A Human-Robot Interaction System for Navigation Supervision based on Augmented Reality

P. Núñez, J.P. Bandera, J.M. Pérez-Lorenzo, and F. Sandoval
Grupo de Ingeniería y Sistemas Integrados, Dpto. Tecnología Electrónica.
Universidad de Málaga, Málaga (Spain)
Email: pmnt@uma.es

***Abstract*— This paper proposes an innovative human-robot interaction mechanism that permits users to interact intuitively with an autonomous mobile robot which localisation problem is solved using a new and fast feature extraction method. To allow that human-robot interaction, we use an Augmented Reality display. This mechanism makes it possible to overlay planning, world model and sensory data provided by the robot over the same field of view. The determination of the camera pose in the AR system is solved using this novelty feature-based localisation method. Thus, the human user can intuitively help to build a topological map in an unknown environment by setting and manipulating map nodes and visualize and correct the robot's path planning.**

I. Introduction

Augmented Reality (AR) is the concept of overlaying computer-generated image elements over the user's field of view. An AR system supplements the real world with virtual, computer-generated objects that appear to coexist in the same space of the real world. There has been much research in the field of AR in last years, an excelent overview and update is given in [1]. AR techniques can be beneficial in programming robotic systems. Thus different applications of this technique appeared in the last years, as the approaches of Yanagihara et al [2] and, more recently, Pettersen et al [3]. In summary these applications define a visual feedback to order and correct tasks of industrial robots. In all these cases, the environment is known a priori and well modeled. Such an approach will fail however if the model is inaccurate or if the environment surrounding the robot is changing in an unpredictable fashion (unstructured environment) [4]. In these cases, it is impossible that the robot knows what to do in all situations. A human operator inserted in the loop is the classical solution to correctly control the robot in classical teleoperation.

More recent approaches elevate the human user to the role of a supervisor. In these cases, the operator controls the robot indirectly through some graphical abstraction. The main problems that appear in this teleoperation task are the reduced visibility and the delay in receiving the video signal [5]. These problems can be minimized by using a semi-autonomous navigation system in which most navigation decision are automatically made by the robot itself. In this case the user does not require a high frame rate nor detailed images to perform the supervision task. AR techniques can be used to provide an interaction channel between the human operator and the robot. Thus, the superposition of synthetic data over the real images can help to improve visual feedback or it may serve as a predictor of motion commands, before they are sent to the real robot [6]. In this work, we present an interaction mechanism between the human supervisor and the autonomous mobile robot for intuitive integration of navigation information based in Augmented Reality. The goal of this application is to allow on-line generation of a topological map by setting and manipulating map nodes, allowing supervision of the paths that the robot proposes to follow. The application notifies the robot if there are problems in the planned path (e.g. stairs or some obstacle). The user is also able to label the topological nodes to allow a 'high level' navigation between two nodes of this topological map. Finally, this paper proposes a new and fast feature extraction method to permit the navigation and localisation tasks of the autonomous robot, so that augmented scene can be represented correctly. The paper is organised as follows: Section II is a brief description of the whole navigation system. Section III details the AR-based control of the mobile platform. Section IV shows some experimental results and, finally, conclusions and future work are presented in Section V.

II. Robot Navigation System

In the proposed system, the robot can navigate without teleoperation, and the human only remains in the control loop to supervise its operation. Therefore, the robot must autonomously solve all navigation problems. Global mobile robot localization is the problem of determining the pose of the robot in an environment when the starting position is unknown. It is crucial to solve the localisation problem to align properly the computer-generated objects with the real scene. The odometric system gives information about the relative motion of the robot. This information can be integrated over time to give an estimation of the robot pose which is valid when moving over short distances. Over large distances, however, the errors in the odometric information will accumulate and result in unbounded pose estimation and then, external sensors must be utilized to bound the pose estimation error. In last years there are different approaches based on a probabilistic point of view [7][8]. A family of these ones, known as Monte Carlo Localization (MCL) is currently among the most popular methods to solve this problem. MCL algorithms represent a robot's belief by a set of weighted

1-4244-0088-0/06/$20.00

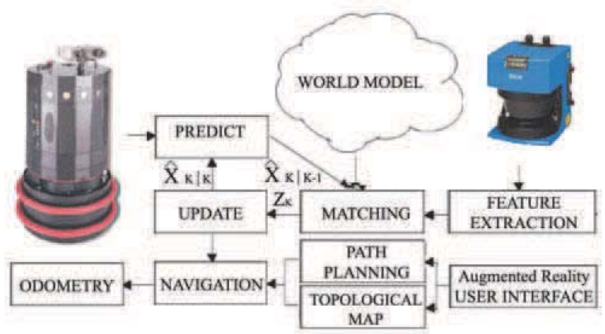

Fig. 1. Illustration of the proposed algorithm

samples, which approximate the posterior probability of where the robot is located by using a Bayesian formulation of the localization problem.

MCL methods can be computationally very expensive. Thus, some authors propose to apply MCL to a constrained set of extracted features [8]. The proposed system applies a particle filter to solve localisation problem. This is a good known MCL method which key is to represent the posterior belief by a set of N weighted and random samples or particles. A novel feature extraction method is used to speed up the data association [9]. In most implementations of feature based localization the system can be divided into three parts. First, features are extracted from sensor data to create higher level information from the raw data. Then, using an action model based in the robot's odometry, a prediction can be made about the localisation of the robot. Finally, extracted features are matched with the world model and used to update the estimated pose. The data flow diagram of the proposed navigation system is given in Fig. 1. To minimize its complexity, it has been split into different modules operating in a concurrent way. Each of these modules performs a specific function and exchanges information to implement the complete system. The proposed feature extraction method permits to build a local parameterized line and curves segment map that the system uses for its localisation. This section details the navigation and localisation tasks of the proposed navigation system.

A. Feature extraction method

One of the main novelties of the proposed system is the feature extraction method. This one is able to obtain corners, line and curve segments of the robot's environment. To do that the system uses a curvature based method. These algorithms have been used in pattern recognition and they permit to obtain different features from raw data. The proposed method for adaptive curvature estimation in laser scan data is a modified version of [10] and, for each range reading i of a laser scan, it consists of the following steps:

1) Calculation of the maximum length of laser scan presenting no discontinuities on the right and left sides of the working range reading i: $K_f[i]$ and $K_b[i]$, respectively. $K_f[i]$ is calculated by comparing the Euclidean distance from range reading i to its $K_f[i]$-th neighbour ($d(i, i + K_f[i])$) to the real length of the laser scan between both range readings ($l(i, i + K_f[i])$). Both distances tend to be equal in absence of corners, even if laser scans are noisy. Otherwise, the Euclidean distance is quite shorter than the real length. Thus, $K_f[i]$ is the largest value that satisfies

$$d(i, i + K_f[i]) > l(i, i + K_f[i]) - U_k \quad (1)$$

being U_k a constant value that depends on the noise level tolerated by the detector. $K_b[i]$ is also set according to Eq. (1), but using $i - K_b[i]$ instead of $i + K_f[i]$. The selection of the U_k value is very important for correct detection of corners and in our case it has been set to U_k=1.0.
2) Calculation of the local vectors $\vec{f_i}$ and $\vec{b_i}$ associated to each range reading i. These vectors present the variation in the x and y axis between range readings i and $i + K_f[i]$, and between i and $i - K_b[i]$. These vectors $\vec{f_i}$ and $\vec{b_i}$ are defined in [10].
3) Calculation of the angle associated to each range reading of the laser scan. The angle at range reading i can be estimated by using the equation:

$$\theta_i = arccos\left(\frac{\vec{f_i} \cdot \vec{b_i}}{|\vec{f_i}| \cdot |\vec{b_i}|}\right) \quad (2)$$

The absolute value of the obtained curvature index, $|\theta_i|$, represents the curvature associated to each range reading in an absolute manner.
4) Detection of line segments over $|\theta_i|$. Line segments result from the scan of planar surfaces. Therefore, they are those sets of consecutive range readings which: i) are under a minimum angle (in our experiments, this minimum curvature height, θ_{min}, has been fixed at 0.05); and ii) have a size greather than a minimum length value (l_{min}=10 range readings).
5) Detection of curve segments over $|\theta_i|$. Curve segments result from the scan of curve surfaces. Contrary to the curvature values associated to a line segment, it can be appreciated that the curvature function associated to a curve segment presents a consecutive set of local peaks, some of them could be wrongly considered as corners. To avoid this error, the algorithm associates a cornerity index to each set of consecutive range readings whose θ_i values are over θ_{min} or under $-\theta_{min}$ and have a size greather than l_{min}. This cornerity index, ci, is defined as

$$ci = \frac{\frac{1}{i_e - i_b}\sum_{j=i_b}^{i_e}\theta_j}{max_{i\in(i_b,i_e)}\{\theta_i\}} \quad (3)$$

where i_b and i_e are the range readings that bound the possible curve segment. If ci is close to one, the mean curvature of the segment and its maximum value are similar, and the segment can be considered as a curve segment. If ci is low, the segment cannot be considered as a curve segment. Therefore, curve segments are those

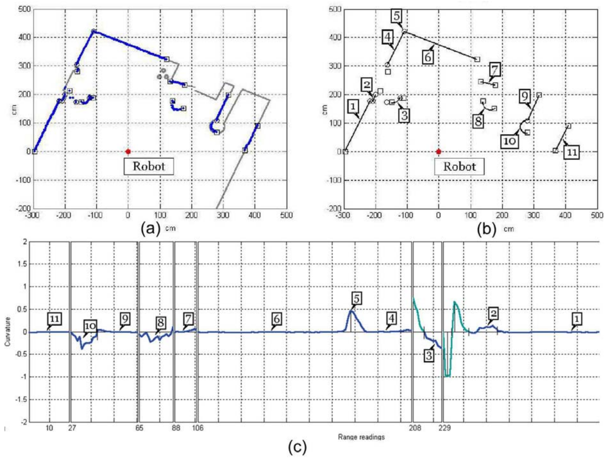

Fig. 2. a) Segmentation of the laser scan (*square*- breakpoints, *circle*-detected corners); b) line segments, corners and curve segments associated to a); and c) curvature functions associated to a).

sets of consecutive range readings which do not define a line segment and have a cornerity index greater than a given threshold U_c (U_c has been fixed at 0.5 in all experiments).

6) Detection of corners over $|\theta_i|$. Corners in this method are always defined by a value associated to a local peak of the curvature function, and a region bounded by two range readings, i_b and i_e. Therefore, it can be characterised by a cornerity index, ci. Taken this into account, corners are those range readings which do not belong to any line or curve segments and satisfy the following conditions: i) they are local peaks of the curvature function; ii) their $|\theta_i|$ values are over the minimum angle required to be considered a corner instead of a spurious peak due to remaining noise (θ_{min}); iii) they are located between two segments which have been marked as line or curve segments, these two segments determine the region of the corner, (i_b, i_e); and iv) their cornerity indexes are lower than U_c.

The advantage of this new method for feature extraction in laser data can be appreciated in Fig. 2. Fig. 2a presents laser scan range readings and the detected line segments, corners and curve segments over the real layout. This information of the laser scan is given in Fig. 2b. Fig. 2c presents the curvature functions associated to the laser scan in Fig. 2a.

B. Path planning

Path planning is the guidance of an agent from an origin to a destination, while avoiding all encountered obstacles. Real situations require the planning and the execution phases to function in parallel. To grant task completion, we employ a hierarchical map structure that permits collaboration between a global and a local path planners. This map structure is described in detail in [11]. The two planners work at different levels of the map structure. The global planner functions at topological level and computes a minimum-cost path using a A* algorithm. This path defines a region R_p at the base level of the map structure which is the global map. The local planner applies a potential field approach over the region R_p, defined by the global planner to modify the path. A potential field is formulated in the free space such that obstacles appear to exert negative potentials and goals positive potentials. In our case, the potential function is the sum of a collection of potential functions emanating from the obstacles and goal models.

III. Supervising the Mobile Platform

The aim of the proposed system is to integrate the navigation system with an interaction mechanism that permits the human user to supervise the robot operation. In order to achieve this aim, an AR system is used to permit the user to visualize the action of the global and local planners.

One of the main problems of any AR system is to align properly real and virtual images. Some commercial tracking systems rely on magnetic fields, camera based approachs or artificial landmarks. The proposed system solves this problem by using the navigation and localisation tasks previously described.

Thus, from the global segment-based map initially provided by the user, a virtual representation of the environment is built by the system, where each 2D segment becomes a three-dimensional wall. A virtual camera looks on the virtual scene from the robot's point of view, generating an internal schematic image of obstacles in the robot's path. This virtual camera is aligned with the real one using the known position, orientation and dimensions of the robot. Real camera attributes are also considered to model the virtual camera so that real and virtual images can be overlapped properly. Our goal is to augment the video images sent by the robot with two-dimensional topological and path information. To do this, the system needs to check visibility of marks, paths and nodes which should be ocludded in the augmented video image if they are behind walls or other items. Instead of analysing the video images, which would be computationally very expensive, the system simply checks the visibility of the item in the virtual camera view.

A. Modification of the Topological Map

We intend on this point to give the operator high-level control over the navigation of the robot. The basic idea is to permit the user to control in every moment a topological map on which to set and manipulate intuitively different nodes. The autonomous mobile robot is able to build its own topological map and use it to move along the environment. Moreover, the robot's localisation is solved correctly. But there are some situations that can precise this human intervention (like to set new nodes to facilitate the 'high level' navigation or to show some obstacles in the environment). Then the user, by means of the easy control of its interface and through the mouse, can decide the placement of a determined node on the map, and select subsequently a name for it. This information in two dimensions is projected finally on the image, appearing like a red cylinder with an unfolded label indicating the name assigned by the user. Connections between topological nodes are augmented too. Through commands, the user is able to modify or eliminate these nodes, and also to indicate the agent a displacement simply selecting a node name.

B. Visualization and Supervision of Planned Paths

The visualization and manipulation of the topological map allows the user to monitor if the path determined by the robot can be carried out without risk, or simply whether it is the expected path to follow. The user decides the destination point of the movement, from the press of the left button of the mouse. Subsequently the information of the path is represented in the screen as a blue line on the floor. This line is ocludded by walls using the previously described mechanism. In each moment, the user knows the path and is able to cancel the movement.

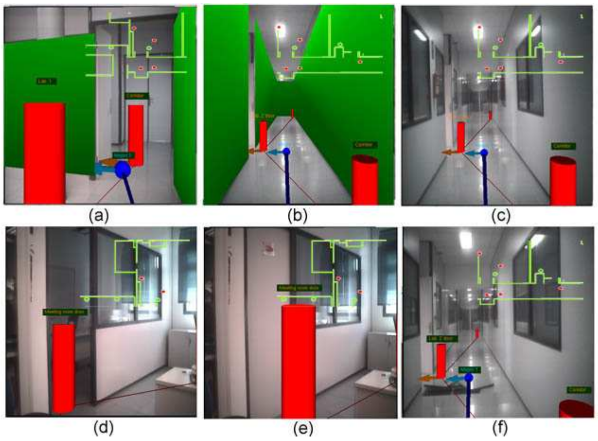

Fig. 3. a -f) different AR scenes in an indoor environment

IV. Experimental results

For the experiments reported here, we have used a Nomad200 mobile robot and a SICK Laser Measurement System (LMS) 200. This robotic platform includes a synchro-drive system where its three wheels move synchronously in two axes, one axis for translation and one for steering. On the other hand the LMS is configured to planar range scan with scanning angle of 180^o. The experiments focused on running the Nomad200 in an indoor environment for about two hours. The system locates the robot properly in standard situations and minor errors introduced in particular cases (e.g. laser reflections) are corrected when more complete data is available. The system also allows the supervision of the robot's navigation generating AR images. The introduction and manipulation of the nodes in the topological map, as well as the visual representation of the planned path, facilitate to the user the supervision of the navigation on the map. The average time to obtain a local segment-based map with our feature extraction method is less than 12 ms in a 400MHz PC, while video images are augmented in real time. Some advantages of the proposed method for teleoperation with the robot can be appreciated in Fig. 3. Figs. 3a-f present different captures of the AR scene. In Figs. 3a-c typic images can be appreciated with superimposition of walls, corners, path planned, map and labeled topologycal map node. Figs. 3d-e are a whole example where the user modifies the position of the topologycal map to correct the real position of the node. Finally in Fig. 3f a cancelation of the path planned is presented to avoid the clear situation of risk.

V. Conclusions

This paper presents an AR system that allows the interaction between an autonomous mobile robot and a human supervisor. The user can create a topological representation of the environment by setting and manipulating map nodes. Besides, the robot's path planning can be supervised. This application demonstrates that AR can be used for robot instruction. In contrast to the work by Giesler et al [6] in this work we propose a mechanism that combine the autonomous robot navigation and localisation and the virtual information set by the user in the environment. This allows to supervise the right navigation by indoor environmet. Future works will be focused on investigate the benefits of using AR to visualize what is the robot intention of movement and to manipulate its representation of the environment.

Acknowledgment

This work has been partially granted by the Spanish Ministerio de Educación y Ciencia (MEC) and FEDER funds Project no. TIN2005-01349.

References

[1] R.T. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier "Recent advances in Augmented Reality", IEEE Computer Graphics and Applications, vol. 21. no. 6. pp. 34-47. November/December 2001.

[2] Y. Yanagihara, T. Kakizaki, K. Arakawa and A. Umeno, "Task world reality for human and robot system - A multimodal teaching advisor and its implementation", Proc. 1996 IEEE Int. Workshop on Robot Human Communication, pp. 38-43, 1996.

[3] T. Pettersen, J. Pretlove, C. Skourup, T. Engedal and T. Lokstad, "Augmented reality for programming industrial robots", Proc. 2nd IEEE and ACM Int. Symp. on Mixed and Augmented Reality, pp. 319-320, 2003.

[4] G. Burdea, "Invited review: the sinergy between virtual reality and robotics", IEEE Trans. on Robotics and Automation, vol. 15, no. 3, pp. 400-410, 1999.

[5] A. Casals, J. Fernandez and J. Amat, "Augmented reality to assist teleoperation working with reduced visual conditions", Proc. 2002 IEEE Int. Conf. Robotics and Automation, pp. 235-240, 2002.

[6] B. Giesler, T. Salb, P. Steinhaus and R. Dillmann, "Using augmented reality to interact with anautonomous mobile platform", Proc. 2004 IEEE Int. Conf. Robotics and Automation, pp. 1009-1014, 2004.

[7] W. Burgard, D. Fox, D. Hennig, and T. Schmidt, "Estimating the absolute position of a mobile robot using position probability grids". Proc. of the 14th National Conf. on Artificial Intelligence, pp. 896-901, 1996.

[8] D. Fox, W. Burgard, F. Dellaert, and S. Thrun. "Monte Carlo localization: efficient position estimation for mobile robots". Proc. of the 16th AAAI National Conf. on Artificial Intelligence, pp. 343-349. 1999.

[9] P. Núñez, R. Vázquez-Martín, J.C. del Toro, A. Bandera and F. Sandoval, "Feature extraction from laser scan data based on curvature estimation for mobile robotics", IEEE Int. Conf. on Robotics and Automation, 2006.

[10] P. Reche, C. Urdiales, A. Bandera, C. Trazegnies and F. Sandoval, "Corner detection by means of contour local vectors", Electronics Letters 38 (14), pp. 699-701, 2002.

[11] A. Bandera, C. Urdiales and F. Sandoval, "A hierarchical approach to grid-based and topological maps integration for autonomous indoor navigation", Proc. 2001 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 883-888,2001.