Visuo-Haptic Interface for Teleoperation of Mobile Robot Exploration Tasks

Nikos C. Mitsou, Spyros V. Velanas and Costas S. Tzafestas

Abstract— **With the spread of low-cost haptic devices, haptic interfaces appear in many areas in the field of robotics. Recently, haptic devices have been used in the field of mobile robot teleoperation, where mobile robots operate in unknown and dangerous environments performing particular tasks. Haptic feedback is shown to improve operator perception of the environment without, however, improving exploration time. In this paper, we present a haptic interface that is used to teleoperate a mobile robot in exploring polygonal environments. The proposed visuo-haptic interface is found to improve navigation time and operator perception of the remote environment. The human-operator can simultaneously select two different commands, the first one being set as "active" motion command, while the second one is set as a "guarded" motion type of navigation command. The user can feel a haptic equivalent for both types of teleguidance motion commands, and can also observe in real-time the sequential creation of the remote environment map. Comparative evaluation experiments show that the proposed system makes the task of remote navigation of unknown environments easier.**

I. INTRODUCTION

TELEOPERATED mobile robots are a major tool in the exploration of unknown and risky environments. Mines removal [1] and exploration of underwater structures [2] are two common applications carried out through mobile robots. Robot motion is usually controlled by system operators with the help of a camera mounted on robot or inspecting the area from above. However, although vision systems provide much information of the environment, they require network bandwidth and much attention from the operator.

To overcome this problem, haptic devices have been recently introduced in the field of telerobotics as a way of enhancing operators perception of the robot environment. They provide operators with the additional sense of "feeling" the robot workspace, thus making it easier to avoid obstacles and reducing the average number of collisions.

However, the force rendering process yields a problem regarding how the haptic feedback affects the exploration time. Ideally, we would like the presence of force feedback to reduce the exploration time or at least not to increase it. In practice, though, this additional sense often adds more information for operators to interpret and leads to an increase in the navigation time (more details on this in Section II).

Another important issue in mobile robot teleoperation is the selection of a proper driving mechanism. Usually, operators have to manually drive the mobile robot through obstacles by explicitly specifying the robot angular and linear velocity. By doing so, they are fully in charge of the robot motion and as a clear viewpoint of the robot environment may sometimes not be available, they could accidentally drive the robot to collisions or choose longer paths than optimal ones. In this paper, we try to deal with such issues for the case of remote exploration of a structured polygonal environment by a miniature mobile robot, with the use of a haptic device. During robot exploration, robot sensor measurements are used to build an occupancy grid map of the environment which is displayed to the operator as a substitute for camera information.

The proposed system deals with the two issues mentioned earlier in the following way. The operator can simultaneously exert two different types of commands: an "active" and a "guarded" motion command. Each command receives force feedback independently (without influencing one another) making force origin clear. A behavior-based system is then responsible for controlling the overall motion performed by the slave mobile robot. The commands received from the haptic device act as a general policy that the robot must follow. Several issues are considered: the driving mechanism, force feedback generation, collision avoidance and driving with no visual information. Two types of experiments were conducted. Results show that exploration of unknown environments

Manuscript received March 15, 2006.

Authors are with the National Technical Univ. of Athens, School of Electrical & Computer Eng. Division of Signals, Control and Robotics, Zographou Campus, 15773, Athens, Greece. E-mails: nmitsou@mail.ntua.gr, svelanas@mail.ntua.gr, ktzaf@softlab.ntua.gr

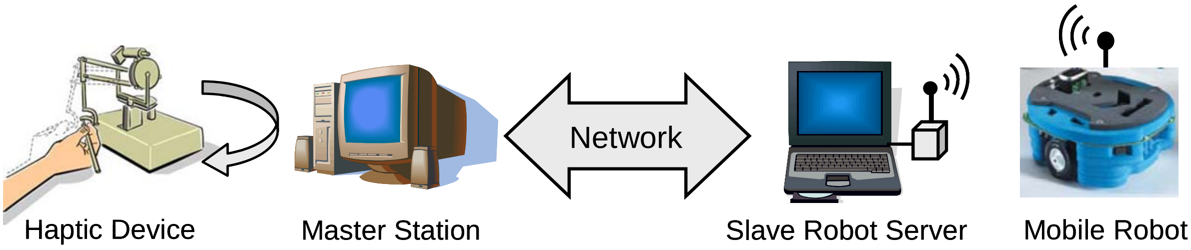

Fig 1. Overview of a mobile robot "haptic teleoperation" system with master (left) and slave (right) side.

can be completed in less time when force feedback is enabled. Moreover, with no visual information available to the operator, exploration tasks consisting of finding the exit in an unknown maze can be completed successfully when force feedback is enabled ('drive by feel' experiment).

The paper is organized as follows: In Section II, we present a brief survey of related work in the area of mobile robot teleoperation systems with haptic interfaces. In Section 3, we present an overview of the proposed teleoperation system, and in Section 4 we describe in detail its operation. Section V describes how the experimental analysis was performed, and Section 6 presents the experimental results obtained. Finally, conclusive remarks and future work directions are given in Section 7.

II. RELATED WORK

Recently, many researchers have combined haptic devices with teleoperated mobile robotic systems, in order to provide the operators with more information about the environment. In most cases, force feedback corresponds to the virtual repulsive forces exerted to the mobile robot originated from the obstacles that exist in its environment.

In [3] and [4], repulsive forces are computed from obstacles around the mobile robot detected by infrared sensors. System operators are explicitly controlling robot motion as they directly specify its angular and linear velocity. In both cases, cameras provide the operators with additional information about the robot environment (e.g. a panoramic webcam in [3] and a camera mounted on robot in [4]). However, as operators are explicitly specifying robot motion, robot is not completely prevented from driving into obstacles. In [4], a “drive-by-feel” experiment is also conducted. Operators were asked to drive the robot through a maze with the camera disconnected. Although they could perform limited tasks, they couldn’t successfully complete their mission.

In [5], authors combine two kinds of forces (environmental and collision-preventing) and show through a number of experiments that haptic feedback minimizes the number of collisions while increasing minimum distance between the robot and the obstacles with a small increase though in navigation time. In [6], however, authors experimentally prove that the number of collisions can also be decreased if the mobile robot is provided with a simple obstacle avoidance local behavior. In their experiments, force feedback has no measurable benefit. In both works, mobile robots are equipped with a camera that provides operators with a limited view of the environment.

In [7], a passivity based control schema for mobile robot teleoperation through a haptic device is described. No camera is used during robot navigation. Operators are blindly driving the mobile robot, avoiding obstacles with the help of force feedback. An occupancy-grid map is constructed and used to compute the virtual force that is exerted to operators hand. Although collisions are avoided, this method cannot be generalized in cases of exploration of unknown environments as the mobile robot is precisely following the operators' commands.

Summarizing, we can state that, as depicted in related work, haptic feedback in mobile robot teleoperation seems to increase operator perception of the remote environment, particularly when combined with an onboard robot behavior, but it usually causes a (small) undesirable increase in navigation time (as in [3],[6]).

III. TELEOPERATION SYSTEM OVERVIEW

An overview of the mobile robot haptic teleoperation system is shown in Fig 1. It consists of two sides: the master side, which contains the haptic device and the master station with the map-building module and the slave side, which contains the mobile robot and a slave robot server with the behavior and the localization module. A more detailed view of the main system modules and their interconnection is schematically shown in Fig 2. The functionality of these modules is described in the sequel.

A. *Master Side*

1) *Haptic Interface Module: Driving Scheme*

To develop a teleoperation interface that will facilitate intuitive teleguidance of a mobile robot exploring unknown environments (such as a rectangular maze), the first step was to implement a specific "driving scheme". According to this scheme, the haptic workspace is divided into three types of areas, as shown in Fig 3. These areas are not

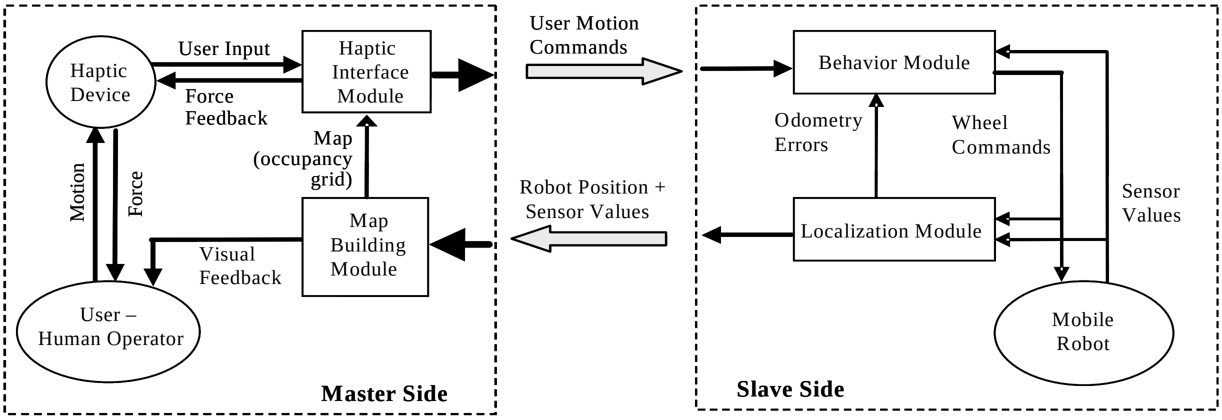

Fig 2. Mobile Robot Teleoperation System Architecture

relative to the local coordinate system of the mobile robot, but rather absolute and relative to the local coordinate system of the master workstation monitor.

The neutral area corresponds to the area in the center of the haptic workspace and implies a stop command. When selected, the robot immediately stops its motion. When the haptic end-point enters one of the Up, Down, Left or Right areas, the robot changes its orientation and moves according to the operator's command. For instance, if the Up area is selected, the robot will start moving upwards, as watched in the operator monitor. Finally, when the haptic end-point enters one of the bidirectional areas, a “combined command” will be issued and sent to the slave side. The robot will be instructed to move towards one direction and simultaneously “wall-follow” a wall.

2) *Master Station: Map-Building Module*

The *Map-Building Module* is responsible for the creation of an "occupancy grid map" representation of the environment under exploration. It receives robot position and sensors values from the *Localization Module* of the slave robot server described in a subsequent paragraph, and updates all affected cells. The map is constructed on-line and is displayed on the *Master Computer*. Cells that hold value greater than a threshold are considered occupied (painted white), cells with value less than this threshold are considered empty (painted black) and cells with no value are considered unvisited (uncertain - painted gray).

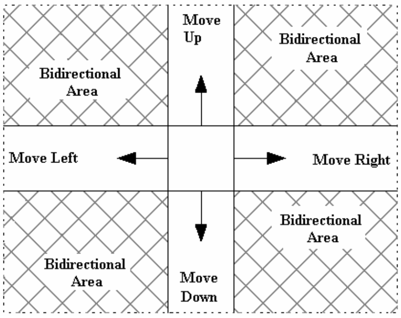

Fig 3. Remote Driving Scheme

The constructed map is thus sequentially created and continuously updated on the master control monitor along with the robot position and orientation (Fig 4). It forms the visual part of the information feedback provided by the interface to assist the human operator in monitoring the task execution and issuing proper navigation commands.

The second principal information feedback channel of the teleoperation interface concerns haptics and is described in the following paragraph.

3) *Haptic Interface Module: Force Generation*

As already mentioned, haptic feedback is added to the system to enhance operator's perception of the remote environment, and assist teleoperation of the robot exploration task. The force fed back to the operator is generated based on a 2D virtual joystick model, as schematically illustrated in Fig 5.

When the haptic control point exits the neutral area, a spring force is exerted to the operator attempting to restore end-point position inside the neutral area. The stiffness coefficient (K) of the virtual spring depends on the absence or presence of an obstacle in this direction, that is, on whether the respective motion command is permissible or not. In case that no obstacle hinders the execution of a specific motion command (move forward/backward, turn left/ right), the respective spring coefficient is set to a

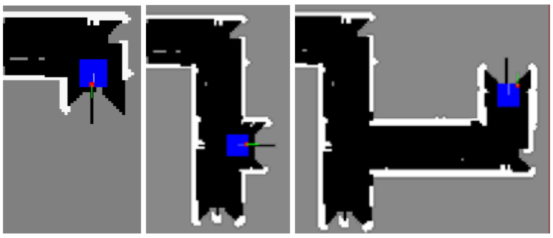

Fig 4. Sequential Map Creation in the Master Teleoperation Interface

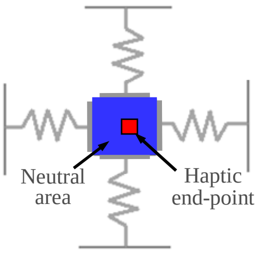

Fig 5. Virtual Joystick Model for Haptic Feedback

minimum value (Kmin). This feature enables the operator to feel if he/she is actually exerting an 'active' command.

In case that an obstacle (e.g. a wall) is detected in the direction of a motion command, the respective spring coefficient is gradually switched to a much larger (high Kmax) value. The presence of a wall can thus be viewed as the origin of a virtual repulsive spring force that is applied to the mobile robot. In this case, the operator is feeling a force on his/her hand as if the wall is pushing the robot away, thus conveying the important information that the respective motion command is not permissible.

It is important to note that, in order to calculate the environmental force that is applied to the human operator via the haptic device, we do not directly use the current sensor values transmitted by the Localization Module. The force is generated according to the values of the cells of the occupancy grid map. If more than n (a threshold we use to deal with noisy sensor values) occupied cells (e.g. forming a linear wall) are found in one direction, then the respective motion command is considered as not permissible. The virtual interaction force is then continuously computed according to the virtual joystick model described above, and sent to the Haptic Device for application. Apart from a spring model, a virtual damper model is also used to smooth the force and avoid large fluctuations in its value.

Additionally, a local robot model is implemented in the master station, to deal with delays in the communication between the master and the slave side. The robot position received from the Localization Module is used to update the local model estimation and the Haptic Interface Module considers the new updated position as the real robot position. In case of a great delay in the communication and no local model is used, the operator would not feel a wall coming closer to the robot. However, as the local model estimates the current robot position, the generated force is going to increase and the operator will feel the real distance from the wall.

B. Slave Side

1) Mobile Robot

In the slave side, the Mobile Robot is responsible for the exploration of the unknown environment. The robot used in our experiments is a Hemisson mobile robot [8]. It has a differential drive system, is equipped with six low-range infra-red sensors and has no wheel encoders. It sends sensors measurements to a server computer (slave robot server) and receives from it the new speed commands for each wheel. The communication between robot and server computer is performed via a wireless link.

2) Slave Robot Server: Behavior Module.

The Behavior Module in the master computer is responsible for the robot motion. Given a policy from the master side, the Behavior Module makes sure that the robot will follow it except for cases of possible collisions. Two different behaviors are implemented: a collision detection behavior and a wall-follow behavior. The first is activated when an obstacle is detected from the three front sensors. In this case, the robot stops when operators' command drives it towards the obstacle but accepts the commands that guide it away from the obstacle. The wall-follow behavior is activated by the operator policy. Under specific policies, the robot can follow walls and automatically turn on corners so as not to lose contact with them.

3) Slave Robot Server: Localization Module

The Localization Module keeps track of speed commands sent to robot and calculates robot position (x, y, orientation) using a Maximum Likelihood Localization algorithm [9]. This position is then transmitted to the master side along with the sensors values.

The lack of wheel encoders in the mobile robot makes the task of localization more difficult as no feedback is received from the wheels to indicate the amount of distance traversed. The orthogonal structure of the environment used in our case eases the localization task as a wall on either side of the robot acts as a global localization landmark.

IV. System Operation

In this section we will describe in more details how the proposed system works and demonstrate how the visual and the haptic feedback are combined.

As already mentioned, visual feedback contains the map created by the Map-Building Module and the robot position and orientation. To enhance operator perception, information about the haptic control point area and the commands issued by the operator are also visually represented on the Master Station monitor. A small square inside the robot drawing indicates the haptic control point position. The body of the robot corresponds to the neutral area of the haptic workspace. When the control point is inside the neutral area, the square is drawn inside the mobile robot; otherwise it is drawn in one of the robot sides, in case of a "simple command", or in one of the robot corners, in case of a "combined command". The

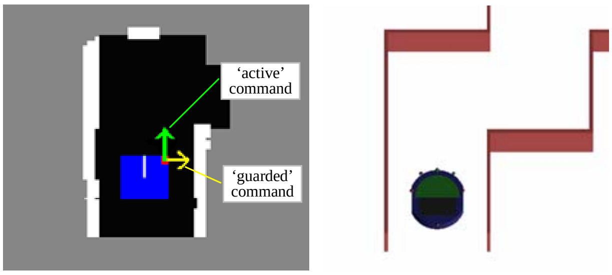

Fig 6. Example of combined command execution (in the left, the map on the master station, in the right the simulator view)

command issued by the human operator is also drawn as an arrow starting from the control point square towards the desired direction. In case of a combined command, an additional arrow is drawn towards a wall to indicate that the mobile robot is following that wall. In this way, the operator continuously knows the position of the haptic control point, as well as the type of command currently executed.

In the next subsections we examine the cases of simple (e.g. left, up) and combined commands (e.g. go left and follow right wall) execution.

A. Simple command execution

In the case of a simple command, this command is regarded as an "active command" and is directly sent to the Behavior Module which decides the next motion sequence with respect to the active command. At any time, two different cases exist:

- The robot can follow the active command: The command is translated to wheel speeds which are then sent to the robot motors. Odometry errors detected by the localization module will be sent to the behavior module which will correct robot orientation accordingly. The operator, however, will not be aware of these actions.
- The robot cannot follow the active command: Whenever an obstacle blocks the robot path, the collision detection behavior is activated and takes control of the robot motion. The robot stays still and any command that drives it towards the obstacles is ignored. At this time, the operator is watching the robot staying still on the master computer monitor and is feeling the obstacle in front of the robot through the haptic feedback. The collision detection behavior is deactivated when an active command that drives the robot away from obstacles is selected.

B. Combined command execution

In the case of a combined command, two different commands are generated, an active and a guarded command. The active command corresponds to the current robot orientation and the guarded command to the desirable orientation. The Behavior Module follows the active command as long as the guarded command cannot be executed. The wall-follow behavior is also enabled in order to force the robot to follow the wall on the guarded side.

Imagine the example of Fig 6, where current command is a simple command “Up” and the robot is heading up. In case that a combined command “Up/Right” is selected, the active command will be “Up” and the guarded command will be “Right”. The existence of a wall on the right side of the robot blocks the guarded command from becoming active. The command will be activated only when the robot reaches the corner and at this moment the robot will automatically turn right. The previous active command is then ignored and the guarded command is considered as the new active command.

When the operator exerts a combined command, he can feel not only the active command but the guarded command also. If the guarded command is blocked, a force is generated from the environment that pushes the operator hand away from the bidirectional area (where the virtual obstacle lies), towards the active command area. In this way, the operator can exert two different commands to the robot and feel the obstacles on each command direction independently.

To summarize, the following cases exist:

- The robot can follow the guarded command. The guarded command is converted to active and then executed.
- The robot cannot follow the guarded command. The command remains guarded (until it is permissible) and the active command is followed. A "wall-follow behavior" is enabled to make robot follow the wall on the guarded side and a force originated from this wall is generated.

Neither command can be executed. In this case, the collision detection behavior is enabled. The operator feels two forces, one from the active command towards the neutral area indicating a wall in front of the robot and one towards the active command area indicating a wall on the guarded side of the environment.

The existence of the wall-following behavior is important in cases of great odometry errors. If no such behavior is activated, the robot would probably lose contact to the wall, making the task of exploration much more difficult.

V. EXPERIMENTAL ANALYSIS

To test the effectiveness of our approach, we performed two different kinds of experiments. The first one was a comparison between our approach and a common teleoperation method, while the second one constituted a “drive-by-feel” experiment.

Thirty-six participants, from 18 to 35 years old, with no previous experience on mobile robots and haptic devices voluntarily participated in the first experiment, while

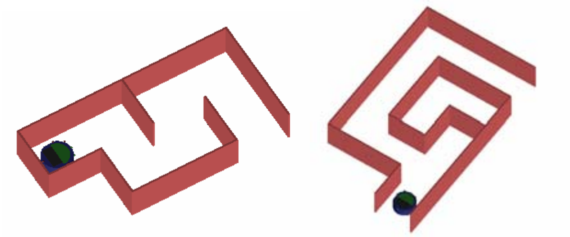

Fig 7. Examples of mazes for Experiment 1.a (left) and 1.b (right)

twelve of them were randomly selected to participate in the second experiment.

Both experiments were performed on a simulated mobile robot, with a virtual world (the slave robot environment) on a separate remote computer. Also, a PHANTOM Omni [10] haptic device was used for force-feedback generation in the master teleoperation interface.

The environment we used was a simulated world created for the purposes of the experiments. Different mazes of the same difficulty (with respect to the number of turns and the length of the preferred path) were created and pseudo-randomly selected so that no participant would perform on the same map twice.

The conducted experiments are explained below.

A. *Experiment 1: Comparison of the two interfaces*

Participants were asked to drive the robot through a number of mazes using the following two methods:

- **CCI**: Combined Command Interface with force feedback. This is the proposed method explained in the previous Sections.
- **SCI**: Single Command Interface with no force feedback. In this interface, the operator can send only one command through the haptic device (no bidirectional areas and combined commands exist) and feels no haptic feedback from the obstacles. A collision detection behavior is activated to avoid collisions with walls, but no wall-follow behavior is used. This method is the most common approach in the field of mobile robot teleoperation and a standard benchmark to proposed interfaces.

In both methods, participants could watch the map of the environment being constructed on the client screen.

Each participant performed three experiments using CCI method and three using SCI method. Half of the participants were asked to perform the first three experiments with CCI interface and the other half with the SCI interface. Each participant had a training session before the experimental sessions in order to feel comfortable with each interface and to understand the required task.

In order to adequately compare the two methods, two different sets of mazes were used. Eighteen participants performed on small mazes with many corners (Exp. 1.a) and eighteen on bigger mazes with fewer corners (Exp. 1.b). In each group, half of the participants performed first with CCI and the other half first with SCI.

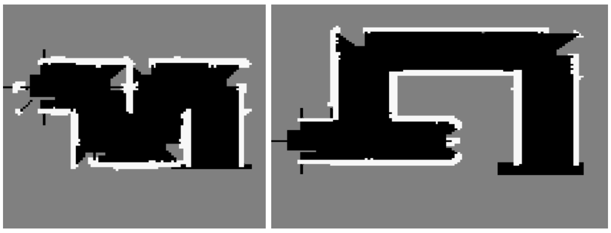

Fig 8. Maps of the mazes of Figure 6

B. *Experiment 2: Drive through a maze with no visual feedback*

In this experiment, participants were asked to "drive-by-feel" the mobile robot through a maze of Exp 1.a and find the exit of the maze, without any visual information about the robot environment. The only visual information provided was the robot orientation. Obviously, SCI method fails on such an experiment as it is based solely on the visual feedback, so only the CCI method was tested.

VI. EXPERIMENTAL RESULTS

During the experiments, we measured the time needed by the participants to find the exit of each maze. Number of collisions, a common measurable factor, is not a case here, as the collision avoidance behavior prevents robot from colliding to walls in both navigational methods.

A. *Experiment 1*

In Table 1, we present the results of Experiment 1. These results show that CCI method does not suffer from the common handicap of haptic driven teleoperated systems, which is the increase in the navigation time. From this table below, we observe, on the contrary, that CCI method allows participants to navigate through the maze in less time than SCI method. The results are statistically significant, particularly for Exp 1.a.

TABLE 1. EXPERIMENTAL RESULTS OF EXPERIMENT 1

Experiment	Method	Avg. Time (sec)	Standard Deviation	p-value
Exp. 1.a	CCI	58.72	2.65	0.0017
	SCI	60.82	4.59	
Exp. 1.b	CCI	83.86	10.56	0.0165
	SCI	86.93	10.53	

In Fig 8, we present an example of two snapshots of the maps created for the mazes shown in Fig 7.

B. *Experiment 2*

The task in Experiment 2 was obviously more difficult than that in Experiment 1. Nevertheless, although no visual feedback was present, all participants succeeded in navigating through the maze. This suggests that the proposed interface improves operators perception of the environment. The operators can detect corners without having to change robot orientation or to diverge from a desirable path. Participants intuitively followed a strategy according to which they alternated between left and right wall following, feeling the walls on the two sides of the robot and thus detecting corners. By replicating this strategy they managed to find the exit.

TABLE 2. EXPERIMENTAL RESULTS OF EXPERIMENT 2

Experiment	Method	Time (sec)	Standard Deviation	p-value
Exp. 2	No Visual CCI	81.17	10.3	0.0007
	Visual CCI	58.72	2.65	

In Table 2, we present the experimental results of Experiment 2. As expected, navigation time is significantly increased. This is mainly due to the fact that participants might lose some time on corners by checking which the correct turn is.

VII. CONCLUSIONS AND FUTURE WORK

In this paper, a mobile robot teleoperation system with a visuo-haptic interface has been presented. The task considered is that of remotely driving a mobile robot to perform an exploration task for an unknown maze-like rectangular environment.

The visual part of the master control interface comprises a sequentially created occupancy-grid map of the robot environment, together with a visual representation of the issued commands and current robot status. The haptic interface is based on a virtual spring-type joystick model, with the generated feedback force depending on a local (master side) occupancy-grid model of the remote environment. Two types of haptic commands are developed and supported by the teleoperation system: simple (direct) and combined (active/guarded) motion commands. The user can feel a *haptic equivalent* for both types of teleguidance motion commands, and can also observe in real-time the sequential creation of the remote environment map.

A comparative experimental analysis has been performed to test the effectiveness and confirm the validity of our approach. The proposed visuo-haptic interface is found to improve navigation time. A conclusion that can be drawn is that the proposed system improves operators perception of the environment and makes a step towards facilitating and automating exploration tasks.

In our future work, we plan to extensively test the proposed system using a real Hemisson robot. Initial experiments (Fig. 9) showed that the proposed approach works sufficiently well in this case too. We also intend to generalize the system architecture so as to successfully cope with applications involving any type of unstructured environment. Also, we are planning to examine system behavior under time delay conditions and develop algorithms to cope with such latency in master-slave round-trip communication.

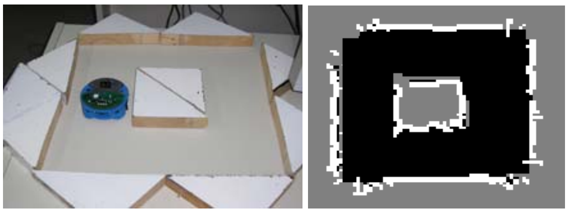

Fig 9. Real-world environment and the constructed map

ACKNOWLEDGEMENT

This work was partially co-funded by the European Commission and the Hellenic General Secretariat for Research and Technology (GSRT) under Measure 3.3 of the Operational Programme "Information Society" in the 3rd Community Support Framework (National Project Name: DIANOEMA, ID: 35).

REFERENCES

[1] F. Smith, D. Backman, and S. Jacobsen, "Telerobotic manipulator for hazardous environments," *Journal of Robotic Systems*, 1992

[2] Q. Lin and C. Kuo, "Virtual tele-operation of underwater robots," in Proceedings of *IEEE International Conference on Robotics and Automation*, 1997.

[3] O.S.Palafox, "Force Feedback Teleoperation of Mobile Robots over the Internet", *Technical Report*, Teleoperation Research Group, Coordinated Science Laboratory, University of Illinois. [http://decision.csl.uiuc.edu/~robotics/index.htm]

[4] T. Fong, F. Conti, S. Grange, and C. Baur. "Novel interfaces for remote driving: gesture, haptic and pda". *SPIE Telemanipulator and Telepresence VII*, 2000.

[5] S. Lee; G.S. Sukhatme, G.J. Kim, C.M. Park, "Haptic Teleoperation of a Mobile Robot: A User Study", *Presence: Teleoperators & Virtual Environments*, June 2005.

[6] Barnes D.P., and Counsell M.S., "Haptic communication for remote mobile manipulator robot operations". In American Nuclear Society, Proc. *8th Topical Meeting on Robotics & Remote Systems*, 1999.

[7] N. Diolaiti and C. Melchiorri, "Teleoperation of a mobile robot through haptic feedback," in Proceedings of *IEEE International Workshop on Haptic Virtual Environments and Their Applications*, 2002.

[8] The "Hemisson" webpage: [http://www.hemisson.com]

[9] N. Mitsou, "Probabilistic Methods for Mobile Robot Mapping", *Master Thesis*, National Technical University of Athens, School of Electr. & Computer Eng., 2004.

[10] [PHANTOM] The Phantom^{TM} webpage: [www.sensable.com/products/phantom_ghost/phantom-omni.asp]