A System for Vision Based Human-Robot Interaction
(Short Paper)

Alessandro Saffiotti, Dimiter Driankov, Tom Duckett
Center for Applied Autonomous Sensor Systems
Dept. of Technology, Örebro University, S-70182 Sweden
{alessandro.saffiotti, dimiter.driankov, tom.duckett}@aass.oru.se

Abstract

We describe our initial steps toward the realization of a robotic system for assisting fire-fighting and rescue services. The system implements the concept of shared autonomy *between the robot and the human operator: the mobile robot performs local navigation, sensing and mapping, while the operator interprets the sensor data and provides strategic navigation goals.*
Keywords: *Shared autonomy, mobile robots, fire-fighting, rescue robotics, mapping.*

1 Rationale

The Swedish fire-fighting and rescue services have identified the use of mobile robots as *remote amplifiers of the perception capabilities* of field personnel. This refers to the safe and reliable collection and communication of information at different levels of abstraction, which is relevant for the successful execution of a task. Target tasks include fire-fighting, search and rescue, and inspection of human-unfriendly sites. Easy, reliable, and task-dependent means of human-robot interaction were identified as crucial aspects to enable the field deployment of mobile robots.

The most important reason for a human-in-the-loop, as opposed to a fully autonomous robot, is psychological: the system should be perceived and accepted as a "trusted partner" by the rescue personnel. Two preferred interaction modes with such a partner have been identified by the field personnel: given a view from a camera and a view of an online local map of the environment, (i) point-and-click on the map or in the image to indicate locations where the robot should go, and (ii) point-and-click on interesting objects seen in the camera image to include them in the map. By "online local map" we mean a map of the space around the path travelled by the robot, built by the robot itself.

The realization of these particular modes of interaction is the subject of our presentation. In the initial phase of our work, we are considering a task in which a professional fire-fighter must remotely operate a robot (see Figure 1) in order to identify and localize gas tubes in an indoor environment after a fire.

Figure 1: A sample task: use a remotely operated robot (top) to identify and locate gas tubes presenting a risk of explosion (bottom). (Courtesy of Södertörns fire brigade, Sweden.)

In this note, we give a brief summary of the overall system architecture and of its main technological components, and we overview the system operation.

2 System Architecture

The architecture that we have adopted in our initial development is based on the "Thinking Cap".[1] The Thinking Cap is a framework to build complex controllers for autonomous and semi-autonomous robots. It provides ba-

[1] The "Thinking Cap" [5] is a joint effort between the University of Örebro, Sweden, and the University of Murcia, Spain. See http://aass.oru.se/~asaffio/Software/TC/.

In: Proc. of the IEEE Int. Workshop on Safety, Security, and Rescue Robotics (SSRR-04)
Gustav Stresemann Institut, Bonn, Germany, May 2004

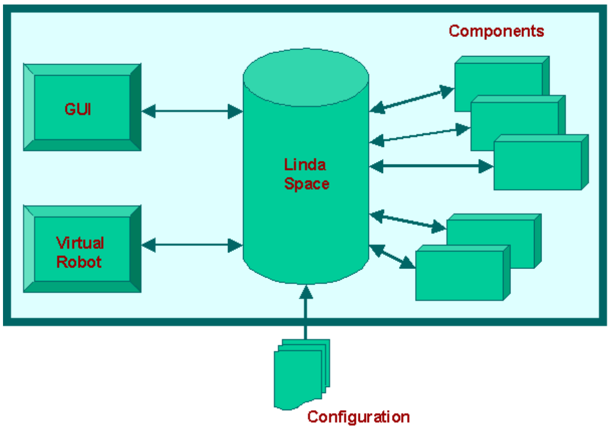

Figure 2: The "Thinking Cap" architecture adopted in our system.

sic storage, execution and communication mechanisms, together with a library of components implementing some of the functionalities needed for autonomous and semi-autonomous navigation. The Thinking Cap is modular and highly customizable, and it has been used on a number of different platforms, domains, and tasks (e.g, [5, 6, 2, 3]).

The version of the Thinking Cap used in this project is shown in Figure 2. It includes a number of independent software components that communicate through a centralized Linda space. The Linda space is a tuple-based blackboard that allows exchange of information by a publish-subscribe-notify mechanism. Two components are always present in the Thinking Cap: the Virtual Robot, which provides a uniform abstract interface to the sensors and actuators on the physical robot; and the Graphical User Interface. The other components used in our instance of the Thinking Cap are discussed below.

3 Main System Components

The main components currently implemented in our system are:

- **Map Builder** It incrementally builds a map (occupancy grid) of the environment from laser data. At the same time, it estimates the robot's position in this map using a multi-level relaxation algorithm [1].

- **Path Planner** It takes navigation goals from the user by point-and-clicking on the map. It plans a path across the free space, and regularly updates this path as new parts of the environment (obstacles) are detected by the robot and included into the map.

- **Path Follower** It moves the robot along the planned path, and reactively adapts to new or dynamic obstacles. The integration of path following and reactive obstacle avoidance is achieved using fuzzy-logic behaviors [7]. The controller can be overridden by the operator using a virtual joystick.

- **Video Streamer** It shows the images taken by the camera. The user can point-and-click on locations to visit or on objects to be included in the map. An image interpretation layer will be added in the future to detect and highlight possible objects of interest for the user, and suggest possible identifications.

4 Example of Operation

We are in the process of preparing a proof-of-concept demonstration system, by which a professional rescuer will operate a remote robot. The target task is the gas-tube finding problem mentioned above. The demonstrator will be run on our iRobot ATRV-Jr robot, shown in Figure 4.

At the current stage of development, the user is presented with the interface shown in Figure 3. The leftmost window represents the robot, in top view, and the sensor readings in the vicinity of the robot. This view is robot-centric: the robot is fixed at the center of the window, and the sensor readings give an outline of the environment around it. This view makes it easier to joystick the robot around if the operator wishes to take full control.

The middle window displays the current map built by the robot, in world-centered coordinates. The map is built incrementally as the robot acquires more sensor data. This map gives the operator a view of the shape of the environment and of the objects in it, together with the awareness of the robot's position in the environment. In this example, the robot has just explored the corridors around an intersection, and it is coming back to the intersection. The operator has cliked on a point in the left corridor (red dot) in order to instruct the robot to enter that corridor. The path plan-

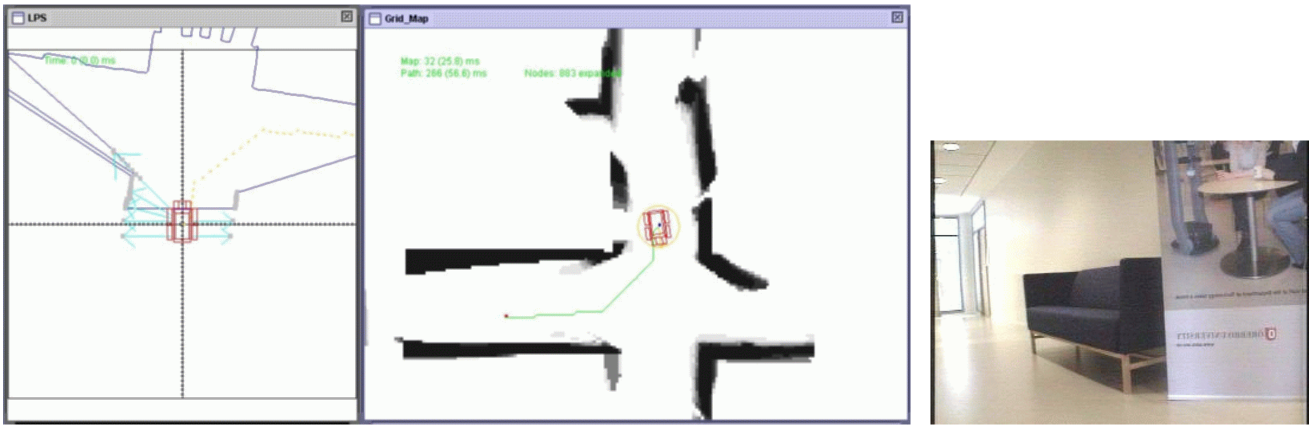

Figure 3: The operator interface. The leftmost window displays the sensor readings in the vicinity of the robot; the middle window displays the current environment map built by the robot; the rightmost window shows the images taken by the robot camera.

ner has generated a path (in green) to that point through the free space, which the robot is following. The path is recomputed every second in order to account for new obstacles that are detected by the robot's sensors.

The rightmost window shows the images taken by the robot camera. The user can point-and-click in this window as well in order to indicate a location where the robot should go. The clicked point is translated to global world coordinates and included in the robot map, and a path to that point is computed as above. In the future, the user will be offered the possibility to click on specific objects in the image in order to put them in the global map.

The system does not include any sophisticated processes dedicated to image analysis, scene understanding, high-level decision making or planning. According to the concept of shared autonomy [4], these high-level functionalities are assumed to be provided by the human operator, while the remote robotic system takes care of local navigation, sensor processing, and mapping.

Figure 4: The ATRV-Jr robot used for demonstration.

5 Conclusions

Our project has just started. Not all the components above have been implemented and incorporated into the system yet, but we are preparing some initial experiments aimed at testing the usability of shared-autonomy from the point of view of professional rescue personnel.

The two main inspiring guidelines of our effort are:

1. From the application point of view: to work in close co-operation with rescue personnel in order to match their needs and their preferred mode of operation, while avoiding the "academic exercise syndrome".

2. From the technological point of view: to build a shared autonomy system in which cognitive and physical functionalities are distributed between the robot and the human operator.

Acknowledgements

The work reported here is partially supported by the Swedish National Center for Innovative Recue and Security Systems (NIROS) and by the Swedish KK Foundation.

References

[1] U. Frese and T. Duckett. A multigrid approach for accelerating relaxation-based SLAM. In *Proc. of the IJCAI Workshop on Reasoning with Uncertainty in Robotics (RUR)*, pages 39–46, Acapulco, Mexico, 2003.

[2] A. Gomez, H. Martinez-Barbera, M. Zamora, B. Ubeda, F. Gomez de Leon, and L.M. Tomas. MIMICS: Exploiting satellite technology for an intelligent convoy. *IEEE Intelligent Systems*, 17(4):85–89, 2002.

[3] H Martinez-Barbera, J.P. Canovas, M. Zamora, and A. Gomez. I-Fork: a flexible AGV system using topological and grid maps. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, Taipei, Taiwan, 2003.

[4] P. Michelman and P. Allen. Shared autonomy in a robot hand teleoperation system. In *Proc. of the Int. Conf. on Intelligent Robotics Systems (IROS)*, pages 253–259, 1994.

[5] A. Saffiotti. *Autonomous Robot Navigation: a fuzzy logic approach*. PhD thesis, Université Libre de Bruxelles, Brussels, Belgium, 1988.

[6] A. Saffiotti, A. Björklund, S. Johansson, and Z. Wasik. Team Sweden. In A. Birk, S. Coradeschi, and S. Tadokoro, editors, *RoboCup 2001*. Springer-Verlag, Germany, 2002.

[7] A. Saffiotti, E.H. Ruspini, and K. Konolige. Blending reactivity and goal-directedness in a fuzzy controller. In *Proc. of the 2nd IEEE Int. Conf. on Fuzzy Systems*, pages 134–139, San Francisco, CA, 1993.