2008 IEEE International Conference on
Robotics and Automation
Pasadena, CA, USA, May 19-23, 2008

Robot and Obstacles Localization and Tracking with an External Camera Ring

Daniel Pizarro, Marta Marron, Daniel Peon, Manuel Mazo, Juan Carlos Garcia, Miguel A. Sotelo, Enrique Santiso

Abstract— **In this paper a ring of calibrated and synchronized cameras is used for achieving robot and obstacle localization inside a common observed area. To avoid complex appearance matching derived from the wide-baseline arrangement of cameras, a metric occupancy grid is obtained by intersection of silhouettes projected onto the floor. A particle filter is proposed for tracking multiple objects by using the grid as observation data. A clustering algorithm is included in the filter to increase the robustness and adaptability of the multimodal estimation task. To preserve identity of the robot from the set of tracked objects, odometry readings are used to compute a Maximum Likelihood (ML) global trajectory identification. As a proof of concept, real results are obtained in a long sequence with a mobile robot moving in a human-cluttered scene.**

I. INTRODUCTION

In this paper a multiple camera ring is used for achieving robot and obstacle localization. The cameras form up the observation layer of an "Intelligent Space" [12], where a set of agents (i.e. robots, ambient conditions, etc.) are directly controlled by a distributed intelligence. A communication network is available between agents and the space, allowing its mutual interaction. For the "Intelligent Space", the capability of achieving localization of robots and obstacles represents a primary low level objective. The choice of using vision sensors guarantees a rich amount of information from the world to be observed.

To cover the environment with a minimum number of sensors, each camera is placed far from each other with an orientation between them that differs substantially. Under such arrangement, commonly known as 'wide baseline', information is hardly correlated between cameras which usually transfers the difficulties to the vision algorithm.

We propose to combine image information from the set of cameras to robustly track robots and obstacles in the common area of observation. To avoid complex appearance matching derived from the wide baseline arrangement, the images from the multiple cameras are combined in a planar visual hull approach. A 2D occupancy grid is obtained by intersecting image silhouettes of the scene projected onto a common reference plane (i.e. floor plane). By using the grid as a measurement, a modified particle filter (PF) is proposed for sequentially tracking multiple objects. The problem of preserving identity of the robot is solved by using odometry readings instead of relying on appearance methods.

This work was supported by the Ministry of Science and Technology under RESELAI project (reference TIN2006-14896-C02-01)

All Authors are with Dept. of Electronics, University of Alcala, 28805, Alcala de Henares, Spain pizarro@depeca.uah.es

There are several existing approaches that share similar objectives with the present paper, however the special treatment of robots in the environment is rarely addressed in these works, and neither their identification. Most of them are oriented on tracking objects from which an accurate motion model is unknown so the identification relies on observation models or trajectory optimization. In [2] up to six people are robustly tracked from four cameras by using an occupancy grid. The authors propose to use global trajectory optimization and color appearance to solve ambiguities in tracking. The global trajectory adds a four seconds delay which makes it impossible to use in robot navigation. In [10], the same idea is used by detecting space coherent clusters in sequences of grid maps. In [4] two occupancy grids are obtained by a set of cameras and array sound sensors respectively. In [5] and [16] a multiple hypothesis probabilistic approach is used to solve ambiguities and occlusions respectively. In [15] and [9] appearance models are used for solving the identity problem.

Our main contribution can be divided in three main aspects which differentiate from these state-of-the-art works. Firstly the occupancy grid is obtained by computing a probabilistic visual hull where each silhouette is previously weighted by a prior distribution which assumes a cylindrical model. Secondly a modified version of the Bootstrap PF is proposed for tracking multiple objects. The basic PF is enriched by clustering the measurements extracted from the grid in the observation process. Finally, by using existing odometry information a robust robot identification is achieved without using appearance models for each robot or costly batch trajectory optimization.

II. OBSERVATION ALGORITHM

Given a set of N_c cameras, the observation algorithm computes frame by frame an occupancy grid $\rho_t = P(R|B_1, \cdots, B_{N_c})$. The grid represents the conditional distribution for a finite division R of a planar reference frame to be occupied, given the set of binary images $B_i, \cdots, B_{N_c}$ extracted by a very simple background subtraction algorithm.

The cameras are calibrated and the model assumes that the distortion effect has been previously compensated. For each camera an homography 3x3 matrix H_i is obtained, which relates directly metric coordinates in the reference frame $Y = (x, y)^T$ with image plane coordinates $m = (u, v)^T$.

A basic solution consists on projecting each binary image B_i, by using the corresponding homography, onto the final grid. A bilinear interpolation function is used to fill each grid's cell according to the projected image. A simple binary intersection of projected images, is proposed as a rough approximation of the grid distribution. The more the number of views, the closer is the resulting grid to the convex projection of the 3D scene. To improve the performance, where only two or three views are available, it is reasonable to assume a cylindrical model for the obstacles and the robot. Each binary image B_i is decomposed in a set of n_i connected blobs $B_i^1, \cdots B_i^{n_i}$. For each B_i^j a prior distribution $p(m|\theta_i^j, o_i^j)$ function is used to modify the blob in the corresponding image plane before its projection onto the grid.

$$p(m|\theta, o) = \begin{cases} 1 & 0 < u_c \leq \sigma_y \\ e^{-\frac{(u_c)^2}{2\sigma_y^2}} & \sigma_y < u_c \end{cases}, \tag{1}$$

where

$$(u_c, v_c)^T = \begin{pmatrix} cos(\theta) & sin(\theta) \\ -sin(\theta) & cos(\theta) \end{pmatrix} (u, v)^T - o$$

The basic idea is that, given a cylindrical object, there exists a line segment (angle θ and initial point o) in the image plane along which the probability of belonging to the floor decreases. In Fig. 1 it is shown the modified blob corresponding to a human shape.

Each blob B_i^j has different o and θ parameters. The initial offset point o must be a point in the image that truly belongs to the projection plane (floor plane). It is obtained directly by searching the minimum vertical coordinate u of the points of B_i^j. This is usually valid for a wide range of camera orientations similar to those presented in Fig. 2. Given o, its position in the floor Y_o is directly obtained by using the corresponding homography H_i. To obtain θ, an orthogonal line l_0 to the projection plane which includes Y_o is obtained and it is projected back onto the image. It corresponds to the line that rules the 3D height of the object in the image plane. Therefore from point o to the limits of the detected blob B_i^j the probability of belonging to the floor decreases in the direction given by θ, being maximum in o.

In Fig. 3 the result of using the proposed distribution is shown by comparing the grid obtained when only two cameras are used for detecting two obstacles (human and a robot shown in Fig. 2.a and Fig. 2.c. It can be observed in Fig. 3.a that when no prior is applied, each object projection becomes considerably bigger than the ones shown in Fig. 3.b. This effect is due to the projection uncertainty when only two cameras are used. By using a prior model, the grid becomes sharper and closer to the real projection of the objects. The detection process is thus improved if the cylindrical assumption holds.

In Fig. 2 three binary images (red blobs) are combined to form the resulting grid.

III. THE ESTIMATION ALGORITHM

In this paper, the grid ρ_t computed at each time t, is used as the probability distribution $p(Y_t)$ of the floor's occupancy represented as $Y_t = (x_t, y_t)$ in advance. The goal is to infer from $p(Y_t)$, the number N_o and state X_t^i of the obstacles

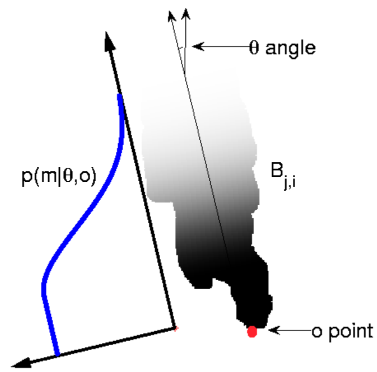
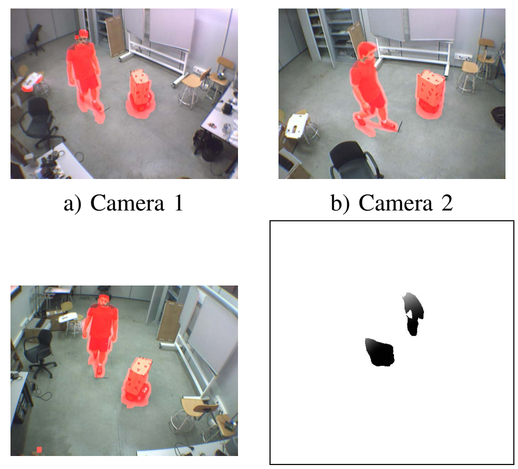


Fig. 1. Modified binary blob by using prior distribution $p(m|\theta, o)$



a) Camera 1 b) Camera 2

c) Camera 3 d) Grid ρ_t

Fig. 2. Occupancy grid from 3 views

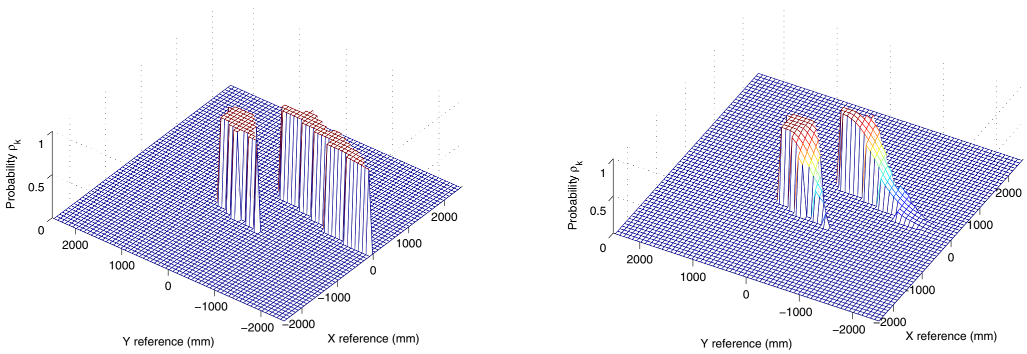
and robots present in the room using a modified version of the Bootstrap Particle Filter (PF) presented in this section.

A. MULTIPLE OBJECTS TRACKER

The task of multiple targets tracking (MTT) appeared with the first autonomous navigation robot, to overcome the obstacle avoidance problem, and soon probabilistic algorithms, such as Kalman filters (KFs) ([21]) and PFs ([8], [18], [20], [11]) were applied to achieve this aim. The objective is in any case to calculate the posterior probability of the state vector $p(X_t|Y_t)$ in the recursive two steps (prediction and correction) standard estimation process by means of the Bayes rule.

The first solution to the MTT application was proposed using a standard estimator to track each object $i = 1, \cdots, N_o$, but such approach could not deal with the interaction of a dynamic number of objects. An association algorithm was included in [6] or in [17] to assign the observation data set Y_t to the correct prior density $p(X_t^i|Y_{1:t-1})$ $i = 1, \cdots, N_o$ in the correction step of the tracking process.

A different approach consists of an expansion of the state vector which includes the state of all elements to track



a) Occupancy grid ρ by intersection

b) Occupancy grid ρ using $p(m|\theta, o)$ on each blob

Fig. 3. Occupancy grid from 2 views

($X_{N_o} = \{X^1, \cdots, X^{N_o}\}$) was firstly proposed in [7] and widely used afterwards as in [8], [18] and [11]. In this case, the estimator used is usually a Particle Filter which obtains not only each object's state vector, but also the number of them being tracked N_o. The complexity of the solution increases exponentially with the state vector size, that is, with the number of objects to track N_o.

In any case most of the MTT solutions are based on the Probabilistic Data Association (PDA) theory [1]. Among them, the best-known is the Joint Probabilistic Data Association Filter (JPDAF) ([17]).

The computational load of the resultant estimator is very high due to the complexity of the PDA. Besides, the basic theory of the probabilistic association does not take into account that the number of objects being tracked N_o can change with time. This point is solved in mostly all vision applications using a gating algorithm supported by the observation model ([21], [19]).

In this context the authors propose in [14] another solution to MTT problem based on a single PF whose multimodality is exploited to perform the tracking task for a variable number of objects with a simple likelihood model. The algorithm is called "eXtended Particle Filter with Clustering Process", (XPFCP). A clustering algorithm is used to organize the measurement distribution $p(Y_t)$ in K_t clusters (C_t^j $j = 1 : K_t$) and insert it wisely in the estimation process. Therefore, the association task is performed implicitly by a mixture of probabilistic and deterministic algorithm that increases the robustness and reliability of the final estimator. This solution has been tested in complex indoor environments with poor sonar data sets ([14]) and with a dense set of position points obtained with a stereo-vision system ([13]) with good results.

In this paper the XPFCP is used as a multimodal tracker to estimate the position (x, y) and speed (v_x, v_y), describing the state vector $X_t^i = (x_t^i, y_t^i, {v_x}_t^i, {v_y}_t^i)^T$ $i = 1, \cdots, N_o$ of the objects detected in the occupancy grid $p(Y_t)$.

B. The XPFCP

In the proposal presented here, the multiple modes of the PF output distribution ($p(X_t|Y_1, \cdots, t)$), characterize the state of a variable number of objects N_o. As commented before, this is more efficient than maintaining one estimator for each hypothesis, or a joint state vector representing all of them. In order to adapt the Bootstrap PF for its use in tracking a variable number of elements, some modifications must be included in the basic algorithm. Here a slight description of the XPFCP is included in Algorithm 1. A deeper explanation can be found in [13].

Algorithm 1 XPFCP General Diagram

Require: Initial distribution of particles $S_{t-1}^i / i = 1 : N_p - N_m$ representing the belief $p(X_{t-1}|Y_{1:t-1})$ at $t-1$

```
for all t > 0 do
    Re-Initialization
        Add N_m/K_t particles into the set randomly taken from
        each cluster C^j_{t-1}/j = 1 : K_t to obtain N_p particles.
    end Re-Initialization
    Prediction
        Propagate through p(X_t|X_{t-1}) the set of N_p particles
        with a constant speed model to obtain S^i_{t|t-1}/i = 1 : N_p.
    end Prediction
    Clustering
        Obtain the set of clusters {C^j_t/j = 1 : K_t} from the
        measurement grid p(Y_t) = ρ_t.
    end Clustering
    Importance Sampling
        for Each particle S^i_{t|t-1}/i = 1 : N_p do
            Search the closest cluster centroid C^j_t/j = 1 : K_t to
            obtain the minimal Euclidean distance d_i
            Compute particles' weight W^i_t = W^i_{t-1} · p(Y|X^i_{t|t-1})
            where p(Y|X^i_{t|t-1}) = e^{-(d_i)^2 / (2σ_x^2)}
        end for
        Normalize W^i_t = W^i_t / ΣW_t where i = 1 : N_p
    end Importance Sampling
    Resampling
        Select N_p − N_m particles using their related weights
        W_t, Ŝ_t to obtain the set S^i_t/i = 1 : N_p − N_m that rep-
        resents the final belief p(X_t|Y_{1:t}).
    end Resampling
end for
```

The XPFCP main contribution is the inclusion of a clustering algorithm to improve the behavior of a PF. In the work presented here a modified K-Means that adapts itself to a variable number of clusters is used. The clustering algorithm organizes the silhouettes in ρ_t in K_t sets $(C_t^j/j = 1 : K_t)$ that are then used in the tracking algorithm. A validation process is added to the K-Means in order to keep a peak in the likelihood in situations of temporal occlusions of the tracked objects. This fact is specially interesting in the present application to prevent from missing an object due to shadows.

With the clustered data $(C_t^j/j = 1 : K_t)$ two important innovations are included in the Bootstrap PF in order to facilitate the multi-tracking process:

1) Re-initialization step: In this step previous to the prediction one, N_m from the N_p total number of particles that formed the belief distribution at $t-1$ in the PF $(p(X_{t-1}|Y_{1:t-1}))$ are directly taken from the input data set Y_{t-1} and included as particles in the set $S_{t-1}^i/N_p - N_m =: N_p$. With this procedure, newly appearing objects in the scene have a representation in the belief. To improve the robustness of the estimator, the inserted data are not selected randomly from the array of blobs ρ_t, but from its segmentation $(C_t^j/j = 1 : K_t)$. Choosing points from all K_t clusters ensures a probable representation of all N_o objects in the scene, and therefore, an increased robustness of the multi-tracker. Thanks to this re-initialization step the posterior distribution dynamically adapts itself to the variable number of objects present in the scene.

2) Resampling step: This step is also modified from the Bootstrap PF. On one hand, only $N_p - N_m$ samples of the prior density $(p(X_t|Y_{1:t-1}))$ have to be extracted in this step, as the N_m resting particles would be inserted in the re-initialization. On the other hand, the clustering process is also used in this step, because the importance sampling function $p(Y_t|X_t^i)/i = 1 : N_p$ used to weight each particle is obtained from the similarity between each $(S_{t|t-1}^i/i = 1 : N_p)$ and the centroid of the clustered set of blobs $(c_t^j/j = 1 : K_t)$. Using cluster centroids to weight the particles related to the newly appeared objects, the probability of these last ones in the belief $(p(X_t|Y_{1:t}))$ is increased, improving the robustness of the final estimator. Without the clustering process, particles related to new objects have low probability at the end of the resampling step, and are rejected from the belief in the selection step. Thus, the multimodality of the PF cannot be exploited.

The robustness problem has been the main reason in the researching community for not using the PF multimodal capability. Only a few works can be found in the related literature with this same idea. In [20] a gaussian mixture is used to model each target behavior $X_t^i/i = 1 : N_o$, and an unique set of particles $S_t^i/i = 1 : N_p$ is distributed among all gaussians in order to propagate their state in time. Therefore, a set of almost independent PFs is implemented, and therefore the multimodality is not really exploited. On the other hand, a clustering is included in this mixture PF, but it is done over the particle set in order to develop a merge and split step. The clustering is not used with the aim of increasing the tracker robustness in a variable number of targets N_o application, as described here.

With the XPFCP proposed, the MTT problem is solved robustly at the same time than the association task, with a constant and low execution time for a variable number of objects, which is essential for its real time execution. Moreover, the algorithm has demonstrated its feasibility in the tracking application within different observation systems, and using simple observation models which increases the flexibility and robustness of the MTT and decreases the global application execution time. Localizing a specific track in the environment can be achieved afterwards using its particular model, as demonstrated in the following paragraph.

IV. Identity of robots from motion

In this section we propose a method to identify which particles belong to any robot controlled by the "Intelligent Space" from the rest of obstacles appearing in the grid.

Usually, as was mentioned before, some appearance model is used in combination with the natural frame-to-frame neighboring, or a costly global trajectory identification. In most of such state-of-the-art works, a real and confident motion model of the object tracked is never available. However in this case, a very accurate one in short trajectories is known through odometry readings from the robot. Usually encoder sensors produce an estimation of the trajectory which suffers from unbound uncertainty with path length. In this case a simple likelihood will be computed to identify which particles at the XPFCP output characterize the belief peak related to the robot in some degree. The likelihood therefore must show the same uncertainty properties that the real motion encoders show. In this section a propagated Gaussian is used for that purpose.

A. Motion model

State vector encoding robot pose at time t is represented by X_t. It is described by a Markov Process with initial distribution $p(X_0)$ and transition kernel $p(X_t|X_{t-1})$.

The transition kernel is derived from a motion model in which odometry uncertainty is included. For simplicity, motion model used in this paper corresponds to a simple wheeled differential robot which moves over a ground plane. Pose parameters $X_t = (r_t, \phi_t)$ will be composed of position in the ground plane r_t with respect a global coordinate origin O and orientation ϕ_t:

$$X_t = X_{t-1} + \begin{pmatrix} (vl_t + w_t^1)\cos(\phi_{t-1} + \Omega_t + w_t^2) \\ (vl_t + w_t^1)\sin(\phi_{t-1} + \Omega_t + w_t^2) \\ \Omega_t + w_t^2 \end{pmatrix}, \quad (2)$$

Input from odometry is represented by the vector $U_t = (vl_t, \Omega_t)$ with linear and angular speed components. Noise from odometry readings is modeled by a Gaussian process $(w_t^1, w_t^2)^t$ which is statistically independent with respect to X_{t-1}.

B. Motion uncertainty

Given the process described by the transition equation (2) and initial distribution $p(X_0)$ we search here for the Gaussian equivalent of $p(X_{1:t}|X_0)$.

Gaussian approximation of the distribution $p(X_t|X_0) = N(\hat{X}_t, \Sigma_t)$ is recovered by using a first order approximation of (2). By combining as input $p(X_{t-1}|X_0)$ and the actual noise $(w_t^1, w_t^2)^t$, the resulting Gaussian for $p(X_t|X_0)$ is a growing covariance process usually present in Dead-Reckoning [3] processes (See Figure 4).

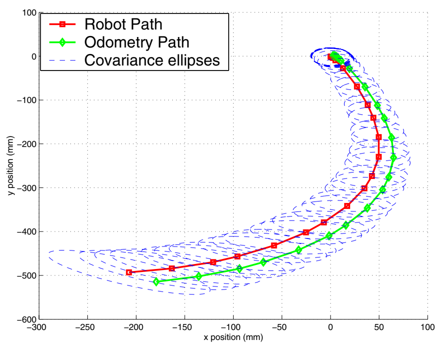


Fig. 4. Uncertainty Growing in robot position

C. Trajectory Likelihood

The joint distribution, simply obtained by

$$p(X_{t-N_f:t}|X_0) = \prod_{i=t-N_f}^{t} p(X_i|X_0),$$

is a Gaussian process $N(\mu, \Sigma)$ defined in N_f different poses.

The main proposal of this section is to use $p(X_{t-N_f:t}|X_0)$ to compute a likelihood, so that given a set of trajectories from different objects, the one which is closer to the real motion of the robot obtains a maximum score.

By using the Mahalanobis distance, the likelihood is converted into a norm, and the ML is converted into a minimum distance criterion.

Given a set of trajectories $\{T^1_{t-N_f:t}, \cdots, T^{N_o}_{t-N_f:t}\}$, the most probable one generated by the robot is the one which minimizes the following expression:

$$d_M = (T^i - \mu)^T (\Sigma)^{-1} (T^i - \mu) \tag{3}$$

Here we assume that either T^i and μ are column vectors with $3 \cdot N_f$.

The criterion for choosing the size of N_f must be a combination of the following constraints:

- By using a bound on the uncertainty of the last pose $p(X_t|X_0)$ there exists a maximum number of frames N_f to choose from the past.
- The minimum distance d_M obtained must be confident with the distribution $\chi(n)^2$ derived from the Mahalanobis test. A confident ratio must be imposed so that the best trajectory fit the $\chi(n)^2$ distribution in some high degree ($> 90\%$).
- The second minimum distance must be far to the first in some degree which can be imposed again in terms of probability (15% less probable at least).

D. Usage with the XPFCP output

To use the ML algorithm described, the output set of particles S_t is also clustered in N_o groups, and the position c_t^i of the $i = 1 : N_o$ different clusters is computed. As the pose X_t used in this section requires also the orientation of the object, a rudimentary orientation estimation is used on each object by using consecutive frames. Once computed, the set of candidate trajectories are tested using (3).

V. EXPERIMENTAL SETUP

The proposals made in this paper were tested using a real environment with calibrated cameras and a mobile robot platform. The cameras are low cost CCD sensors with 640x480 pixels of resolution. The adquisition and processing cluster performs synchronization by using a local network between its nodes. Calibration is made previously with a chessboard pattern and the room floor is used as a common reference plane.

The robot platform has odometry sensors and a wireless network for receiving commands from the environment and for sending odometry readings to the system. Odometry noise is tuned by using a close loop experiment.

The experiments use 3 cameras inside a room with humans crossing around the robot. The system achieves good detection without requiring many parameters to tune. Trajectory identification is done each time the conditions commented before are reached and bounding the uncertainty to a radius of $70cm$

In Fig. 5 the trajectories of three humans and the robot are shown compared to odometry information. In Fig. 6 it can be seen a long trajectory tracking for the robot and humans using 3 cameras (see accompanying video). A manual ground truth predicts an accuracy near $10cm$ which is enough for robot positioning.

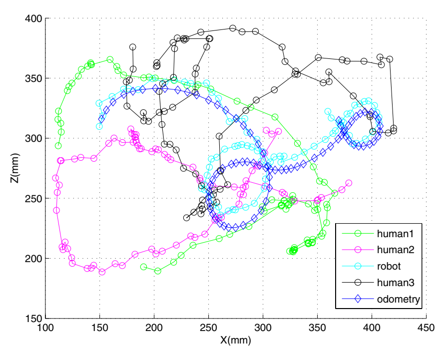


Fig. 5. Path comparison between odometry, robot and human

VI. CONCLUSIONS AND FUTURE WORKS

In this paper a complete robot and obstacle tracking system is proposed by using a ring of calibrated cameras in wide baseline distribution. The system is flexible and robust, by combining a very simple but powerful visual hull approach and by using a probabilistic approach for tracking. A robot identification methodology is proposed which avoids relying

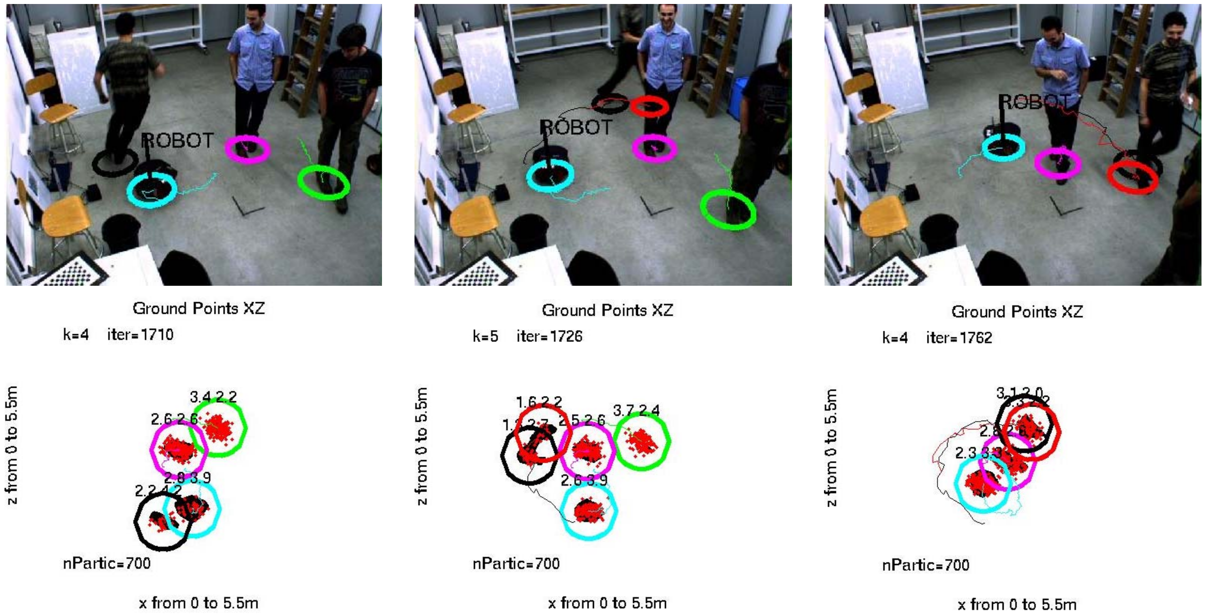


Fig. 6. Localization in a real environment

in complex and risky appearance models. Instead, it is based on the capability of the "Intelligent Space" to control the robot and the extra odometry information.

The results included in the paper show that the MTT performs well in real environments where multiple people are crossing near the robot and there are multiple occlusions. Besides, the multimodal estimator proposed solves the association and tracking task for a variable number of objects in a constant and real time, and faces robustly the overlapping silhouettes and shadows problems of the observation system.

The weakest point of the algorithm comes from he fact that a background extraction method is used for detecting objects. In this paper a very rudimentary algorithm is proposed that in a near future must be replaced by a state-of-the-art method.

References

[1] Y. Bar-Shalom and T. Fortmann. *Tracking and data association. Mathematics in Science and Engineering, V.182*. Academic Press, January 1988.

[2] J. Berclaz, F. Fleuret, and P. Fua. Robust people tracking with global trajectory optimization. *Proceedings of the Fifth IEEE Conference on Computer Vision and Pattern Recognition (CVPR06)*, 1:744–755, 2006.

[3] J. Borenstein and L. Feng. Measurement and correction of systematic odometry errors in mobile robots, 1996.

[4] N. Checka, K. Wilson, V. Rangarajan, and T. Darrell. A probabilistic framework for multi-modal multi-person tracking. *cvprw*, 09:100, 2003.

[5] D. Focken and R. Stiefelhagen. Towards vision-based 3-d people tracking in a smart room. In *Proceedings of the Fourth IEEE International Conference on Multimodal Interfaces*, pages 400–405, October 2002.

[6] C. Hue, J.P. Le Cadre, and P. Perez. A particle filter to track multiple objects. In *Proceedings of the IEEE Workshop on Multi-Object Tracking, ISBN: 0-7695-1171-6*, pages 61–68, August 2001.

[7] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, 1998.

[8] M. Isard and J. MacCormick. Bramble: A bayesian multiple-blob tracker. In *Proceedings of the Eighth IEEE International Conference on Computer Vision (ICCV01)*, volume 2, pages 34–41, July 2001.

[9] J. Kang, I. Cohen, and G. Medioni. Tracking people in crowded scenes across multiple cameras. *Asian Conference on Computer Vision*, 2004.

[10] S.M. Khan and M. Shah. A multiview approach to tracking people in crowded scenes using a planar homography constraint. *Proc. European Conf. Comp. Vis*, 2006.

[11] Z. Khan, T. Balch, and F. Dellaert. Mcmc-based particle filtering for tracking a variable number of interacting targets. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 27, pages 1805–1918, November 2005.

[12] J.H. Lee, K. Morioka, and H. Hashimoto. Intelligent space and mobile robots. In *The Industrial Information Technology Handbook*, pages 1–15. 2005.

[13] M. Marron, J.C. Garcia, M.A. Sotelo, D. Fernandez, and D. Pizarro. Xpfcp: An extended particle filter for tracking multiple and dynamic objects in complex environments. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 234–239, August 2005.

[14] M. Marron, M.A. Sotelo, and J.C. Garcia. A probabilistic multimodal algorithm for tracking multiple and dynamic objects. In *World Automation Congress 2004 (WAC04)*, volume 4, June 2004.

[15] A. Mittal and L.S. Davis. M2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene using region-based stereo. In *Proceedings of the Seventh European Conference on Computer Vision (ECCV02)*, pages 18–36, 2002.

[16] K. Otsuka and N. Mukawa. Multiview occlusion analysis for tracking densely populated objects based on 2-d visual angles. In *Proceedings of the Third IEEE Conference on Computer Vision and Pattern Recognition (CVPR04)*, volume 1, pages 90–97, June 2004.

[17] C. Rasmussen and G.D. Hager. Probabilistic data association methods for tracking complex visual objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):560–576, 2001.

[18] D. Schulz, W. Burgard, D. Fox, and AB Cremers. People tracking with anonymous and id-sensors using rao-blackwellised particle filters. In *Proceedings of the Eighteeenth International Joint Conference on Artificial Intelligence (IJCAI03)*, August 2003.

[19] K. Smith, D. Gatica-Perez, and J.M. Odobez. Using particles to track varying numbers of interacting people. In *Proceedings of the Fourth IEEE Conference on Computer Vision and Pattern Recognition (CVPR05)*, volume 1, pages 962–969, June 2005.

[20] J. Vermaak, A. Doucet, and P. Perez. Maintaining multimodality through mixture tracking. In *Proceedings. Ninth IEEE International Conference on Computer Vision (CVPR03)*, pages 1110–1116vol.2, 13-16 Oct. 2003.

[21] T. Zhao and R. Nevatia. Tracking multiple humans in crowded environment. *Proceedings of the Third IEEE Conference on Computer Vision and Pattern Recognition (CVPR04)*, 2:406–413, 2004.