Mixing Reality and Virtual Worlds in an Educational Mobile Robotics Remote Lab

Riccardo Cassinis
Department of Information Engineering, University of Brescia, Via Branze 38, I-25123 Brescia, Italy

Keywords: Educational Robotics, Remote Labs, Augmented Reality, Internet-based Teaching.

Abstract: This paper describes MOBOLAB, a project aimed at the construction of a remotely controlled mobile robotics laboratory. MOBOLAB was primarily designed to aid educators who wish to use robotics as an educational tool for pupils ranging from elementary to high school, and who don't have educational robotic equipment readily available at their place, or who wish to use a standardized environment offering several useful features to enhance their teaching activity. MOBOLAB also offers other interesting usage possibilities, such as on-line training of educators, student robotic competitions, etc.

Although far from being complete (in fact, MOBOLAB was designed as an ever-expanding project), some interesting results have already been obtained from practical experiments performed with pupils and educators.

1 INTRODUCTION

During the last few years the importance of robotics, especially mobile robotics, at all levels of education has become unquestionable (Johnson, 2003). Robots are being used to teach not only the principles of robotics, but also a variety of concepts and notions spanning completely different areas, such as computer programming, geography, mathematics, etc.

The goals that can be achieved using mobile robots in the classroom are multiple, and have been discussed by several authors; they range from the understanding of what a mobile robot is and what problems it has to face in the real world (perhaps comparing them with the more structured world in which industrial manipulators operate), to the basic principles of programming (Seymour Papert's Logo (Papert, 2005) was a great precursor in this field), to a quite amusing and unpredictable teaching aid for other subjects, such as geography or math, if the robot is wandering over a map or a table that bears right and wrong solutions for a given problem.

It often happens however that schools are not equipped with robotic laboratories, and educators are not yet used to robots and require specific training. Investing in robotics equipment, especially in small schools, is often unaffordable or at least economically un-convenient.

On the other hand, remote real and virtual labs that can be accessed via the Internet have become very popular in several fields, including robotics. They offer a standard environment, possibly including also sophisticated equipment, at a very low cost given the scale factor (a single lab can be used to satisfy the needs of several schools). On the user side, they require standard devices—actually, a PC and an Internet connection usually are all that is required to start—that are already available in most cases.

2 THE PROBLEM

Virtual and remotely accessible labs are not new, not even in the field of robotics (Khamis et al., 2003). However, due to several reasons, most of the existing robotics laboratories are related to manipulation, rather than to mobile robotics.

Furthermore, they are often aimed at high school or college students, while the system described here is specifically targeted to younger pupils (up to an age of about 15 years) and to their teachers.

A distinction should be made between virtual and physical labs. The former serve data from a computer simulator of the system to be studied. Some of these systems offer very sophisticated services, as the MIND Project

Cassinis R..
Mixing Reality and Virtual Worlds in an Educational Mobile Robotics Remote Lab.
DOI: 10.5220/0004390001480153
In *Proceedings of the 5th International Conference on Computer Supported Education* (CSEDU-2013), pages 148-153
ISBN: 978-989-8565-53-2

(http://www.mind.ilstu.edu). Anyway, in most cases, it is useless to resort to a remote simulator, since a local one can be used, unless high computing power is required. The only reason for using a remote simulator is when people located far from each other must work together or compete. This happens for instance in computer simulated games as the simulated soccer league of Robocup (Kitano et al., 1997).

Examining the pro's and con's of mobile robot simulators is not in the scope of this paper, but it can be said that, in general, they don't give users, specially the younger ones, the feeling of something physical really happening somewhere in the world. Users tend to quickly lose interest and the educational result is quite poor.

Most remote labs, on the other hand, are devoted to experiments where some physical device or measuring instrument is connected to a computer. In this case, "localizing" the experiment would require the acquisition of (often expensive) instruments. Commands issued by the person performing the experiment and measured data are directly transmitted through a computer, and this requires quite a simple procedure to provide the user with a suitable interface. In other words, the connection between the user and the real world takes place in the form of well-standardized methods and protocols.

Remote robotics labs, on the other hand, have different requirements. This is because the robot physically interacts with the real world through non-standardized, poorly modeled interfaces (the gripper in the case of a manipulator arm, wheels or legs in a mobile robot). The effects of a user-initiated action cannot in general be anticipated, and in most cases not even measured. This poses at least two additional requirements:

a) The user must be able to see, and often also to hear what is happening in the lab;

b) The lab must be capable of automatically returning to a known initial state, no matter how wrong the received commands were.

The above requirements are quite hard to meet when manipulators are involved, and usually require that some human assistance be available at the lab site. In the field of mobile robots instead, they can be more easily met if the environment and the robots are simple and carefully designed in order to avoid entanglements, robots capsizing and other major accidents.

Additionally, mobile robots need some means for recharging themselves when they are not being used. The recharging procedure should be fully automatic.

As it was said, the alternate choice of using software simulators doesn't seem to be very appealing, because simulators cannot fully replicate the real world and their users are often unsatisfied and get quickly bored (Tzafestas et al., 2006). On the other hand, the real robots, even if at a remote location, are much more appealing especially to younger people, and obviously perform in a more realistic way.

A number of very interesting and inspiring realizations in this field is already available (Guimaraes et al., 2003), (Casini et al., 2008), (Casini et al., 2009), (Casini et al., 2011): however, the project described here has some original characteristics such as the capability of being remotely controlled and programmed in different ways to accommodate the needs of different users, and the use of some augmented reality to enhance the performance of the whole system.

3 DEFINING PERFORMANCE LAYERS AND SERVICES

As it was said, MOBOLAB project is targeted to the needs of different users: students of various grades on one side, educators on the other. It must therefore be structured in such a way as to behave differently and to allow different activities, depending on the chosen level.

3.1 Common Services

From the users point-of-view, MOBOLAB is a web server that can be accessed using an ordinary web browser. A few common services, available from the home page (Figure 1), have been established to allow easy usage of the lab. These services include:

- An authentication mechanism, to allow only registered users to access the system at various levels, according to their authorization level;
- A booking system, as the lab was designed to be used by a single user (or group of users) at a time;
- A forum, which can be used as a source of information (descriptions, user manuals, etc.) and as a place for discussion among users.

All these components were implemented using off-the-shelf free software components (SMF for authentication and forum, MRBS for the reservation system).

3.2 Layered Services

In order to satisfy the needs of different classes of users, the following service levels have been defined so far:

Observation. The user at this level can't interact with the system. He/she can only observe and listen what is happening at the remote site. This level is mainly intended for demonstrations, where an instructor does all the teaching and pupils attend remotely. If bandwidth allows, this mode can be augmented with a Skype group call to enhance the presence effect.

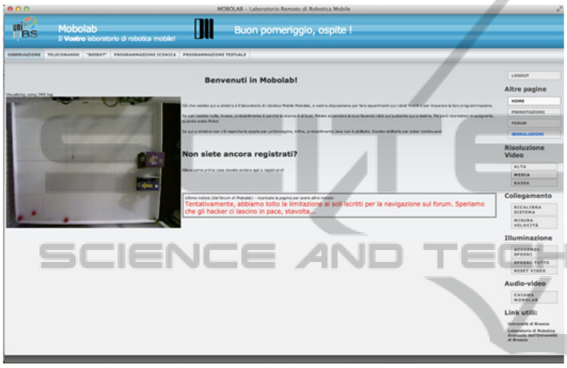

Figure 1: MOBOLAB home page.

Tele-operation. In this mode, the user can remotely control movements of a robot. At the present stage, a simple non-holonomic robot built around a Lego NXT brick is being used, that is controlled using a number of virtual buttons on the user's screen, as it can be seen in Figure 2. No programming is available in this mode, and the only automatic function available is a "return home" function that can be called at any time.

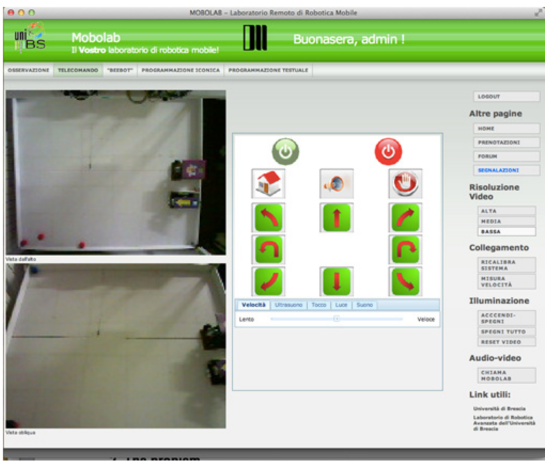

Figure 2: Tele-operation layer.

In addition, the user has a chance of getting some sensor data from the robot, and to set some parameters as rotational and translational velocity.

"BeeBot" programming. This mode (Figure 3) allows an emulation of the BeeBot robot (Demo, 2008). This machine was designed for the first approach to mobile robots, and consists in a bee-shaped robot bearing some pushbuttons on its back. These buttons allow programming movements on a flat surface divided into uniform squares (the robot can only move one square forward, one square backward, or turn 90° right or left. Steps described pushing these buttons are stored and the whole program can then be played as a sequence of movements, closely replicating some features of the Logo turtle.

This level currently uses a second robot, built around a Lego NXT brick, which was specifically designed for this purpose.

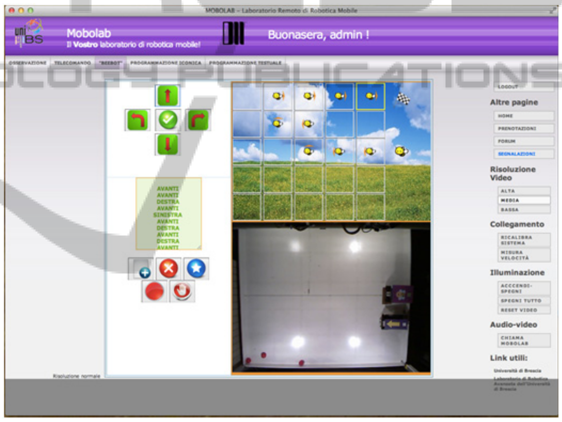

Figure 3: "BeeBot" layer.

Iconic Programming. Iconic programming is achieved using the classical Lego Mindstorms NXT-G programming language. So far, it has been implemented replicating a remote display mechanism based on VNC, and uses the same robot used for tele-operation layer.

Textual Programming. Textual programming can be achieved using NXC language. A very simple interface has been built that allows editing, compiling, uploading and executing programs written in NXC on the same robot used for tele-operation. Also this layer uses the tele-operation robot.

4 MAIN DESIGN ISSUES

Once the basic idea that real robots should be used instead of a simulator was established, it became clear that some efforts should be devoted to optimizing resource exploitation and to maximize cost-to-benefit ratio.

4.1 Bandwidth Optimization

Obviously, as far as bandwidth is concerned, video and audio transmission are the most demanding parts of the whole system. Luckily enough, in a normal configuration where the user is connected to the Internet via an ADSL connection, the fastest path goes in the right direction (towards the client). However, smaller bandwidth connections should also be taken into account.

The research followed these steps: the design criteria required that two video channels and one audio channel should be available. The video channels should carry images from two cameras placed in different positions over the lab, while the audio channel should provide acoustic feedback to the user and, being bi-directional, also allow communication with a human operator when he/she is present at the server's location.

Before establishing video and audio communication, the available connection speed is measured, and the most suitable image size and frame rate are automatically chosen.

4.2 Image Acquisition

The cost requirements of the system call for inexpensive components to be used whenever possible, and the imaging system is no exception. At the client side, low to medium resolution terminals will normally be found. Most often, a video projector or an interactive blackboard will be found due to the classroom usage, and the maximum display resolution can be assumed to be 1280x1024 px.

Since several pieces of information need to be displayed at the same time, in most cases there will be no need to resort to images at a resolution higher than VGA (640x480 px), and in many cases a resolution of only 320x240 px will have to be used in order to fit all images in the screen.

4.3 Image Processing

As testing of MOBLOAB began, it became clear that, as the lab can be used at various levels, different backgrounds were desirable. At the "BeeBot" level, for instance, kids would be amused by the possibility of switching between "natural" backgrounds (grass, flowers, etc.) and "artificial" ones (maps, arrays of numbers or letters, etc. as can be found for instance in http://www.terrapinlogo.com/bee-botmats.php). At higher levels, different tracks, obstacle-cluttered environments, etc. are desirable to perform different experiments.

The idea of mechanically changing the mat over which robots move was soon discarded because it is too complex and prone to faults, and it was decided to implement a virtual background system, following the technique commonly used in TV studios.

For this reason, a background subtraction system was implemented, that allows removing the background from images gathered by the cameras, and to substitute it with a still picture chosen by the user or by the system, according to circumstances.

So, while the real robots wander over a white floor, any static picture can be superimposed giving the feeling of the robot moving in a different

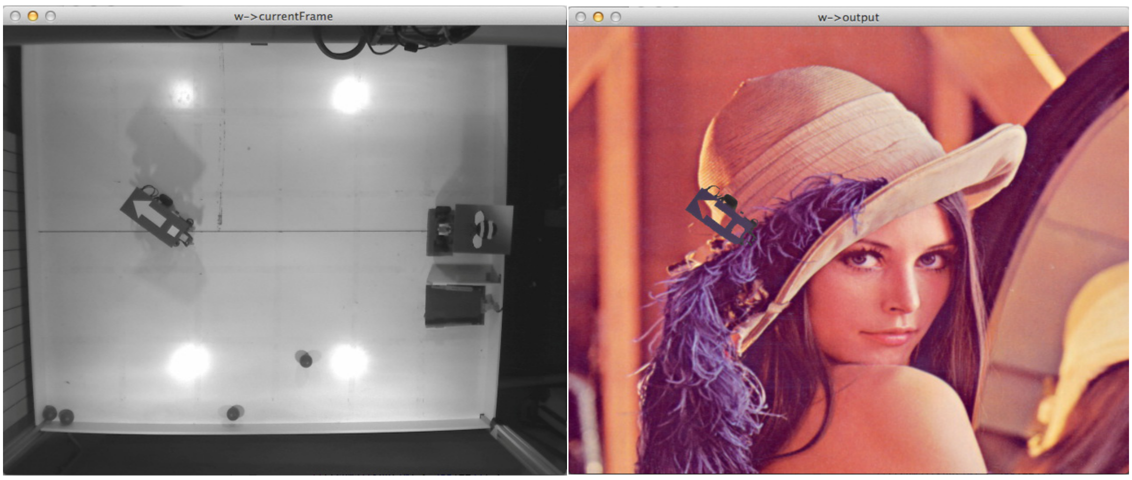

Figure 4: Background substitution demonstration.

environment, as it can be seen in Figure 4, where the white floor has been substituted by an image that is very popular among artificial vision researchers.

4.4 Pose Estimation

In its current implementation, MOBOLAB uses two robots built around Lego NXT bricks using Lego components. The precision attainable with such components is low, and position data gathered by odometry are unreliable even after short movements from a known position. This calls for an external localization system.

Luckily, the background substitution mechanism described above provides robot position data as a byproduct, and with some enhancements full pose data can also be obtained. Such data are used by several other parts of the system for driving and monitoring the robots. In order to achieve this, each robot was equipped with a unique set of passive markers, which can be recognized by the system and used to compute the robot's pose.

The used technique will also allow defining "virtual" obstacles, i.e. obstacles that can be "seen" by sensors but do not exist in reality, using a method similar to the one described in Casini et al., 2012.

4.5 Further Bandwidth Reduction

As the last two parts were completed, the idea arose that position data could also be used to synthesize robots images. In other words, the idea was that the background and each robot's image should be transmitted only once, leaving the task of correctly placing the robot's image over the background to the client, with the server providing only real time position data. This of course would almost totally remove the feeling of watching real robots, but would in turn dramatically reduce the required bandwidth, making the use of the system possible even with very poor Internet connections.

Moreover, the user (or the system) can very easily switch among the various combinations offered by the system (real or synthetic background, real per synthetic robot images, thus making the system able to cope with a number of different situations.

4.6 Other Issues

As MOBOLAB, in its current form, was clearly designed as a single-user system (there may be several observers in different places, but only one of them can be in control of the system at any given time), very simple lock mechanisms were implemented to prevent multiple users to physically access the system at the same time. Observation level, the forum and the reservation system are instead always accessible.

5 EXPERIMENTAL RESULTS

The system has so far been tested in a fourth grade elementary school class (9 years old kids), with a group of elementary school teachers (at a very basic level of robotic skills) and with some technical high school students.

All tests were performed using a laptop computer connected to a video projector, that allowed one person at a time to control the system, while all the other watched the screen (shouting suggestions in the case of younger kids).

The results seem to replicate the results reported in previous researches (Trevelyan, 2008), i.e. that remote laboratories can be very well used instead of local ones, and that the educational outcome is very good even with an extremely limited investment. An interesting consideration is that these systems seem to work best with younger people: our "digital natives" had no problem at all in learning how to use the simpler layers of the system (tele-operation and BeeBot emulation), while teachers experienced some difficulties even at these simple levels. Most problems were however related to the interface: for instance, some of the icons used for buttons had to be changed to make the more understandable to older people.

Similar results were obtained with a group of second grade pupils (7 years): in both cases the level of attention was extremely high, and at the end of the test (about three hours long) it was quite difficult to stop because the kids weren't tired at all yet.

The experiments performed with second and fourth grade pupils took place in a school located about 150 Km from MOBOLAB. Interestingly enough, no kid had any problem in understanding what was going on locally and what at the remote site. The question "do you think it is possible to control from here a robot that is located in the town of Brescia" got an unanimous affirmative answer, and some of the kids even explained how that could be achieved with a quite good technical precision, appropriately using terms as "server", "Internet", "webcam", etc.

High school students, on the other hand, were not particularly interested in the lower level services, and concentrated on the robot programming aspects.

However, they proved extremely helpful in debugging the man-machine interface and in suggesting some important improvements.

6 CONCLUSIONS AND FUTURE DEVELOPMENTS

An ongoing project was described, that aims at bringing mobile robots in schools even in underdeveloped or economically weak areas at a very low cost, using existing infrastructures and optimizing usage of equipment through sharing.

Although the to-do list still has countless items, the system is already useable and is being used in practical applications.

Among the most important additions, it is worth mentioning a better integration of virtual backgrounds with simulated sensor data, the substitution of the actual robots with more versatile ones (holonomic, and with a better-designed docking system for recharging), and a number of improvements on the user interface side.

REFERENCES

Casini, M., Chinello, F., Prattichizzo, D., Vicino,A., 2008. RACT: a Remote Lab for Robotics Experiments, *Proc. 17th IFAC World Congress, Seoul.*

Casini, M., Garulli, A., Giannitrapani, A., Vicino, A. 2009. A Matlab-based Remote Lab for Multi-Robot Experiments, *8th IFAC Symposium on advances in control education, Kumamoto*.

Casini, M., Garulli, A., Giannitrapani, A., Vicino, A., 2011. A LEGO Mindstorms multi-robot setup in the Automatic Control Telelab, in *Proc. 18th IFAC World Congress, pp. 9812-9817, Milano, Italy.*

Casini, M., Garulli, A., Giannitrapani, A., Vicino, A., 2012. A Remote Lab for Multi-Robot Experiments with Virtual Obstacles., *Advances in Control Education,* Vol. 9, No. 1, pp. 354-359.

Demo, G. B., 2008. Programming Robots in Primary Schools Deserves a Renewed Attention, *Emerging Technologies and Information Systems for the Knowledge Society, Lecture Notes in Computer Science, Springer*.

Guimaraes, E., Maffeis, A., Pereira, J., Russo, B., Cardozo, E., Bergerman, M., Magalhaes, M.F., 2003. REAL: a virtual laboratory for mobile robot experiments, *IEEE Transactions on Education, Vol. 46, N. 1*.

Johnson, J., 2003. Children, robotics, and education, *Artificial Life and Robotics* Vol. 7, N. 1-2, Springer-Verlag.

Khamis, A., Rivero, D. M., Rodriguez, F., and Salichs, M., 2003. Pattern-based architecture for building mobile robotics remote laboratories, *Proc. ICRA '03. IEEE International Conference on Robotics and Automation*.

Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I., Osawa, E., and Matsubara, H., 1997. RoboCup - A Challenge Problem for AI, *AI Magazine Volume 18 Number 1.*

Papert, S., 2005. Teaching Children Thinking. *Contemporary Issues in Technology and Teacher Education*, *5(3)*. AACE.

Trevelyan, J., 2008. Lessons learned from 10 years experience with remote laboratories, *Proc. International Conference on Engineering Education and Research "Progress Through Partnership", Ostrava.*

Tzafestas, C. S., Palaiologou, N., Alifragis, M., 2006. Virtual and remote robotic laboratory: comparative experimental evaluation, *IEEE Trans. On Education, Vol. 49, N. 3*.7.