Tele-Programming for a Mobile Robot via Internet

DORIN POPESCU, SORIN VAVA, DAN SELISTEANU,
IONUT DINULESCU, LIVIA CARMEN POPESCU
Department of Automation and Mechatronics, University of Craiova
Craiova, 107 Decebal Blvd.
ROMANIA

Abstract: - This paper presents a range of tele-experiments with a mobile robot with particular emphasis on microcontroller programming. The students learn the microcontroller programming, sensor use, path planning, trajectory following. The idea is for the students to be able to perform real experiments, in real time, on real equipment, but over the Internet. The remote user is connected by Internet to a dedicated web server that interacts with the computer of the laboratory used for controlling and monitoring the mobile robot.

Key-Words: tele-programming, mobile robot, Internet, microcontroller, telematics.

1 Introduction

Modern Multimedia and Telematics technologies offer good potential for tele-education in order to present the theory and practical experiments in an interesting and economical way ([1], [2], [3]).

The high cost necessary to implement experiments (for didactical purposes) with manipulators and robots led to the development of the tele-operating facilities, such that the physical robots, sensors, and operating environment can be remotely controlled via the Internet.

A tele-robotic operation involves interaction between a human operator and a remote robotic system via communication networks. The development of tele-operated systems has gained considerable attention in recent years due to new potential applications, such as remote production monitoring, remote exploration, manipulation in inhospitable environments, tele-surgery, tele-teaching, and so on.

Some researchers achieved an analysis of the current challenges in Internet tele-programming and tried to find the possible solutions under the current environment, where there are some unsolved common problems associated with this new technology such as limited bandwidth and unreliable signaling ([1], [2], [3], [4], [5]).

Remote laboratories are laboratory experiments that run remotely via a web interface. Usually, either the student can set some parameters on the web, then a software interface converts those parameters to a form that is accepted by the local computer running the experiment or he can tele-operate an equipment ([6], [7], [8], [9], [10]).

We proposed more, namely the student can achieve a program, which can run on the microcontroller in order to control a mobile robot. The student has thorough freedom to choose what experiment wants to do. The goal is remote programming of a control board, which controls a mobile robot for use in tele-education.

Web-based tele-robotic systems have only become available on the Internet in the last decade. In the beginning they utilized a Common Gateway Interface (CGI) to access the robots ([4], [10]). With the introduction of Java and its integration into web browsers, developers could create tele-operated systems that sustained an interactive link to the robot during its execution. Many new tele-robotic systems were created that gave the user much more control and provided functionality, which under CGI could have never been possible ([1], [3], [4], [12]).

The Internet lab technology offers to the students the opportunity to work with sophisticated equipment, of the kind they are more likely to find in an industrial setting, and which may be too expensive for most faculties to purchase.

This paper gives a solution for tele-programming of a certain mobile robot. It tries to show an aspect of higher education and training based on Information Technology. The final purpose of the remote laboratory is to allow the student to take full control of the equipment, in order to fulfill the task required by the teacher.

The rest of the paper is organized as follows. In Section 2 we describe the mobile robot and architecture of telematics application. The following section presents the telematics application. In section 4 we describe how the user can work with our mobile robot and the features of our tele-education application. Finally, Section 5 presents conclusions and future directions.

2. The hardware environment

A. The mobile robot

There are two criteria that allow establishment of a clear classification of the experimentation environments from the point of view of student use: the nature of the system (virtual or real) and the way of accessing the resources for experimental purposes (local or remote access) [1]. Our experimentation environment represents a real system with remote access. It represents access to a real equipment through the Internet. The user operates and controls, in a remote way, a real system through Internet.

One vital aspect of robotics education is the practical work necessary to give students a taste of real situations. The idea is for the students to be able to perform real experiments, in real time, on real equipment, but over the Internet.

Fig. 1. Mobile robot.

The mobile robot, presented in Fig. 1, is a didactical mobile robot for tele-operating and tele-programming via the Internet to test different control methods. A video network camera enables the user to see the robot's movements in real-time while a Java applet interface is used to send programs and to change data, commands with the mobile robot server. The commands, programs from the server to the microcontroller are transmitted through serial connection, and for the control of the actuators is used a BS2SX Basic Stamp microcontroller.

The mobile robot has two driven wheels and two castor wheels. It is a differential-drive robot.

This variant of mobile robot is not wireless. It has connections by cables with power supply and computer.

As sensors, the mobile robot has in this moment three IR sensors, sonar, and encoders. The sonar can be turn around (180 degree) by an actuator.

Teleoperation of the mobile robot really allows users to take the lab home, because the system is correctly built and it is possible to experiment from any computer connected to the Internet, any time of the day and any day of the year.

The remote user is connected by Internet to a dedicated web server that interacts with the computers of the laboratory used for controlling and monitoring the mobile robot.

B. Application Architecture

The goal of this project in the remote programming area is to discover and develop the system by combining network technology with capabilities of microcontrollers and mobile robots. The use of Internet technology for remote programming application offers the advantage of low-cost deployment. There is no longer a requirement for expensive purpose built equipment at each operator's location. Almost every computer connected to the Internet can be used to control a tele-operable device. The downside is the limitation of varying bandwidth and unpredictable time delays. These Internet features should be defined and considered before designing an efficient remote programming system. Besides that, several functional requirements should also be defined before designing any tele-operable system ([1], [5]).

This web-based application allows users to perform own experiments remotely from another computer. Using a standard web browser and a connection to the Internet, the user can write his program for the microcontroller in order to control the mobile robot. After this, the user uploads his program and the results can be viewed on screen. The experiments have video streams available for visual monitoring.

The main requirement of this application is to provide a friendly user interface that allows application developers to control the devices from anywhere in the world via a thin client that does not need the installation of any additional software on the user side.

Another requirement is related to the costs that are involved during the project development (e.g. development tools, hardware etc.).

After considering several alternatives, it has been opted for web based technologies that considerably reduced the implementation costs while accomplishing the other project requirements.

Thus, Java (for client) and Visual Basic 6 (for server) were chosen for the application development ([16], [17], [18], [19], [20]). In order to speed up the development process and eliminate software bugs, additional Java open source libraries and Application Programming Interfaces (APIs) have also been used. The role of each one within the

whole system will be explained in the next paragraphs. The general system structure is described in Fig. 2.
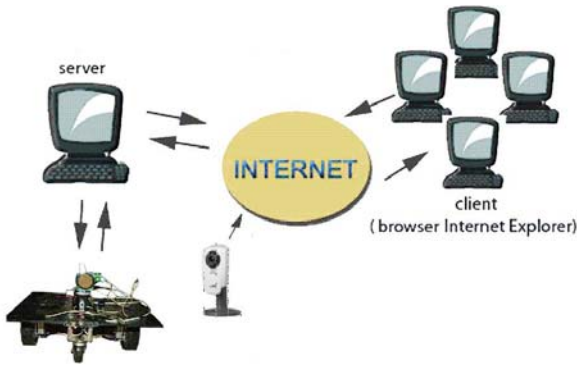

Fig. 2. Architecture of telematics system.

The only software needed on the user side is a conventional web browser that can run Java applets. This application has been tested and optimized to work on the most popular web browsers including Internet Explorer and FireFox, while other ones are currently being tested.

The Visual Basic (VB) server application is responsible for the application administration module, that involves users management, communication with control board, microcontroller program editor etc.

An AXIS 205 network video camera is used for sending live video of the mobile robot to the user's browser. This camera model includes a built-in WEB server running under Embedded Linux that transmits the video stream either via an ActiveX control or a Java applet. While the ActiveX transmission method gives better results, the Java applet is a good choice for browsers that don't support ActiveX technology (such as Opera or Konqueror).

3. Telematics application

The application is achieved using Java and Visual Basic and contains two modules: the client (Java) and the server (VB). The client application is an applet hosted by website department. The server application is hosted by a computer connected to the mobile robot via serial connection.

We built the client in Java language as an applet, because of the advantages of Java programming language, which most important of them we considered to be portability on other platforms, integration in almost all Internet browsers and of course open source feature. The APIs are all part from Java standard core.

We build the server program in Visual Basic, so that the one who has to install the server program does not have to be an advanced user of the computer, because in Java, he would have had to install a API separately from the server, for the serial communication, so in VB all the libraries that we needed were integrated in it.

The client window is presented in Fig. 3. The client application offers various buttons and an editor window for instruction list.

The buttons are used for next functions:

- Send Program - after the program was written in the editor window or open, it can be uploaded on the server using this button; the server will check the program and then it will sent to control board

- Open Program - to open a program which it was achieved using editor window of applet and saved through Save Program button

- Save Program - to save an instruction list program edited in the editor window

- Read IR Sensors – to open a window that read values from sensors

- Read Sonar - to open a window that shows the environment from front of the mobile robot (the sonar's actuator begins to turn it)

- Read Sonar (by angle) – to open a window (see Fig. 5) where the user can choose an angle for that is read the value from sonar (the distance from mobile robot to obstacle)

- Motion Control – to open the window for tele-operating of the mobile robot

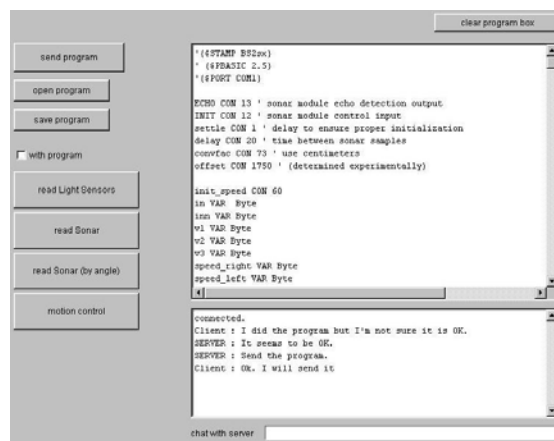- Clear Program - to clear the editor window

Fig. 3. Client window.

Chat with server allows the users to change messages with the supervisor of the mobile robot. All the messages will appear in the messages' window.

The editor window allows writing the microcontroller program as instruction list. A syntax

checking system was developed. If the user writes a wrong instruction mnemonic or a wrong variable, he will receive an error message.

The error message window indicates the number of line where it is the error and the wrong instruction mnemonic or the wrong variable. In this way it will be easier for user to correct the program.

The server application is responsible for the application administration module that involves users' management, communication with control board of the mobile robot, robot status, etc. Only one user can be connected to server application (it means to mobile robot). First time we didn't introduce restriction for user time. But we saw it was not well. Therefore we introduce a limitation of user time to 30 minutes if there is another request.

The server sends the program to microcontroller at the transmission rate of 9600 bps.

The server has a window (Fig. 4) that allows to see the uploaded user program and another window in order to change messages with the user. The robot's supervisor can see the IP address of users' computer, too.
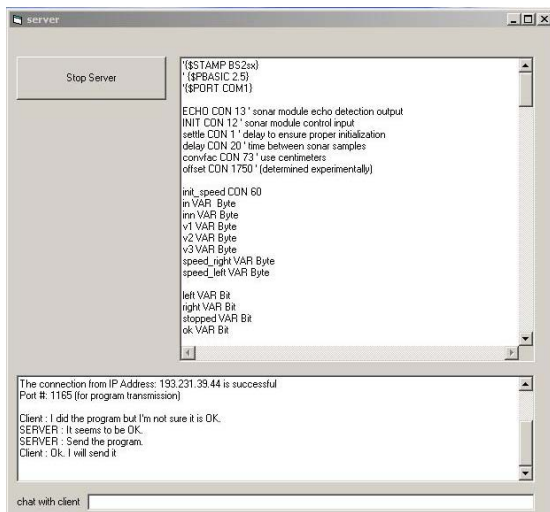

Fig. 4. Server window.

The sonar's actuator turns the sonar for 180 degrees and with Read Sonar button the distance from mobile robot to obstacles is read at each 2 degrees.

The student can write his programs for motion control or read sensors. In this way the user can achieve programs such that the mobile robot to avoid obstacles or to follow a trajectory.

Read Sonar (by angle) button allows to open a window (Fig. 5) where the user can choose an angle (for example 94 degrees) for that is read the value from sonar (the distance from mobile robot to obstacle, 113 cm).
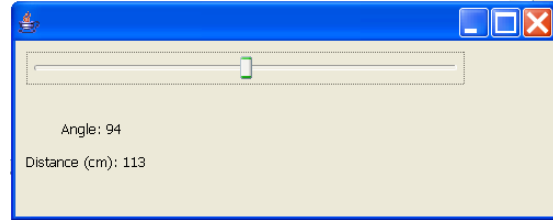

Fig. 5. Window that gives the distance from mobile robot to obstacle (113 cm) for rotation angle of sonar (94 degrees).

Many on-line robots allow tele-operating facilities. Our first goal was to offer more - tele-programming. But then we implemented the tele-operating, too. With a particular window, the student can move the robot through its environment. The user can choose the same speed for both motors of wheels or he can choose different speeds for them. The mobile robot can be moved forward, backward, left and right using the buttons.

4. Features of tele-education application

A user with a computer and Internet has to work in the following way after he navigate through our website: he has to ask permission to work with our mobile robot; he will receive an ID and a password; he will connect to our server and he can read or download our documentation about system (mobile robot, sensors, microcontroller); he has to answer to some questions and then he can write a user program with the original language of the microcontroller; the application points out the syntactic errors; he can send his user program. The user program for mobile robot will be run and the user can observe the movements and operation of the mobile robot by means of a network video camera (Fig. 6).

Fig. 6. Image from Axis 205 network video camera.

To perform an experiment with our system the user has to read the documentation about the mobile robot and microcontroller, because before the remote experiment he has to answer some simple questions. Only if the user answers correctly he can access the user interface, writes a program and run it.

The user can write programs for the mobile robot in order to solve some problems such as: trajectory following, obstacle avoidance, path planning, PID control, etc.

Tele-education technology implements a distributed education environment independent of time, space, and individuals.

Some advantages of our mobile robot tele-education application are:
– self learning for work
– advanced training system
– facilities for studying independently of both time and location
– appropriateness of the subject to remote operation
– better presentation of complex material by using multimedia elements
– tele-control of remote device and application
– testing robot with regard to the accuracy in positioning
– PID control
– remote surveillance (through webcam & chat server)
– the number of users is no restricted
– tele-education permits continuing education with minimum costs.

Some basic features of our experimentation environment are:
– access to our remote environment was carried out exclusively via the Internet. A web navigator is the only necessary software tool that students need when undertaking a practical experiment;
– the video network camera window allows users to see the movements of the mobile robot in almost real-time;
– students need not to acquire any additional software. The only expense that they will have is access via the Internet, and, when they work in the faculty laboratories, they will have no expenses;
– our remote mobile robot promotes students' interest and motivation when they obtain real and coherent responses. The system reacts in almost real time to any action from the student;
– the chat window allows students to discuss the experiments with the robot's supervisor;
– There is no time restriction (after the student received his ID). The only limitation is that which is created by another user when he occupies the mobile robot or when maintenance work is being done; the time window shows the available time of the user in order to control the mobile robot or the waiting time until the experiment will be available.

5. Conclusion

This project is currently being used at University of Craiova (in the Robotics class). Such remote laboratory experiment method enables the student to use laboratory equipment, which is not usually available to the students. Students can watch the mobile robot via the network video camera.

The application has been tested inside the faculty where a computer has been connected to Internet that the user had used to program the microcontroller in order to control the mobile robot which was in another laboratory. But application was tested in a small contest, too. Five teams of two students had to solve a problem of robot control through tele-programming of the mobile robot.

For the majority of remote experiments via Internet in mechatronics, the student can tele-operate a robot or set some parameters on the web and then he can watch the robot movements. Such type of application was developed for example to University of Applied Sciences Ravensburg-Weingarten, Germany, by Prof. Schilling and Popescu [8]. We performed more, namely the student can write a program, which can run on the microcontroller in order to control the mobile robot as he wants.

The application is provided with friendly user interface. The exercises help students to understand and compare different control methods using a microcontroller.

The use of on-line experiments with real mobile robot changed the students' attention and interest in robotics as well as the robotic contest. It will be created a questionnaire about new methods used in robotics training. Some of the questions will be about their interest in robotics, understanding of theoretical and practical problems previously studied and their interest in real, simulated, virtual and on-line experiments before and after Robotics course.

As future research and developments we intend to do:
– examples of obstacle avoidance and path planning experiments
– advanced user interface design based haptic feedback and virtual reality methods
– to process the images from web-cam
– example with achievement of map of limited environment

These will imply more applications in tele-education, tele-operation, tele-programming, tele-maintenance.

The Internet represents a major opportunity for robotics education: the development of remote laboratories where students can gain practical experience by efficient use of hardware resources.

Equipment will be made available to more students as the time and space constraints normally associated with the traditional laboratory can be reduced.

References:

[1] J. de Lafontaine, and C.A. Brunet (eds.), *Proceedings 2nd International Workshop on Tele-Education in Engineering Using Virtual Laboratories*, Sherbrooke, Canada, 2002.

[2] S. Dormido (ed.), Internet Based Control Education, *Proceedings of the IFAC Workshop on Internet Based Control Education*, 2001.

[3] K. Schilling, and H. Roth (eds.), *Telematics Applications in Automation and Robotics*, Pergamon Press, 2001.

[4] P. Backes, K Tso, J. Norris, G. Tharp, J. Slostad, R. Bonitz and K. All, Internet based operations for the Mars polar lander mission, in *Proceedings of the 2000 IEEE International Conference on Robotics & Automation*, 2000, pp. 2025-2032.

[5] G. Belmonte Bermudez and M.A. Perez Sanchez, Robots Tele-programming, in *1st Workshop on Robotics Education and Training*, RET 2001, Weingarten, Germany, 2001, pp. 19-24.

[6] R. Berntzen, J.O. Strandman, T. Fjeldly and M. Shur, Advanced Solutions for Performing Real Experiments over the Internet, in *Int. Conference on Engineering Education*, Oslo, 2001, pp. 21-26.

[7] H. Ewald and G.F. Page, Performing Experiments by Remote Control Using the Internet, in *Global J. of Engineering Education*, Vol. 4, No. 3, 2000.

[8] D. Popescu and K. Schilling, OM – A Remote Laboratory for Mobile Robots Education, in *Proc. 7th International Conference on Intelligent Engineering Systems*, Assiut, Egypt, 2003, pp. 447-453.

[9] S. Grange, T. Fong and C. Baur, Effective vehicle teleoperation on the world wide web, in *Proc. IEEE International Conference on Robotics & Automation*, 2000, pp. 2007-2012.

[10] K. Schilling, Virtual Laboratories for Mobile Robots, in *Proceedings IFAC Workshop on Internet Based Control Education*, Madrid, 2001, pp. 115-120.

[11] S. Amin, M. Zakaria, N. Majid, L. Siong, L. Horvath and J. Tar, Internet-based telerobotics: UTM's experince and future direction, in *Proc. The 10th International Conference on Advanced Robotics*, ICAR 2001, Budapest, Hungary, 2001, pp. 313-319.

[12] D. Popescu, I. Dinulescu, L. Popescu, Tele-programming of Programmable Logic Controllers via Internet, in *WSEAS Trans. on Advances in Engineering Education*, 2, Vol.2, 2005, pp. 118-126.

[13] I. Dinulescu, D. Popescu, G. Terejanu, Web based Telematics Application using Open-Source technologies, in *Proc. 12th Int. Symposium on Systems Theory, SINTES12*, Craiova, 2005, pp. 267-272.

[14] R. Marin and P. Sanz, The UJI Telerobotic Training System, in *1st Workshop on Robotics Education and Training*, RET 2001, Weingarten, Germany, 2001, pp. 25-30.

[15] D. Popescu, G. Terejanu, I. Dinulescu, M. Leoveanu, PLC based Telemanipulation via Internet, in *Proc. 14th Int. Workshop on Robotics in Alpe-Adria-Danube Region*, Bucharest, 2005, pp.441-447.

[16] D. Flanagan, *Java Script – The Definitive Guide*, O'Reilly, 2001.

[17] M. Hall, *Core Servlets and Java Server Pages*, Prentice Hall, 2000.

[18] B. Kurniawan, *Java for the Web with Servlets, JSP, and EJB: A Developer's Guide to J2EE Solutions*, New Riders Publishing, 2002.

[19] B. McLaughlin, Java and XML, O'Reilly, 2000.

[20] *Microsoft Visual Basic 6.0 – Programmer's Guide*, Ed. Microsoft Press.