolution of this belief  $p(\vec{x}_t|\vec{y}_{1:t-1})$  is defined by a Markov Process, with transition kernel  $p(\vec{x}_t|\vec{x}_{t-1})$ , as follows:

$$p(\vec{x}_t|\vec{y}_{1:t-1}) = \int p(\vec{x}_t|\vec{x}_{t-1}) \cdot p(\vec{x}_{t-1}|\vec{y}_{1:t-1}) \cdot \partial\vec{x} \quad (1)$$

The transition kernel is derived from a simple motion model, which can be expressed as follows:

$$\begin{aligned}
 p(\vec{x}_t|\vec{y}_{1:t-1}) &\equiv \vec{x}_{t|t-1} = f(\vec{x}_{t-1}, \vec{o}_{t-1}), \\
 \vec{x}_{t|t-1} &= \begin{bmatrix} 1 & 0 & 0 & t_s & 0 \\ 0 & 1 & 0 & 0 & t_s \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \vec{x}_{t-1} + \vec{o}_{t-1}
 \end{aligned} \tag{2}$$

On the other hand, the measurements vector  $\vec{y}_t$  contains the 3D position information sensed by the vision system (see Table 1).

The probabilistic relation between this vector  $\vec{y}_t$  and the state one  $\vec{x}_t$  is given by the likelihood  $p(\vec{y}_t|\vec{x}_t)$ , that defines the observation model from a stochastic approach.

The observation model, that describes the deterministic relation expressed by the likelihood, is defined as follows:

$$\begin{aligned}
 p(\vec{y}_t|\vec{x}_t) &\equiv \vec{y}_t = h(\vec{x}_t, \vec{r}_t), \\
 \vec{y}_t &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \cdot \vec{x}_t + \vec{r}_t
 \end{aligned} \tag{3}$$

Both observation and motion models are used to estimate the state vector over time. As commented in the introduction section, different algorithms can be used in order to achieve this functionality. Our contribution in this point is to use a single PF to obtain a multi-modal distribution  $p(\vec{x}_t|\vec{y}_{1:t})$  that describes the estimated stochastic position of every object being tracked at each sample time  $t$ .

## 4 The Stereovision Classifier

Most of tracking systems developed in last years for autonomous navigation and surveillance applications are based on visual information; this is due to the diverse and vast amount of information included in a visual view of the environment.

Developing obstacle tracking tasks for robot’s navigation requires 3D information about the objects position in the robot moving environment.

As shown in Fig. 1, position information obtained with the stereovision system is related both with the environment and the objects to track. Therefore it is needed a classification algorithm in order to organize measurements coming from the vision system in two groups or classes:

- *Objects class.* Formed by points that inform about position of objects. These conform the data set that is input in the multiple objects tracker as the measurement vector  $\vec{y}_t$ .

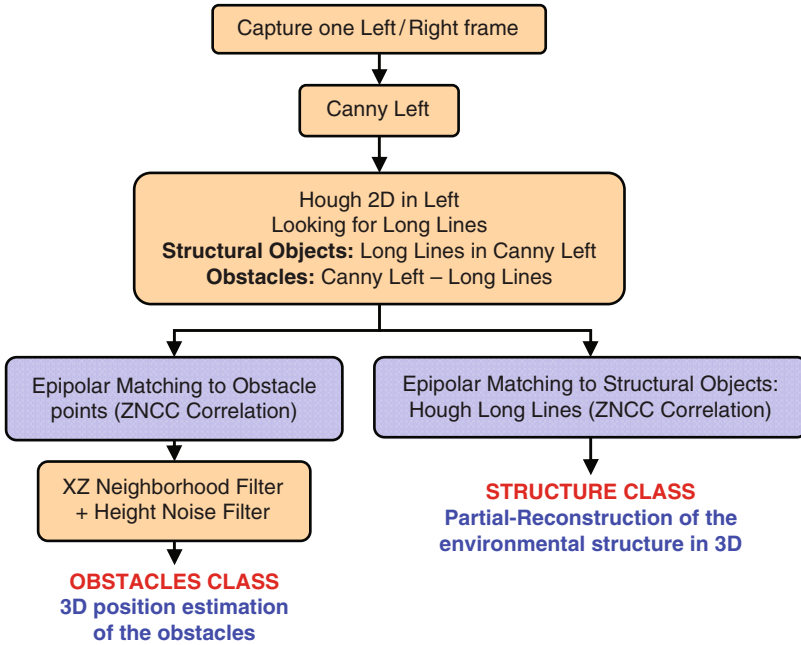


Fig. 2. Block diagram of the stereovision classifier and object detector

- *Structure class.* Formed by points related to elements in environmental structure (such as floor and walls). This data set can be used to implement a partial reconstruction of the environment in which the tracked objects and the robot itself are moving.

Figure 2 shows the proposal to classify the measurements extracted with the stereovision system. The detection and classification process is deeply described by the authors in [21], but a slight revision of its functionality is included in the following paragraphs:

1. The stereovision system proposed is formed by two synchronized black and white digital cameras statically mounted to acquire left and right images.
2. As the amount of information in each image is too big, a canny filter is applied to one of the pair of frames.
3. The classification process is performed to the edge pixels that appear in the canny image. Environmental structures edges have the common characteristic of forming long lines in the canny image. Due to this fact, the Hough transform has been chosen as the best method to define the pixels from the canny image that should be part of the structure class. The rest of the points in the canny image are assigned to the objects class.
4. Two correlation processes are used in order to find the matching point of each member in both classes. 3D position information is obtained with a

matching process applied to the pixels of each pair of frames, using the epipolar geometry that relates the relative position of the cameras.

With the described functionality, the classification process here proposed behaves as an obstacle detection module.

Also, this pre-processing algorithm selects wisely the most interesting data points from the big set of measurements that is extracted from the environment. This fact is especially important in order to achieve the real time specification pursuit. In fact, a processing rate of 15–33 fps has been achieved in different tests run with this classifier.

Some results of the classification process described are included in the results section of this chapter.

## 5 The Estimation Algorithm

A particle filter (PF) is used as a multi-modal tracker to estimate position and speed of objects in the environment, from the measurement array obtained in the classification process.

PF is a particularization of the Bayesian estimator in which the densities related to the posterior estimation (also called belief) is discretized. A detailed description of the PF mathematical base can be found in [2] and in [5].

As the state vector is not discretized, like it is in most of Bayes filter implementations, the PF is more accurate in its estimation than the KF or estimators based on a grid (MonteCarlo estimators). Moreover, due to the same reason, the computational load of this Bayes filter form is lower in this than in other implementations, and thus more adequate to implement real time estimation.

Finally, PFs include an interesting characteristic for multi-tracking applications: the ability of representing multiple estimation hypotheses with a single algorithm, through the multi-modality of the belief. This facility is not available in the optimal implementation of the Bayes estimator, the KF.

For all these reasons, the PF has been thought as the most appropriated algorithm to develop a multi-tracking system.

### 5.1 The XPFCP

Most of the solutions to the tracking problem, based on a PF, do not use the multi-modal character of the filter in order to implement the multiple objects position estimation task. The main reason of this fact is that the association process needed to allow the multi-modality of the estimator is very expensive in execution time (this is the case of the solutions based on the JPDAF) or lacks of robustness (as it is the case in the solution presented in [22]).

The XPFCP here presented is a multi-modal estimator based on a single PF that can be used with a variable number of models, thanks to a clustering

process that is used as association process in the estimation loop. The functionality of the XPFCP is presented in the following paragraphs.

The main loop of a standard Bootstrap PF [12] based on the SIR algorithm [13] starts at time  $t$  with a set  $S_{t-1} = \left\{ \vec{x}_{t-1}^{(i)}, \vec{w}_{t-1}^{(i)} \right\}_{i=1}^n$  of  $n$  random particles representing the posterior distribution of the state vector estimated  $p(\vec{x}_{t-1} | \vec{y}_{1:t-1})$  at the previous step. The rest of the process is developed in three steps, as follows:

1. *Prediction step.* The particles are propagated by the motion model  $p(\vec{x}_t | \vec{x}_{t-1})$  to obtain a new set  $S_{t|t-1} = \left\{ \vec{x}_{t|t-1}^{(i)}, \vec{w}_{t|t-1}^{(i)} \right\}_{i=1}^n$  that represents the prior distribution of the state vector at time  $t$ ,  $p(\vec{x}_t | \vec{y}_{1:t-1})$ .
2. *Correction step.* The weight of each particle  $\vec{w}_t = \left\{ w_t^{(i)} \right\}_{i=1}^n \equiv w(\vec{x}_{0:t})$  is then obtained comparing the measurements vector  $\vec{y}_t$  and its predicted value based on the prior estimation  $h(\vec{x}_{t|t-1})$ . In the Bootstrap version of the filter, these weights are obtained directly from the likelihood function  $p(\vec{y}_t | \vec{x}_t)$ , as follows:

$$w(\vec{x}_{0:t}) = w(\vec{x}_{0:t-1}) \cdot \frac{p(\vec{y}_t | \vec{x}_t) \cdot p(\vec{x}_t | \vec{x}_{t-1})}{q(\vec{x}_t | \vec{x}_{0:t-1}, \vec{y}_{1:t})} \tag{4}$$

$$\xrightarrow{q(\vec{x}_t | \vec{x}_{0:t-1}, \vec{y}_{1:t}) \propto p(\vec{x}_t | \vec{x}_{t-1})} w(\vec{x}_{0:t}) = w(\vec{x}_{0:t-1}) \cdot p(\vec{y}_t | \vec{x}_t)$$

3. *Selection step.* Using the weights vector  $\vec{w}_t = \left\{ w_t^{(i)} \right\}_{i=1}^n$ , and applying a re-sampling scheme, a new set  $S_t = \left\{ \vec{x}_t^{(i)}, \vec{w}_t^{(i)} \right\}_{i=1}^n$  is obtained with the most probable particles, which will represent the new belief  $p(\vec{x}_t | \vec{y}_{1:t})$ .

The standard PF can be used to robustly estimate the position of any kind of a single object defined through its motion model, but it cannot be directly used to estimate the position of appearing objects because there is not a process to assign particles to the new estimations.

In order to adapt the standard PF to be used to track a variable number of elements, some modifications must be included in the basic algorithm. In [22] an adaptation of the standard PF for the multi-tracking task is proposed. The algorithm described there was nevertheless not finally used because it is not robust enough.

The extension of the PF proposed by the authors in [20] includes a clustering algorithm to improve the behavior of the first extended PF, giving as a result the XPFCP process, shown in Fig. 3.

The clustering algorithm, whose functionality is presented in next section, organizes the vector of measurements in clusters that represent all objects in the scene. These clusters are then wisely used in the multi-modal estimator. Two innovations are included in the standard PF to achieve the multi-modal behavior:



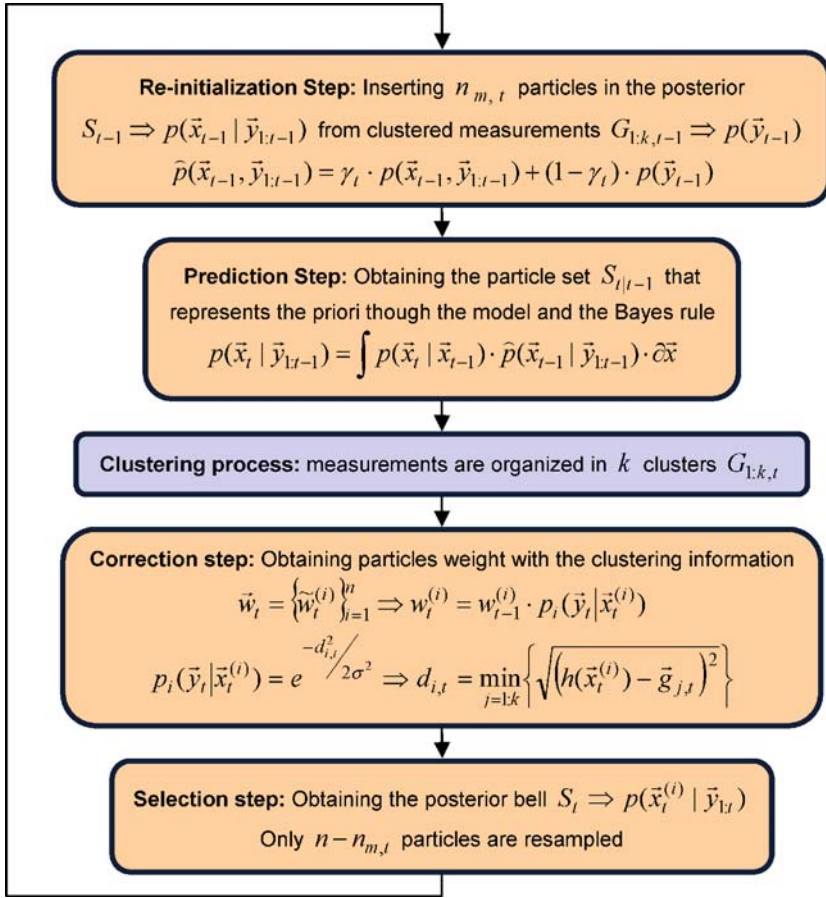


Fig. 3. Description of the XPFPCP functionality

- *With a new re-initialization step.*  $n_{m,t}$  from the  $n$  total number of particles that form the belief  $p(\bar{x}_t | \bar{y}_{1:t})$  in the PF are directly inserted from the measurements vector  $\bar{y}_t$  in this step previous to the prediction one. With this modification, measurements related to newly appearing objects in the scene have a representation in the priori distribution  $p(\bar{x}_{t-1} | \bar{y}_{1:t-1})$ . To improve the robustness of the estimator, the inserted particles are not selected randomly from the array of measurements  $\bar{y}_{t-1}$  but from the  $k$  clusters  $G_{1:k,t-1}$ . Choosing measurements from every cluster ensures a probable representation of all objects in the scene, and therefore, an increased robustness of the multi-tracker. Thanks to this re-initialization step the belief dynamically adapts itself to represent the position hypothesis of the different objects in the scene.
- *At the Correction step.* This step is also modified from the standard PF. On one hand, only  $n - n_{m,t}$  samples of the particle set have to be extracted

in this step, as the  $n_{m,t}$  resting ones would be inserted with the re-initialization. On the other hand, the clustering process is also used in this step, because the importance sampling function  $p_i(\vec{y}_t | \vec{x}_t^{(i)})$  used to calculate each particle weight  $w_t^{(i)}$  is obtained from the similarity between the particle and the  $k$  cluster centroids  $\vec{g}_{1:k,t}$ . Using the cluster centroids to weight the particles related to the newly appeared objects, the probability of these particles is increased, improving the robustness of the new hypotheses estimation. Without the clustering process, the solution proposed in [22] rejects these hypotheses, and thus, the multi-modality of the PF cannot be robustly exploited.

Figure 3 shows the XFPCP functionality, described in previous paragraphs. Some application results of the multi-modal estimator proposed by the authors to the multi-tracking task are shown at the end of this chapter. The robustness of this contribution is demonstrated there.

## 5.2 The Clustering Process

Two different algorithms have been developed for clustering the set of measurements: an adapted version of the K-Means for a variable number of clusters; and a modified version of the Subtractive fuzzy clustering. Its reliability is similar, but the proposal based on the standard K-Means shows higher robustness rejecting outliers in the measurements vector. A more detailed comparative analysis of these algorithms can be found in [23].

Figure 4 shows the functionality of the proposed version of the K-Means. Two main modifications to the standard functionality can be found in the proposal:

1. It has been adapted in order to handle a variable and initially unknown number  $k$  of clusters  $G_{1:k}$ , by defining a threshold  $distM$  in the distance  $d_{i,1:k}$  used in the clustering process.
2. A cluster centroids' prediction process is included at the beginning of the algorithm in order to minimize its execution time. With this information, the process starts looking for centroids near their predicted values  $\vec{g}_{0,1:k,t} = \vec{g}_{1:k,t|t-1}$ .

A validation process is also added to the clustering algorithm in order to increase the robustness of the global algorithm to spurious measurements. This process is useful when noisy measurements or outliers produce a cluster creation or deletion erroneously. The validation algorithm functionality is the following:

- *When a new cluster is created*, it is converted into a candidate that will not be used in the XFPCP until it is possible to follow its dynamics.
- *The same procedure is used to erase a cluster* when it is not confirmed with new measurements for a specific number of times.

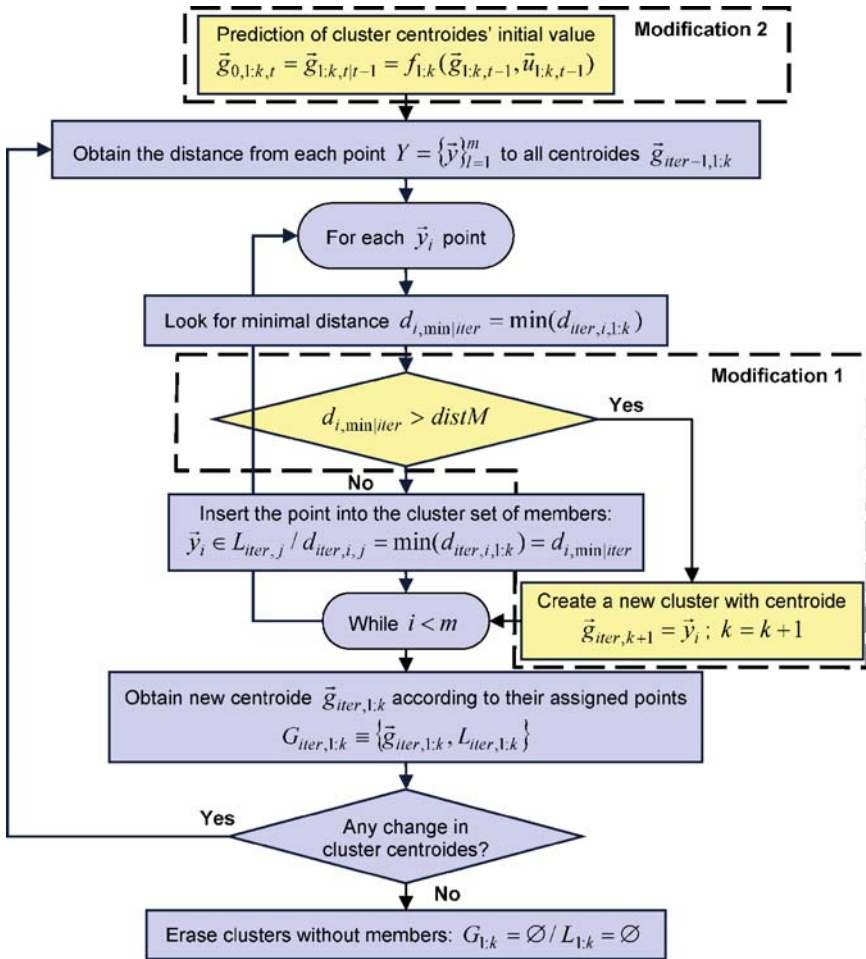


Fig. 4. Description of the K-Means clustering algorithm

The validation process is based in two parameters, which are calculated for each cluster:

- *Distance between the estimated and the resulting clusters centroide.* The centroides estimation process, already commented, is also used in the validation process. The estimated value of the centroides  $\bar{g}_{1:k,t|t-1}$  is compared with its final value at the end of the clustering process  $\bar{g}_{1:k,t}$ , in order to obtain a confidence value for the corresponding cluster validation.
- *Cluster likelihood.* A cluster probability value is calculated as a function of number of members in each cluster  $L_{1:k}$ .

The effectiveness of the clustering proposal is demonstrated in the following section, with different results.

## 6 Results

The global tracking algorithm described in Fig. 1 has been implemented in a mobile platform. Different tests have been done in unstructured and unknown indoor environments. Some of the most interesting results extracted from these tests are shown and analyzed in this section.

The stereovision system is formed by two black and white digital cameras synchronized with a Firewire connection and located on the robot in a static mounting arrangement, with a gap of 30 cm between them, and at a height of around 1.5 m from the floor.

The classification and tracking algorithms run in an Intel Dual Core processor at 1.8 GHz with 1 GB of RAM, at a rate of 10 fps. The mean execution time of the application is 80 ms.

### 6.1 Results of the Stereovision Classifier

Figure 5 shows the functionality of the classifier. Three sequential instants of one of the experiments are described in the figure by a pair of images organized vertically, and with the following meaning:

- *Upper row* shows the edge images obtained from the canny filter directly applied to the acquired frame. Both obstacles and environmental structure borders are mixed in those images.
- *Bottom row* shows the final frames in which points assigned to the objects class are highlighted over obstacle figures.

From the results shown in Fig. 5, it can be concluded that the classification objective has been achieved. Only points related to the obstacles in the scene have been classified in the obstacles class. As it can be noticed, the analyzed experiment has been developed in a complex and unstructured indoor environment, where five static and dynamic objects are present and cross their

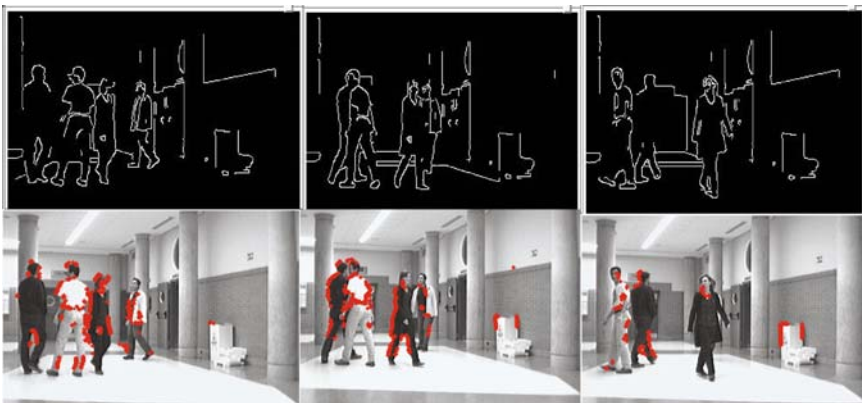


Fig. 5. Results of the classification algorithm in a real situation

paths generating partial and global occlusions. In any case, the proposed classification algorithm is able to extract 3D position points from every object in the scene.

The set of 3D position points assigned to the objects class can now be used in the multi-tracking task.

Nevertheless, the number of objects present in each final frame in Fig. 5 cannot be easily extracted from the highlighted set of points.

Furthermore, it can be noticed that the set of points are not equally distributed among all objects in the environment, and hence, the tracking algorithm should be able to manage object hypotheses with very different likelihood.

### 6.2 Results of the Estimation Algorithm

Figure 6 displays the functionality of the XPFCP in one of the tested situations. Three sequential instants of the estimation process are represented by a pair of images.

- *Upper row* displays the initial frames with highlighted dots representing the measurement vector contents obtained from the classification process, and rectangles representing the K-Means output.
- *Lower row* shows the same frame with highlighted dots representing each of the obstacle position hypotheses that the set of particles define at the XPFCP output. This final set of particles has also been clustered using the same K-Means proposal in order to obtain a deterministic output for the multi-tracker. Rectangles in this lower frame represent the clustered particles.

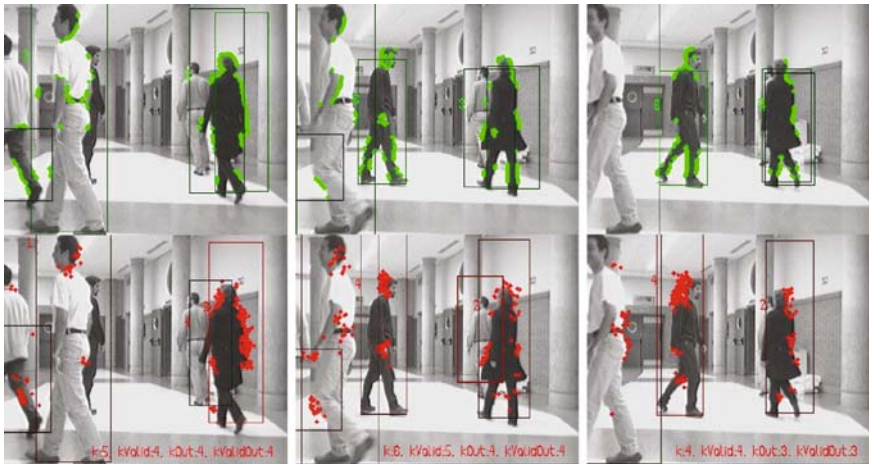


Fig. 6. Results of the multi-tracking algorithm XPFCP in a real situation

**Table 2.** Rate of different types of errors obtained with the XPFCP at the output of the K-Means, and at the end of the multi-tracking task in a 1,054 frames experiment complex situations with five and six objects

	K-Means (% frames with error)	XPFCP (% frames with error)
Missing	10.9	2.9
Duplicates	6.1	0
2 as 1	3.9	0
Total	20.9	2.9

Comparing the upper and lower image in Fig. 6, it can be noticed that the tracker based on the XPFCP can solve tracking errors such as object duplications generated in the input clustering process. An example of an object duplication error generated by the K-Means and successfully solved by the XPFCP can be seen in the third vertical pair of images (on the right side, in the last sequential instant) of Fig. 6.

Table 2 shows a comparison between the errors at the output of the clustering process and at the end of the global XPFCP estimator. In order to obtain these results an experiment of 1,054 frames of complex situations similar to the ones presented Figs. 5 and 6 has been run. The results displayed in Table 2 demonstrate the reliability and robustness of the tracker facing up to occlusions and other errors.

Figure 7 displays the tracking results extracted from the XPFCP output in another real time experiment. In this case 9 sequential instants of the experiment are shown, and each image represents one of them, from (a) to (i). The meaning of every frame is the same as in the lower row in Fig. 6.

The results displayed in Fig. 7 show that the tracker estimates correctly each obstacle position in the dynamic and unstructured indoor environment.

## 7 Conclusions

In this chapter the authors describe the functionality of a global tracking system based on vision sensors to be used by the navigation or obstacle avoidance module in an autonomous robot.

In order to achieve this objective, a specific classification algorithm for stereovision data has been developed. This process is used to separate visual position information related with obstacles from the one related with the environment.

An algorithm, called XPFCP, is used to estimate obstacles' movement and position in an unstructured environment. It has been designed as the kernel of the multi-tracking process. The XPFCP is based on a probabilistic multi-modal filter, a PF, and is completed with a clustering process based on a standard K-Means.



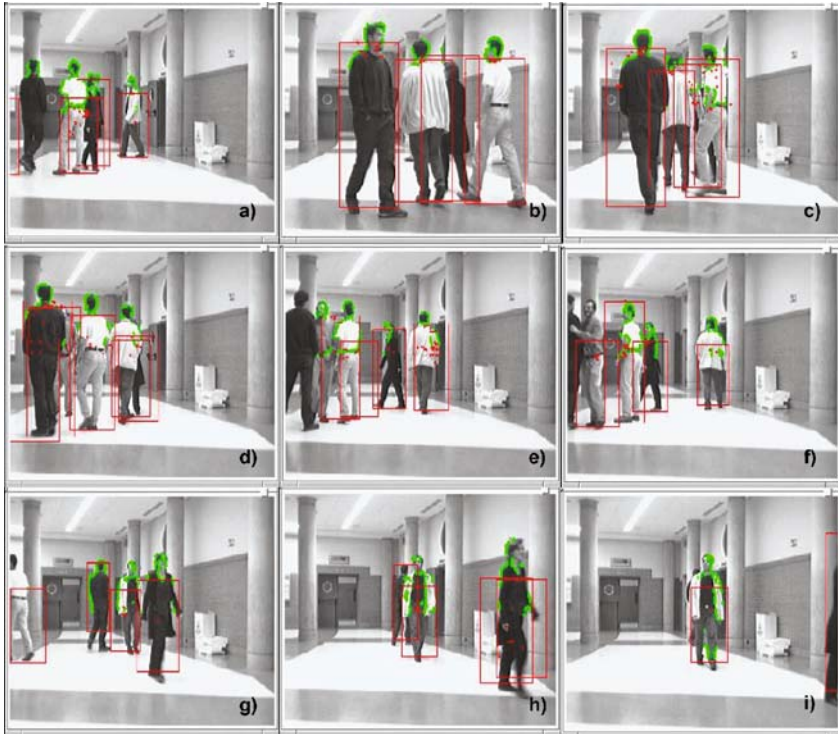


Fig. 7. Sequential images of a real time experiment with stereovision data

Results of the different processes involved in the global tracking system have been presented, demonstrating the successful behaviour of the different contributions. The main conclusions of these proposals are:

- The proposed tracking system has shown high reliability in complex situations where a variable number of static and dynamic obstacles are constantly crossing, and no preliminary knowledge of the environment is available.
- It has been demonstrated that the estimation of a variable number of systems can be achieved with a single algorithm, the XPFCP, and without imposing model restrictions.
- The use of a clustering process as association algorithm makes possible a robust multi-modal estimation with a single PF, and without the computational complexity of some other association proposals such as the PDAF.
- Thanks to the simplicity of its functional components (a PF and a modified K-Means) the XPFCP accomplishes the real time specification pursuit.
- Though vision sensors are used in the tracking process presented in the chapter, some other XPFCP designed can easily handle data coming up from different kinds of sensors. This fact makes the tracker proposed more flexible, modular, and thus, easy to use in different robotic applications than other solutions proposed in the related literature.

## Acknowledgments

This work has been financed by the Spanish administration (CICYT: DPI2005-07980-C03-02).

## References

1. D.B. Reid, An algorithm for tracking multiple targets, *IEEE Transactions on Automatic Control*, vol. 24, no 6, pp. 843–854, December 1979
2. M.S. Arulampalam, S. Maskell, N. Gordon, T. Clapp, A tutorial on particle filters for online nonlinear non-gaussian bayesian tracking, *IEEE Transactions on Signal Processing*, vol. 50, no 2, pp. 174–188, February 2002
3. N.J. Gordon, D.J Salmond, A.F.M. Smith, Novel approach to nonlinear/non-gaussian bayesian state estimation, *IEE Proceedings Part F*, vol. 140, no 2, pp. 107–113, April 1993
4. A. Doucet, J.F.G. de Freitas, N.J. Gordon, *Sequential montecarlo methods in practice*. Springer, New York, ISBN: 0-387-95146-6, 2000
5. R. Van der Merwe, A. Doucet, N. de Freitas, E. Wan, The unscented particle filter, *Advances in Neural Information Processing Systems*, NIPS13, November 2001
6. S. Thrun, Probabilistic algorithms in robotics, *Artificial Intelligence Magazine*, Winter 2000
7. D. Fox, W. Burgard, F. Dellaert, S. Thrun, Monte Carlo localization. Efficient position estimation for mobile robots, *Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI99)*, pp. 343–349, Orlando, July 1999
8. M. Isard, A. Blake, Condensation: Conditional density propagation for visual tracking, *International Journal of Computer Vision*, vol. 29, no 1, pp. 5–28, 1998
9. K. Okuma, A. Taleghani, N. De Freitas, J.J. Little, D.G. Lowe, A boosted particle filter: multi-target detection and tracking, *Proceedings of the Eighth European Conference on Computer Vision (ECCV04)*, *Lecture Notes in Computer Science*, ISBN: 3-540-21984-6, vol. 3021, Part I, pp. 28–39 Prague, May 2004
10. T. Schmitt, M. Beetz, R. Hanek, S. Buck, Watch their moves applying probabilistic multiple object tracking to autonomous robot soccer, *Proceedings of the Eighteenth National Conference on Artificial Intelligence (AAAI02)*, ISBN: 0-262-51129-0, pp. 599–604, Edmonton, July 2002
11. K.C. Fuerstenberg, K.C.J. Dietmayer, V. Willhoeft, Pedestrian recognition in urban traffic using a vehicle based multilayer laserscanner, *Proceedings of the IEEE Intelligent Vehicles Symposium (IV02)*, vol. 4, no 80, Versailles, June 2002
12. J. MacCormick, A. Blake, A probabilistic exclusion principle for tracking multiple objects, *Proceedings of the Seventh IEEE International Conference on Computer Vision (ICCV99)*, vol. 1, pp. 572–578, Corfu, September 1999
13. H. Tao, H.S. Sawhney, R. Kumar, A sampling algorithm for tracking multiple objects, *Proceedings of the International Workshop on Vision Algorithms at (ICCV99)*, *Lecture Notes in Computer Science*, ISBN: 3-540-67973-1, vol. 1883, pp. 53–68, Corfu, September 1999



14. J. Vermaak, A. Doucet, P. Perez, Maintaining multimodality through mixture tracking, Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV03), vol. 2, pp. 1110–1116, Nice, June 2003
15. P. Pérez, C. Hue, J. Vermaak, M. Gangnet, Color-based probabilistic tracking, Proceedings of the Seventh European Conference on Computer Vision (ECCV02), Lecture Notes in Computer Science, ISBN: 3-540-43745-2, vol. 2350, Part I, pp. 661–675, Copenhagen, May 2002
16. Y. Bar-Shalom, T. Fortmann, Tracking and data association (Mathematics in Science and Engineering, V.182), Academic Press, New York, ISBN: 0120797607, January 1988
17. C. Rasmussen, G.D. Hager, Probabilistic data association methods for tracking complex visual objects, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no 6, pp. 560–576, June 2001
18. D. Schulz, W. Burgard, D. Fox, A.B. Cremers, People tracking with mobile robots using sample-based joint probabilistic data association filters, International Journal of Robotics Research, vol. 22, no 2, pp. 99–116, February 2003
19. M. Marrón, M.A. Sotelo, J.C. García, Design and applications of an extended particle filter with a pre-clustering process -XPFCP-, Proceedings of the International IEEE Conference Mechatronics and Robotics 2004 (MECHROB04), ISBN: 3-938153-50-X, vol. 2/4, pp. 187–191, Aachen, September 2004
20. M. Marrón, J.C. García, M.A. Sotelo, D. Fernandez, D. Pizarro. XPFCP: An extended particle filter for tracking multiple and dynamic objects in complex environments, Proceedings of the IEEE International Symposium on Industrial Electronics 2005 (ISIE05), ISBN: 0-7803-8738-4, vol. I–IV, pp. 1587–1593, Dubrovnik, June 2005
21. M. Marrón, M.A. Sotelo, J.C. García, D. Fernández, I. Parra. 3D-visual detection of multiple objects and environmental structure in complex and dynamic indoor environments, Proceedings of the Thirty Second Annual Conference of the IEEE Industrial Electronics Society (IECON06), ISBN: 1-4244-0136-4, pp. 3373–3378, Paris, November 2006
22. E.B. Koller-Meier, F. Ade, Tracking multiple objects using a condensation algorithm, Journal of Robotics and Autonomous Systems, vol. 34, pp. 93–105, February 2001
23. M. Marrón, M.A. Sotelo, J.C. García, J. Brodfelt. Comparing improved versions of ‘K-Means’ and ‘Subtractive’ clustering in a tracking applications, Proceedings of the Eleventh International Workshop on Computer Aided Systems Theory, Extended Abstracts (EUROCAST07), ISBN: 978-84-690-3603-7, pp. 252–255, Las Palmas de Gran Canaria, February 2007