Detection Model in Collaborative Multi-Robot Monte Carlo Localization.

R. Barea, E. López, L.M. Bergasa, S. Álvarez, M. Ocaña

Department of Electronics
University of Alcalá
Alcalá de Henares. Madrid. Spain

barea@depeca.uah.es

***Abstract* - This paper presents an algorithm for collaborative mobile robot localization based on probabilistic methods (Monte Carlo localization) used in assistant robots. When a root detects another in the same environment, a probabilistic method is used to synchronize each robot´s belief. As a result, the robots localize themselves faster and maintain higher accuracy. The technique has been implemented and tested using a virtual environment capable to simulate several robots and using two real mobile robots equipped with cameras and laser range-finders for detecting other robots. The result obtained in simulation and with real robots show improvements in localization speed and accuracy when compared to conventional single-robot localization.**

Index Terms* - *Mobile robots, Collaborative multi-robot localization, Monte Carlo Localization, Assistant robots.

I. INTRODUCTION

Mobile robot localization is the problem of estimating a robot's pose (location, orientation) relative to its environment [1]. The localization problem is a key problem in mobile robotics and plays a pivotal role.

The mobile robot localization problem comes in many different flavors [2]. The most simple localization problem is position tracking. Here the initial robot pose is known, and the problem is to compensate incremental errors in a robot's odometry. More challenging is the global localization problem [3], where a robot is not told its initial pose but instead has to determine it from scratch. The global localization problem is more difficult, since the error in the robot's estimate cannot be assumed to be small. Consequently, a robot should be able to handle multiple, distinct hypotheses. More difficult is the kidnapped robot problem [4], in which a well-localized robot is tele-ported to some other place.

Probabilistic methods have been applied with remarkable success to single-robot localization [5,6,7], where they have been demonstrated to solve problems like global localization and localization in dense crowds. The global localization and kidnapped robot problem in a highly robust and efficient way can be overcome using Monte Carlo localization (MCL) algorithm [1]. Monte Carlo Localization is a family of algorithms for localization based on particle filters, which are approximate Bayes filters that use random samples for posterior estimation. Recently, they have been applied with great success for robot localization.

On the other hand, assistant robots have received special attention from the research community in the last years. One of the main applications of these robots is to perform care tasks in indoor environments such as houses, nursing homes or hospitals, and therefore, they need to be able to navigate robustly for long periods of time [7,8,9].

This paper is focused in a probabilistic algorithm for collaborative mobile robot localization based on Monte Carlo localization. The key idea of multi-robot localization is to integrate measurements taken at different platforms, so that each robot can benefit from data gathered by robots other than itself. Therefore, When one robot detects another, this information are used to synchronize the individual robots' beliefs, thereby reducing the uncertainty of both robots during localization [10].

II. MULTI-ROBOT MONTE CARLO LOCALIZATION

Monte Carlo Localization have been widely studied in [1,8]. MCL is a recursive Bayes filter that estimates the posterior distribution of robot poses conditioned on sensor data.

The key idea of Bayes filtering is to estimate a probability density over the state x space conditioned on the data. This posterior is typically called the *belief* and is denoted:

$$Bel(x_t) = p\left(x_t \middle| \; o_t, a_{t-1}, o_{t-1}, a_{t-2}, \ldots, o_0\right) \qquad (1)$$

Where x_t is the state at time t, o denotes *observations* (*perceptual data* such as laser range or vision measurements) and a represents *actions* (*odometry data* which carry information about robot motion).

Bayes filters estimate the belief recursively. The initial belief characterizes the initial knowledge about the system state. In the absence of such knowledge, it is typically initialized by a uniform distribution over the state space. In mobile robot localization, a uniform initial distribution corresponds to the global localization problem, where the initial robot pose is unknown.

To derive a recursive update equation, (1) can be transformed by Bayes rule to:

The authors would like to express their gratitude to the Comunidad de Madrid and the University of Alcalá for their support through the projects RoboCity2030 (CAM-S-0505/DPI/000176) and SIMCA (CAM-UAH-2005/018).

0-7695-2589-X/06 $20.00 © 2006 IEEE

IEEE COMPUTER SOCIETY

$$Bel(x_t) = \eta \cdot p(o_t|x_t, a_{t-1}, \ldots, o_0)\, p(x_t|x_{t-1}, a_{t-1}, \ldots, o_0) \quad (2)$$

$$\eta = p(o_t|a_{t-1}, \ldots, o_0)^{-1} \quad (3)$$

Bayes filters assume that the environment is Markov, that is, past and future data are (conditionally) independent if one knows the current state. the Markov assumption implies:

$$p(o_t|x_t, a_{t-1}, \ldots, o_0) = p(o_t|x_t) \quad (4)$$

$$p(x_t|x_{t-1}, a_{t-1}, \ldots, o_0) = p(x_t|x_{t-1}, a_{t-1}) \quad (5)$$

Therefore, the belief can be denoted by:

$$Bel(x_t) = \eta \cdot p(o_t|x_t) \cdot \int p(x_t|x_{t-1}, a_{t-1}) Bel(x_{t-1}) dx_{t-1} \quad (6)$$

Where $p(o_t/x_t)$ is called *perceptual model* and $p(x_t/x_{t-1},a_{t-1})$ represents the *motion model*.

The key idea of multi-robot localization is to integrate measurements taken at different platforms, so that each robot can benefit from data gathered by robots other than itself. Therefore, when a robot *n* is detected by robot *m* it is necessary to introduce the detection model according with data obtained r_m in (6). In the absence of detections, the Markov localization works independently for each robot. A summary of the multi-robot Markov localization algorithm is:

- Initialize the belief $Bel_n(x)$ according with initial data (typically uniform distribution).
- If the robot *n* receives an observation on (new sensory input) o_n, it is applies the perception model:

$$Bel_n(x) = \eta \cdot p(o_n \mid x) \cdot Bel_n(x) \quad (7)$$

- If the robot n do some action an (receives a new odometry reading), It is applies the motion model:

$$Bel_n(x') = \eta \cdot \int_x p(x' \mid x, a_n) \cdot Bel_n(x) \cdot dx \quad (8)$$

- And finally, if the robot n is detected by the m-th robot it is applies the detection model

$$Bel_n(x') = \eta \cdot Bel_n(x) \int_x p(x_n = x' \mid x_m = x, r_m) \cdot Bel_m(x) \cdot dx \quad (9)$$

The idea of MCL is to represent the *belief* by a set of *m* weighted samples distributed according to *Bel(x)*:

$$Bel(x_t) \approx \{x^i, w^i\}_{i=1,\ldots,m} \quad (10)$$

Whre x^i is a *sample* of the random variable *x* (pose) and w^i is called *importance factor* and represents the importance of each sample. The set of samples, thus, define a discrete probability function that approximates the continuous belief *Bel(x)*.

The initial set of samples represents the initial knowledge $Bel(x_o)$ about the state of the dynamical system. For instance, in global mobile robot localization, the initial belief is a set of poses drawn according to a uniform distribution over the robot's universe, annotated by the uniform importance factor 1/m. If the initial pose is known up to some small margin of error, $Bel(x_o)$ may be initialized by samples drawn from a narrow Gaussian centered on the correct pose.

The recursive update is realized in three steps:

- Sample $x^i_{t-1} \sim Bel(x_{t-1})$. Each such particle x^i_{t-1} is distributed according to the belief distribution $Bel(x_{t-1})$.
- Sample $x^i_{t-1} \sim p(x_t/x^i_{t-1}, a_{t-1})$. In this case, x^i_t is distributed according to the product distribution $p(x_t/x_{t-1},a_{t-1}) \cdot Bel(x_{t-1})$.
- The importance factor is assigned to the i-th sample:

$$w^i = \eta \cdot p(o_t/x^i_t) \quad (11)$$

III. General Arquitecture

Our goal is to develop a workbench for testing collaborative mobile robot localization based on Monte Carlo localization. For this, we have implemented a testing platform that permit simulate and work with real robots in an indoor environment (Fig 1).

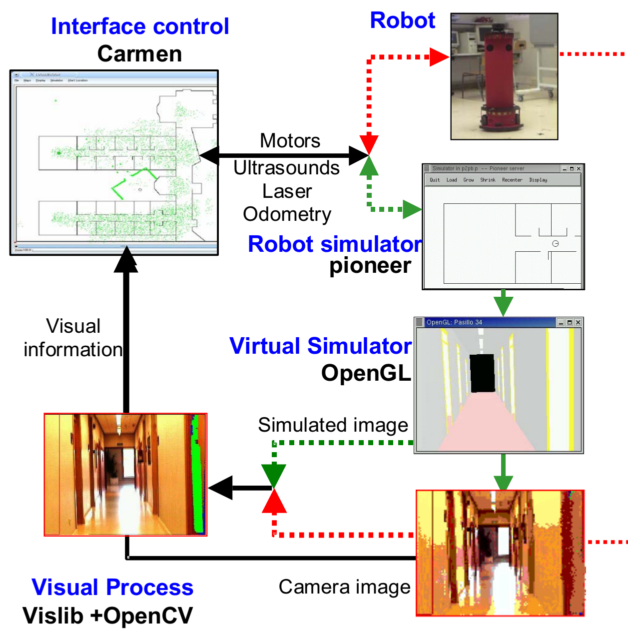

Fig. 1. General architecture.

A. Robots

We have two robotic platform (based on PeopleBot and pioneer AT robots of ActivMedia Robotics [11]). Its architecture is composed of four large modules: environment perception, navigation, human-machine interface and high-level services as we show in Fig. 2. The perception module is endowed with encoders, bumpers, sonar ring, laser sensor and a vision system based on a PTZ (pan-tilt-zoom) color camera connected to a frame grabber. The human-machine interface is composed of loudspeakers, microphone, a tactile screen, the same PTZ camera used in the perception module, and wireless Ethernet link. The system architecture includes two human-machine interaction systems, such as voice (synthesis and recognition speech) and touch screen for simple command selection (for example, a destination room to which the robot must go to carry out a service task). The high-level services block controls the rest of the modules and includes several tasks of tele-assistance, tele-monitoring, providing reminding and social interaction [12].

0-7695-2589-X/06 $20.00 © 2006 IEEE

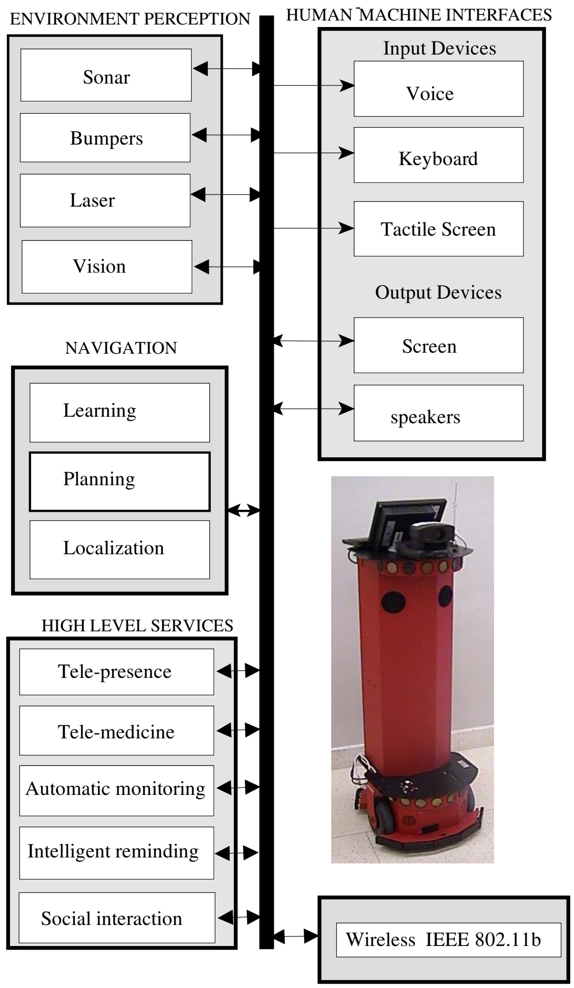

Fig. 2. General architecture of the robots.

B. Navigation module

The navigation module combines information from perception module for carrying out different tasks.

The core of this module is CARMEN (Carnegie Mellon Robot Navigation Toolkit) [13] which is an open-source collection of software for mobile robot control. CARMEN is modular software designed to provide basic navigation primitives including: base and sensor control, obstacle avoidance, localization, path planning, people-tracking, and mapping.

This source has been modified to implement the multi-robot localization, because CARMEN only permits works with a single robot, and different initial distribution for studying the robot localization. This source implements the motion, perception and detection models. Besides, a virtual simulator has been developed for testing the detection model using visual information and the localization process.

B.1. Detection model

Robots must possess the ability to sense each other. The detection model describes the probability that robot n is at location x, given that robot m is at location x´ and perceives robot n with measurement r_m.

To determine the relative location of other robots, our approach combines visual information obtained from an on-board camera with proximity information coming from a laser range-finder. Camera images are used to detect other robots and together with laser range-finder scans are used to determine the relative position of the detected robot and its distance.

The robots are marked by a unique and coloured marker (Fig. 3) to facilitate its recognition (green cylinder). This way, the marker can be detected regardless of the robot's orientation.

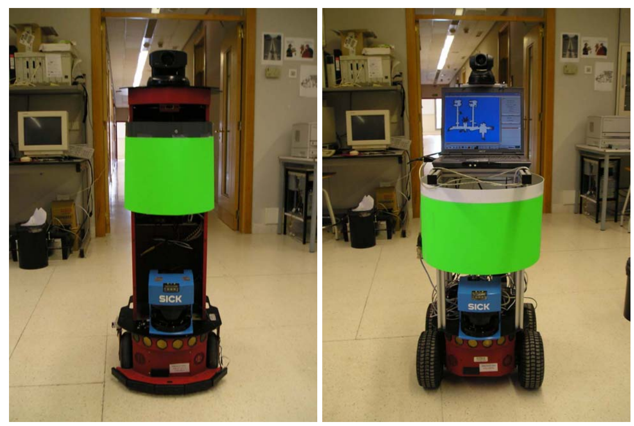

Fig 3. Robots with colored marker.

To find robots in a camera image, our system first filters the image by employing local colour histograms (HIS space colour). Thresholding is then employed to search for the marker's characteristic colour transition. If found, this implies that a robot is present in the image.

Once a robot has been detected, we process de object detected (size and position in the image) and the laser scan for calculating the relative location of the robot. This multi-sensor technique has been proved robust in practical test because the detection rate in field of view is superior to 99%. Currently, images are analyzed approximately at a rate of 10Hz, this rate is sufficient for the application at hand.

Fig. 4 shows examples of camera images recorded in our laboratory and simulated images and the process followed.

When a robot detects to the other one a detection model is generated (usually type gaussian) that it represents the probability that the detected robot is in this point. This detection model is introduced in the equation 9 to carry out the adjustment of the particles by means of Collaborative Monte Carlo localization.

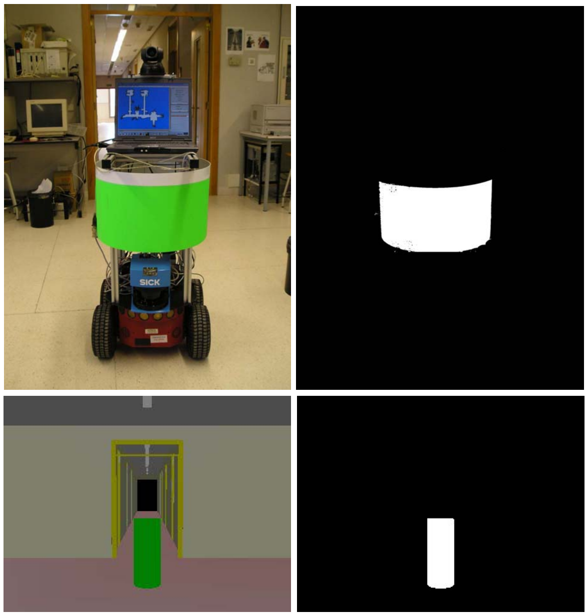

Fig. 4. Robots detection in real environment and simulation.

Next an example of two robots' collaborative localization is commented in a working environment (Fig. 5). Robot 1 are located in the position [x, y] = [6, 21] with a gaussian distribution of particles shown in figure 6. On the other hand, robot 2 are really in the position [10, 21] and it has an uniform distribution of particles in the whole environment (see Fig. 7). If we consider the robot's position like the mean value of the position of the particles, we obtain that the robot 1 is in position [6.0481, 21.0060] and robot 2 in [24.1551, 13.2715]. It can be observed that robot 1 are practically well located while robot 2 are very bad located.

When robot 1 detects robot 2, a gaussian detection model is generated centered in the detected position (Fig. 8). This model is sent to robot 2 and the particles weights are adjusted in function of this model giving place to a new localization of the particles (Fig. 9). It can be observed that detection model allows to reduce the uncertainty vastly on the localization of robot 2. Now the robot 1's position is [x,y] = [6.0481, 21.0060] and for robot 2 is [x,y] = [9.8819, 21.1525]. Now, the new position of robot 2 practically it coincides with the real one.

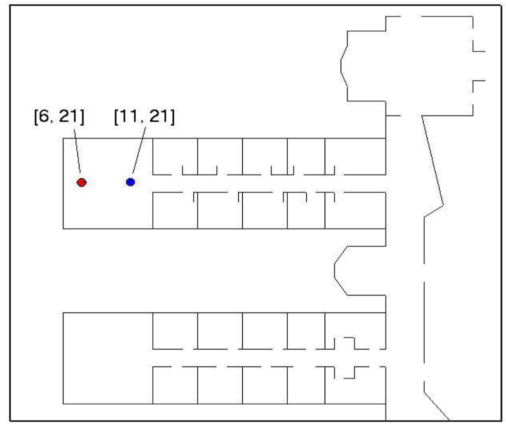

Fig 5. Robots position.

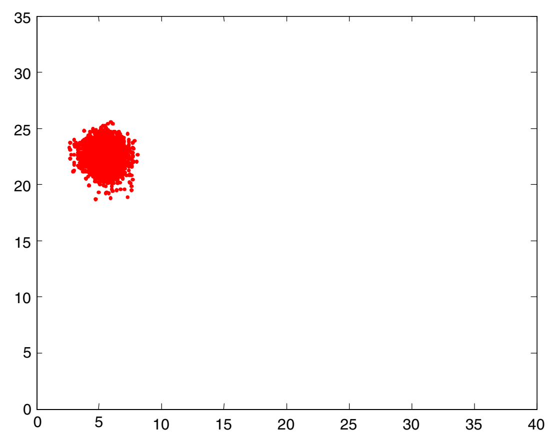

Fig 6. Robot 1 [6, 21] with gaussian distribution.

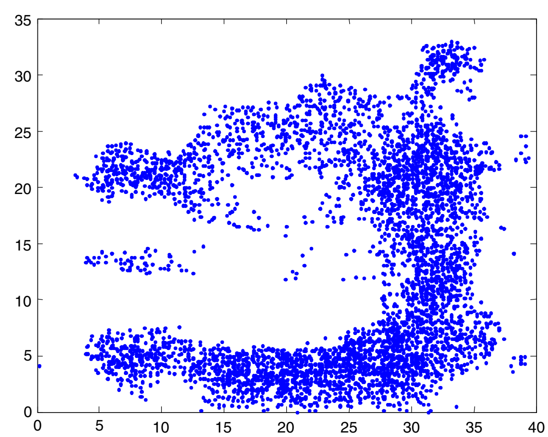

Fig 7. Robot 2 [10, 21] with uniform distribution.

Proceedings of the IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06)
0-7695-2589-X/06 $20.00 © 2006 IEEE

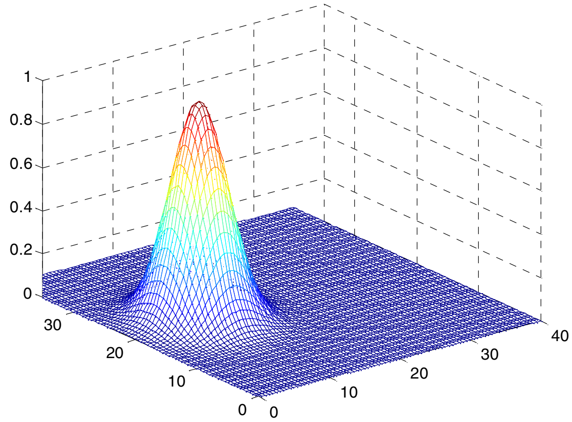

Fig 8. Detection model.

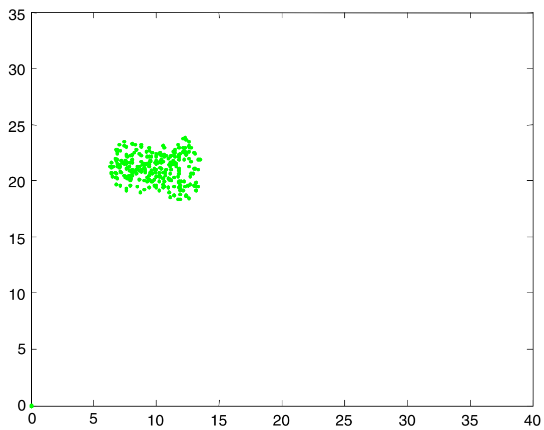

Fig 9. Robot 2 particle distribution after detection.

IV. Results

This section shows results obtained with simulated robots. The objective is to show how cooperative multi-robot localization improves the individual localization process.

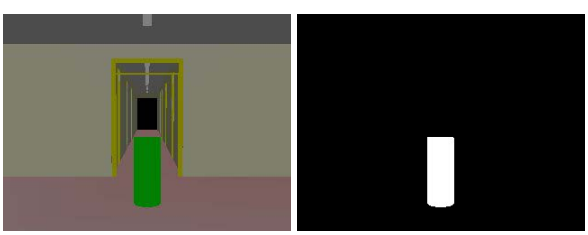

Fig. 10. Robots detection in simulation.

In the following experiments it is used a simulation tool (navigation tool from CARMEN) which simulates robots on the sensor level, providing raw odometry and proximity measurements. Besides, a visual simulator using OpenGL have been developed for simulating the camera image.

Robot detections have been simulated by using the positions of the robots extracted from the map and the camera image simulated extracted from visual simulator.

In the simulation experiments we use two robots, which are all equipped with laser and camera sensors. The task of the robots is to perform global localization in the environment of the Department of Electronics composed by laboratories, offices, corridors and halls.

For studying the multi-robot localization we work with two robots fixed in the laboratory (robot 1) and in an office 2 (robot 2). The objective is to test the global localization of the robot 1 (with initial uniform belief) with different initial belief of robot 2.

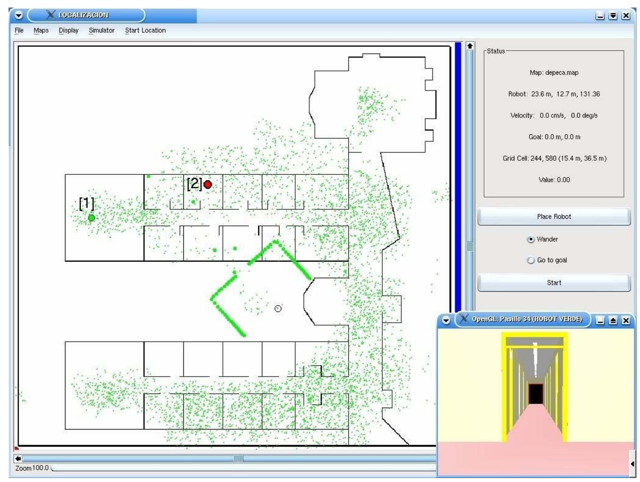

Fig. 11. Initial uniform belief´s robot.

Firstly, we suppose robot 2 is located but it has a great uncertainty. This can be modelled using a gaussian belief with a high standard deviation σ = 0.8 (Fig. 12). Now, when robot 2 detects robot 1, the robot 1 upgrades its belief, it can be see that only exists particle in the top horizontal corridor. Therefore, the robot knows in which corridor is but it unknows where is with accuracy, but before reaches the end of the corridor, it is localized himself correctly (Fig. 13).

On the other hand, if we suppose robot 2 is located with accuracy, we can model its belief using a gaussian with narrow standard deviation σ = 0.2 (Fig. 14). In this case, when robot 2 detects robot 1 in the corridor, the belief´s robot 1 is upgraded with precision and it know where is in the corridor with accuracy. When robot 1 is located in the map we can send navigation task, such as, to reach a goal (Fig. 15).

Proceedings of the IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06)
0-7695-2589-X/06 $20.00 © 2006 IEEE

Fig. 12. Initial gaussian belief for robot 2 (σ = 0.8).

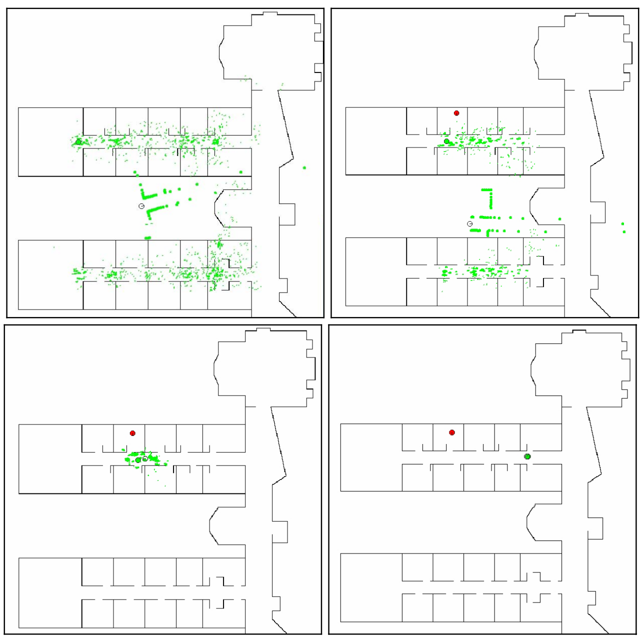

Fig. 13. Belief´s robot 1 (Robot 2 with gaussian belief with σ = 0.8).

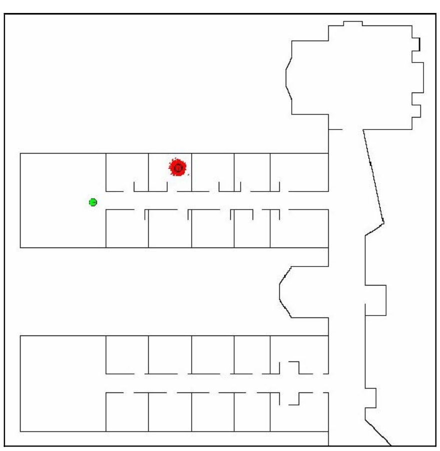

Fig. 14. Initial gaussian belief for robot 2 (σ = 0.2).

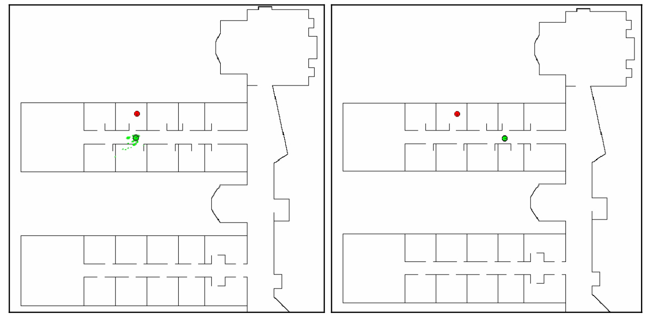

Fig. 15. Belief´s robot 1 (Robot 2 with gaussian belief with σ = 0.2).

V. CONCLUSIONS

The results obtained using collaborative multi-robot Monte Carlo localization shows improvements in localization speed and accuracy when compared to conventional single-robot localization. This way, if we have a robots´ fleet working in an environment, this system permits to upgrade the localization module of each robot when detecting another, and therefore, each robot is located in a robust and effective way.

REFERENCES

[1] S. Thrun, D. Fox , W. Burgard and F. Dellaert, "Robust Monte Carlo Localization for Mobile Robots," Artificial Intelligence, 2001.

[2] J. Borenstein, B. Everett, and L. Feng, "Navigating Mobile Robots: Systems and Techniques," A. K. Peters, Ltd., Wellesley, MA, 1996.

[3] W. Burgard, A. Derr, D. Fox, and A.B. Cremers, "Integrating global position estimation and position tracking for mobile robots: The dynamic markov localization approach," In Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'98), 1998.

[4] D. Fox, W. Burgard, and S. Thrun, "Markov localization for mobile robots in dynamic environments," Journal of Artificial Intelligence Research, 11:391–427, 1999.

[5] I.J. Cox and J.J. Leonard, "Modeling a dynamic environment using a bayesian multiple hypothesis approach," Artificial Intelligence, 66:311–344, 1994.

[6] D. Fox, W. Burgard, and S. Thrun, "Active markov localization for mobile robots," Robotics and Autonomous Systems, 25(3-4):195–207, 1998.

[7] E. López, L.M. Bergasa, R. Barea and M. Escudero. "A navigation system for assistant robots using visually augmented POMDPs," Autonomous Robots, 19: 67-87, 2005

[8] Nursebot Project. http://www-2.cs.cmu.edu/~nursebot

[9] Morpha Project. http://www.morpha.de

[10] D. Fox, W. Burgard, H. Kruppa and S. Sebastian Thrun, "A Probabilistic Approach to Collaborative Multi-Robot Localization," Special issue of Autonomous Robots on Heterogeneous Multi-Robot Systems, 8(3), 2000.

[11] ActivMedia Robotics. http://www.robots.activmedia.com

[12] R. Barea, L.M. Bergasa, E. López, M.S. Escudero, J.A. Hernández and Y. Willemaers, "Tele-medicine system based on a personal robotic assistant," 10th IEEE Internacional Conference on Methods and Models in Automation and Robotics MMAR´04, 2004

[13] Carnegie Mellon Robot Navigation Toolkit. http://www.cs.cmu.edu/~carmen/index.html