Patient monitoring in health care working with robotic assistants

R. Barea, L.M. Bergasa, *Member IEEE*, E. López, M. Ocaña, D. Schleicher and A. León

***Abstract*— The main objective of this work is to develop a robots´ fleet working together to make assistance tasks in a hospital or at home in a collaborative way. This paper presents a method to detect, recognize and track people using mount cameras fixed on a building and an algorithm for collaborative mobile robot localization based on probabilistic methods (Monte Carlo localization). The performance of this system has been tested successfully. Some experimental results and conclusions are presented.**

I. INTRODUCTION

Health care assistants carry out the routine care tasks required to look after patients who are staying in hospital or at home. It is hoped that in the future teams of robots working together may be used to help on hospitals or geriatrics and free staff to spend more time with patients.

The developed countries are facing an explosion of costs in the health-care sector for elderly. Current nursing home costs range between $30,000 and $60,000 annually. Over the last decade along, costs have more than doubled. The dramatic increase of the elderly population along with the explosion of costs poses extreme challenges to society. The current practices of providing care for the elderly population are already insufficient. Undoubtedly, this problem will multiply over the next decade.

The society needs to find new technologies and alternative ways of providing care to the elderly, where the need for personal assistance is larger than in any other age group. Several factors suggest that now is the time to establish new applications in the home-care sector: firstly, we actually have the technology (internet) to develop new applications in home health-care sector. Secondly, at the currently, the robots exhibit the necessary robustness, reliability, and level of capability. Thirly, the need for cost-effective solutions in the elderly care sector is larger than ever before. For this reason, 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. Nowadays exists several research groups working in this area, and some important projects such us "Nursebot" [1] and "Morpha" [2]. Basically, the goal of these projects is the development of personal robotic aids that serves high level services specially though for assisting the elderly such us tele-presence, tele-medicine, intelligent reminding, safeguarding, mobility assistance and social interaction [3].

In order to offer a solution for described problem, a research group of Electronics Department at the University or Alcalá is working in a project called LOMUCO (spanish acronym for Collaborative Multi-robot Localization for Elderly Assistant) and "ROBOCITY 2030" (spanish acronym for Service Robots Upgrading Urban Standards of Living) [4].

This paper describes the general architecture of the system and presents an initial application of our system in people recognition and patient monitoring to carry out assistance tasks by mean of robots. Therefore, this paper is focussed in patient recognition and robot navigation in a collaborative way. For recognition, one of the most used methods is the principal component analysis (PCA) [5]. PCA tries to solve the recognition problem by reducing the dimensionality of the data (both training and sample data). Then, it keeps the most significant components that will be used on the decision phase. This method is widely used for face recognition. The objective of this method is to perform the dimensionality reduction while preserving as much as possible the separation of the different classes. On the other hand, mobile robot localization is the problem of estimating a robot's pose (location, orientation) relative to its environment [6]. The localization problem is a key problem in mobile robotics and plays a pivotal role. Probabilistic methods have been applied with remarkable success to single-robot localization [7,8,9,10], 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. 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.

II. ARCHITECTURE

The general objective of this work is to develop a robots´ fleet working together to make assistance tasks in a collaborative way (Fig.1). This way, we have placed

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 LOMUCO (CCG06-UAH/DPI-0721).

R. Barea is with the Electronics Department, University of Alcalá, Spain (e-mail: barea@depeca.uah.es; http://www.depeca.uah.es/personal/barea).

L.M. Bergasa is with the Electronics Department, University of Alcalá, Spain (e-mail: bergasa@depeca.uah.es).

E. López Guillén is with the Electronics Department, University of Alcalá, Spain (e-mail: elena@depeca.uah.es).

M. Ocaña is with the Electronics Department, University of Alcalá, Spain (e-mail: mocana@depeca.uah.es).

1-4244-0830-X/07/$20.00 ©2007 IEEE.

cameras fixed on a building that permit recognize users and its position, then, the best placed robot in the environment make the assistance task. In this case, it is very important to know where is located each robot. For this reason, we have implemented a collaborative multi-robot localization system based on particles filters (PF).

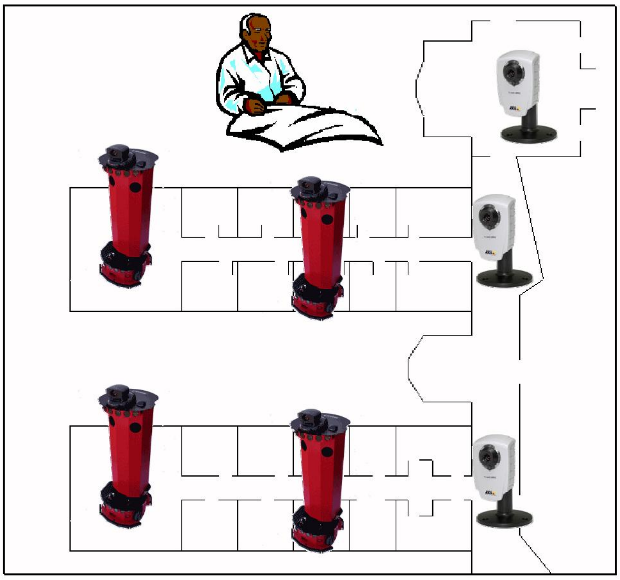


Fig. 1. Robot´s fleet working.

A. Robots

Four robotic platform have been developed (based on PeopleBot, pioneer DX 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].

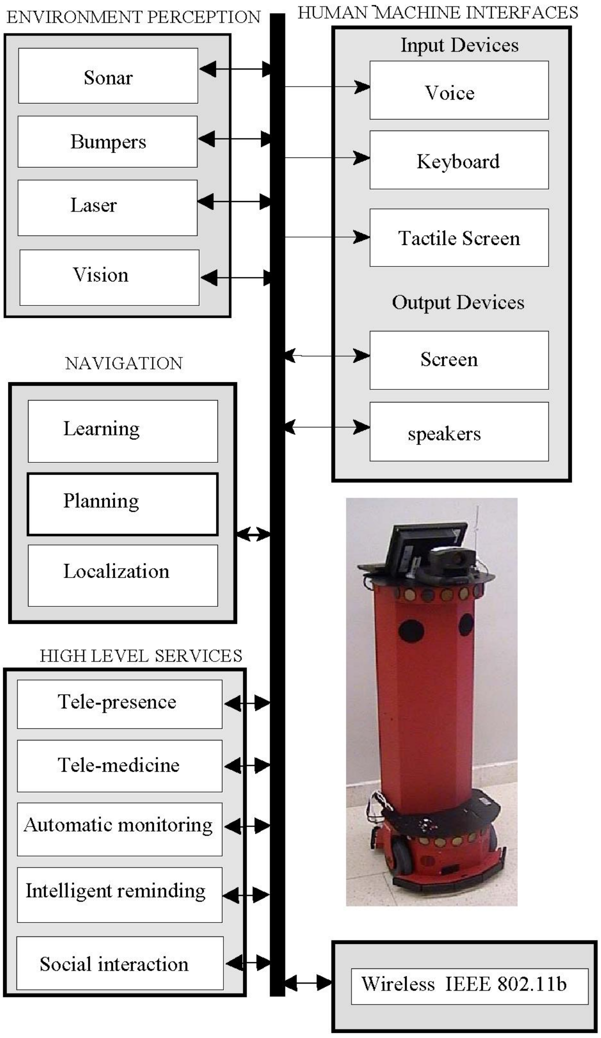


Fig. 2. General architecture of the robots.

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.

III. Multi-robot Monte Carlo localization

Monte Carlo Localization have been widely studied in

[6,14]. 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:

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

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:

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

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

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 $Bel(x)$.

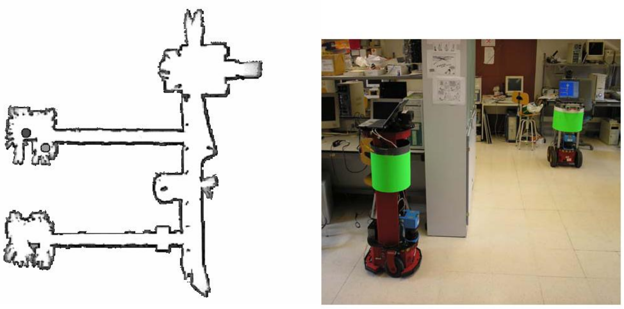
The initial set of samples represents the initial knowledge $Bel(x_o)$ about the state of the dynamical system. For instance, in global mobile robot localization, the initial belief is a set of poses drawn according to a uniform distribution over the robot's universe, annotated by the uniform importance 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:

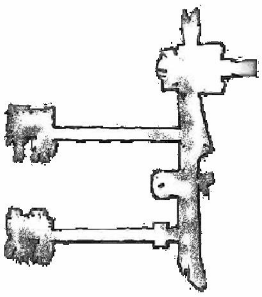
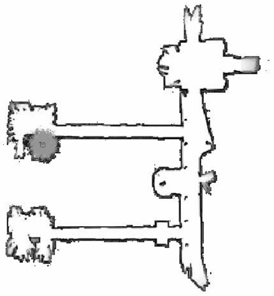
- Sample $x^i_{t-1} \sim Bel(x_{t-1})$. Each such particle x^i_{t-1} is distributed according to the belief distribution $Bel(x_{t-1})$.
- Sample $x^i_{t-1} \sim p(x_t/x^i_{t-1},a_{t-1})$. In this case, x^i_t is distributed according to the product distribution $p(x_t/x_{t-1},a_{t-1}) \cdot Bel(x_{t-1})$.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

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

The following example shows how collaborative multi-robot Monte Carlo localization improves single localization [15]. Robot 1 is initialized with uniform belief and Robot 2 with gaussian belief. Figs. 3a.b show initial robots´ position and Figs. 3c.d show the initial distributions of particles.



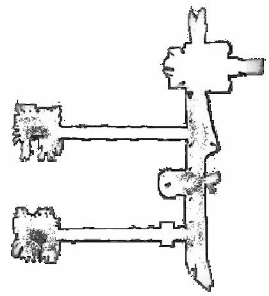
a) b)

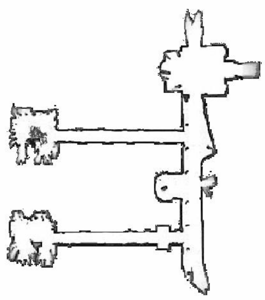
c) d)

Fig. 3. Initial distributions.

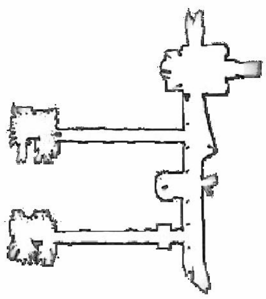
If Robot 1 wanders across top horizontal corridor (Fig. 4a), when Robot 2 detects Robot 1 (Fig. 4b), the detection model is sent and Robot 1 updates its belief distribution. This way Robot 1 is well-located before reaches the corridor (Fig. 4c).

a) b)

c) d)

e) f)

Fig .4. Collaborative multi-robot Monte Carlo localization.

IV. Face Recognition

The problem of user identification consists of the following: a fixed camera acquires images of people in the work environment; an automated system extracts faces from these images and quickly identifies them using a database of known individuals. The system must easily adapt as people are added or removed from its database, and the system must be able to recognize individuals in the room. The face localization system is based on a detector of simple features called *Haar-like* and a classifier based on the learning algorithm *AdaBoost* [16]. This section focuses on the face recognition technology that is required to address this real-world task. Face recognition has been actively studied [17], in this domain, techniques based on Principal Components Analysis (PCA) popularly termed eigenfaces, have demonstrated excellent performance. In our case, we use a simple, memory-based PCA algorithm for face recognition.

A. *Database and Preprocessing*

The system uses human face images from a database composed by 20 tightly-cropped images of different individuals with only minor variations in pose (±20º) and facial expression. The faces are consistently positioned in the image frame, and very little background is visible. Fig. 5 shows several faces of a person and Fig. 6 shows faces of some individuals.

For detecting faces, the system acquires an image and detects where exits some faces. To do it we use a PCA algorithm as can be see in Fig. 7.

After that, using the previous database compares the face obtained with faces stored in database using PCA. With this information the user is recognized. If the system does not recognize the user, ask for his name and introduced a set of faces of him in the database and the database is trained again. This way, the system stores all users and the database is increased on-line.

Fig. 5. Different images of the same user.

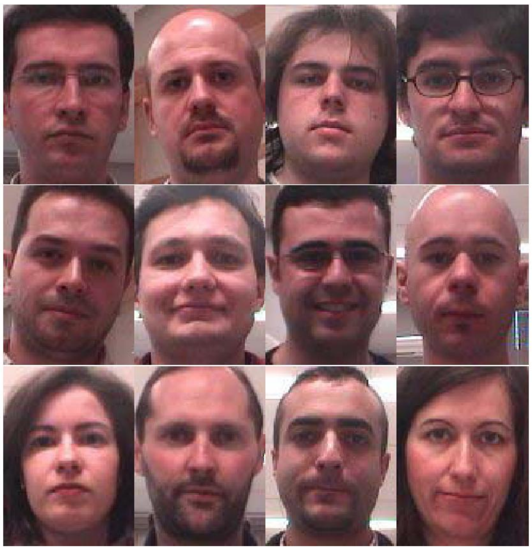


Fig. 6. Images from different users.

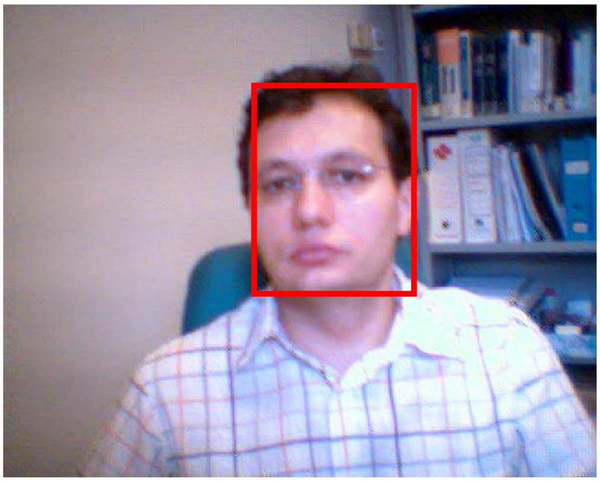


Fig. 7. Face detection.

B. Principal Components Analysis (PCA)

One of the most widely used baseline for face recognition is *eigenfaces* [18]. It employs Principal Components Analysis (PCA) which is based on the discrete Karhunen-Loeve (K-L), or Hotelling Transform [10], and is the optimal linear method for reducing redundancy, in the least mean squared reconstruction error sense. PCA has become popular for face recognition with the success of *eigenfaces*.

PCA algorithm projects points in R^d into R^m, (where $m \leq d$, and typically $m \ll d$). For face recognition, given a dataset of N training images (X), it is create a N d-dimensional vectors (X_1, X_2, … X_N). The principal components of this set of vectors is computed to obtain a d x m projection matrix, W.

Now, the image X may be compactly represented as *weights*, $\theta i = (\theta i1, \theta i2, \ldots \theta im)^T$, such that $Zi = \mu + W\theta i$ approximates the original image, where μ is the mean of the Xi and this reconstruction is perfect when m = d. The columns of W form an orthonormal basis for the space spanned by the training images.

Each training image is first projected into the eigenspace, and represented as a weight vector $\theta i = W^T(Xi-\mu)$. The centroid of the weight vectors for each person's images in the training set is computed and stored.

When a test image is presented to the system, it is first projected into the eigenspace and its weight vector θ_{new} is computed. θ_{new} is then compared against the stored weight vectors, Θ, and the θ_K that is closest θ_{new} is located. The label of θ_{best} is returned as the identity of the face represented by θ_{new}.

V. ASSISTANCE TASKS

Our objective is to design a fleet of robots to perform basic tasks such as transport medicines, taking messages or guiding visitors or patients to hospital beds or rooms.

The implemented system allows control a set of robots working together to carry out assistance tasks. A software application has been developed (Java) and allows to control the whole system from a remote computer or even from a mobile device (PDA). The system allows configure the position of the cameras and the number of robots. Robots can locate themselves in the environment and to navigate in a sure way. Fig. 8 shows a chart of this architecture and Fig.9 the configuration window.

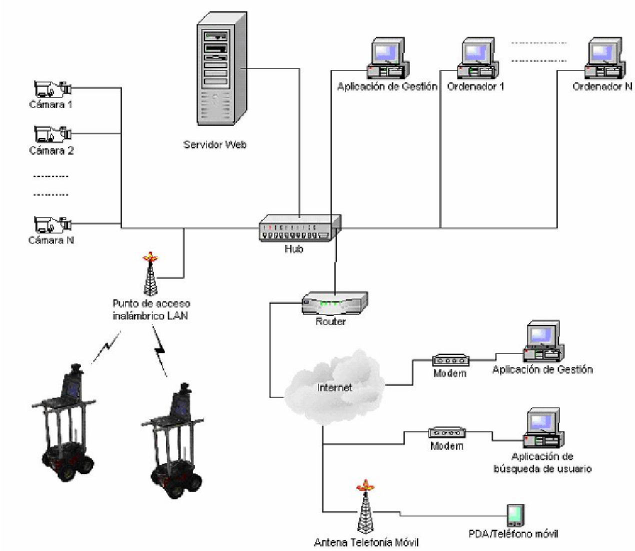


Fig. 8. Control scheme.

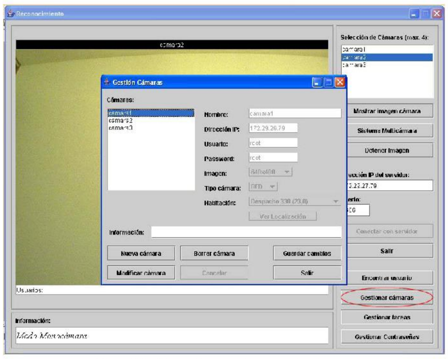
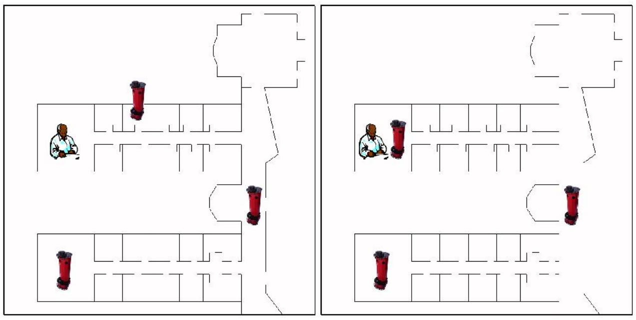


Fig. 9. Configuration window.

Once the system is configured, the tasks can be introduced, such as, taking messages to patients, e. g. "you have to take your medicine" or "you have to go sleep." Each task is assigned at a user and the hour and date of realization. The system continually is detecting patients or people in the images of the cameras and it takes a list from where they are. This way, when a task is executed it is looked for where the user is and the nearest robot is sending to carry out it.

For instance, if three robots are working in the environment (Fig. 10a) and the task activated is “going to the patient's room”, the system detects what robot is better located (Fig. 11) and sends the corresponding orders so that it will carry out the assigned task (Fig. 10b).



a) b)

Fig. 9. Assistance task in collaborative way.

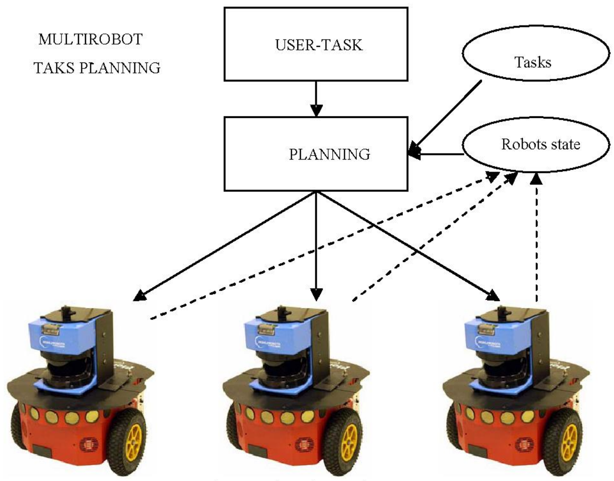


Fig. 9. Planning scheme.

VI. Conclusion

Teams of robots working together may be used to help on hospitals or geriatrics. These robots can carry out some easy or routine assistance task and free staff to spend more time with patients.

This paper reported the initial design and results of a robot´s fleet working together to make assistance tasks in a hospital, at home or in a geriatric in a collaborative way. Shortly, the multi-robot Monte Carlo localization algorithm applied to assistant robots has been studied and how the results obtained improves localization speed and accuracy when compared to conventional single-robot localization. This allows that the realization of collaborative tasks is more effective. On the other hand, an application for controlling and programming assistance tasks has been commented.

References

[1] Nursebot Project. http://www-2.cs.cmu.edu/~nursebot
[2] Morpha Project. http://www.morpha.de
[3] N. Roy, G. Baltus, D. Fox, F. Gemperle, J. Goetz, T. Hirsch, D. Margaritis, M. Montemerlo, J. Pineau, J. Schulte, and S. Thrun, “Towards Personal Service Robots for the Elderly,” Workshop on Interactive Robots and Entertainment (WIRE 2000), 2000.
[4] Robocity 2003 Project. “Service robots upgrading urban standards of living” http://www.robocity2030.org/
[5] M. Turk, A. Pentland, ”Eigenfaces for Recognition,” Journal of Cognitive Neuroscience. Vol 3, No. 1. 71-86, 1991.
[6] S. Thrun, D. Fox , W. Burgard and F. Dellaert, “Robust Monte Carlo Localization for Mobile Robots,” Artificial Intelligence Journal, Vol. 128 (1-2), 2001, pp. 99-141.
[7] I. J. Cox and J.J. Leonard, ”Modeling a dynamic environment using a bayesian multiple hypothesis approach,” Artificial Intelligence, Vol. 66, 1994, pp. 311–344.
[8] D. Fox, W. Burgard, and S. Thrun, “Active markov localization for mobile robots,” Robotics and Autonomous Systems, Vol. 25 (3-4), 1998, pp. 195–207.
[9] E. López, R. Barea, L.M. Bergasa y M. Escudero, “A human_robot cooperative learning system for easy installation of assistant robots in new working environments,” Journal of Intelligent and Robotic Systems, Vol. 40 (3), 2004, pp. 233-265.
[10] E. López, L.M. Bergasa, R. Barea and M. Escudero, “A navigation system for assistant robots using visually augmented POMDPs,” Autonomous Robots, Vol. 19, 2005, pp. 67-87.
[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, pp. 909-915.
[13] Carnegie Mellon Robot Navigation Toolkit. http://www.cs.cmu.edu/~carmen/index.html
[14] D. Fox, W. Burgard, and S. Thrun, “Active markov localization for mobile robots,” Robotics and Autonomous Systems, Vol. 25 (3-4), 1998, pp. 195–207.
[15] R. Barea, E. López, L.M. Bergasa, S. Álvarez, M. Ocaña, “Collaborative Multi-Robot Monte Carlo Localization in assistant robots,” Accepted for publication in International Transactions on Systems Science and Applications, 2007.
[16] P. and M. Jones, “Rapid Object Detection using a Boosted Cascade of Simple Features,” Proceedings IEEE Conf. on Computer Vision and Pattern Recognition. 2001.
[17] T. Sim, R. Sukthankar, M. Mullin and S. Baluja, “Memory-based Face Recognition for Visitor Identification,” Proceedings of International Conference on Automatic Face and Gesture Recognition, 2000.
[18] M. Turk and A. Pentland, “Eigenfaces for recognition,” Journal of Cognitive Neuroscience, 3(1), 1991.