Flexible virtual and remote laboratory for teaching Robotics

F. A. Candelas Herías*, C. A. Jara Bravo, F. Torres Medina

Automatics, Robotics & Computer Vision Group (AUROVA). University of Alicante.
Crta. San Vicente del Raspeig s/n, 03690, San Vicente del Raspeig, Alicante, Spain.

This paper describes the new features included in the Robolab 2 System, which permits the students to work with a simulation of and industrial robot and carry out operations with the real robot through teleoperation. Among these features, the flexibility of managing different robots or including new robot models and the very friendly user interface are highlighted. Some aspects about how Robolab is applied for teaching Robotics, and some experiences with this virtual laboratory are also described.

Keywords: robotics; virtual; remote; simulation; learning; graphics; Java.

1. Introduction

The teaching of practical subjects or professional courses in Robotics requires traditionally expensive equipment and many times is insufficient to be used by many students simultaneously, as for example robots and their controllers, in addition to the need for the students to go to such laboratories according to some strict schedules [1, 2]. It also is possible that this equipment can be damaged if it is used improperly. Nowadays, new technologies such as the Internet, the tele-operation of systems and the virtual reality, allow the student to carry out practical exercises, and to do a self-evaluation of the knowledge they have acquired [3], from other places (like at home) without the above-mentioned problems by using virtual laboratories [3-5].

There is a great amount of Java-based tele-operation applications thanks mainly to the portability of this language (http://java.sun.com). Many of the applications for Robotics are for simulating or tele-operating educational equipment or simple robots [4, 6-8]. In the field of industrial robot arms there are fewer applications, and generally they are designed for specific robots [4, 9]. Very few of them are based on an open architecture, which offers the required flexibility to change the robot being used or to add new robots without modifying either the user-interface or the architecture of the system [10, 11]. With regard to the simulation, there are not many Java-based applications for industrial robots that offer a realistic virtual environment, and the majority represent only wired-models or simplified structures.

This paper presents the main features of the second version of the Robolab system. This system is focused on the training of kinematics and the trajectory design for industrial robot arms. The main feature of Robolab 2, in contrast to other systems, is that it offers the flexibility of managing different robots or including new robot models and equipment, as well as other kinds of passive objects in the workspace, without changes in the user-interface and the system's architecture [10]. Robolab also has other interesting features, such as a feedback to the user based on updating a graphic simulation while the robot is moving, the use of high-level communication protocols, and the modelling and the virtual representation of the robots, passive objects and scenarios based on Java 3D (http://www.j3d.org) [11]. This system displays a more realistic simulation than the majority of the proposed Java-based systems for simulating robots, and the simulated robot can be very similar to the real robot. It also offers a user interface which is very user-friendly for operators who are not specialists in Robotics.

The paper is structured as follows. The following section describes the basic aspects of the architecture of Robolab 2. Afterwards, Section 3 explains how the authors apply the system for teaching Robotics, and describes some experiences with this virtual laboratory. Finally, some important conclusions and the related current work are shown in Section 4.

* Corresponding author: e-mail: Francisco.Candelas@ua.es

2. Robolab 2

This section explains the main aspects of the last version of the Robolab System, called Robolab 2. This version incorporates new features with regard to the previous version based in Java and VRML [5]. The two versions are available at http://www.disclab.ua.es/robolab/.

2.1 Equipment and architecture

With regard the equipment in the laboratory, the only pieces that require any considerable investment are the robots together with their controllers (Figure 1-a). The remaining pieces are commercial items. The “main server” is a PC that affords the Web services, supplies the Java applet with the user interface as well as the self-evaluation questionnaires. It also manages the user’s access and accounts. The “teleoperation servers” are other PCs that validate the commands to the robot received from a user’s computer, translate them to the appropriate language and send them to the robot’s controllers. They also obtain information about the current state of a robot to allow an on-line feed-back. In addition, there is a “video server” which gives the option of a video stream feed-back for remote operation [10].

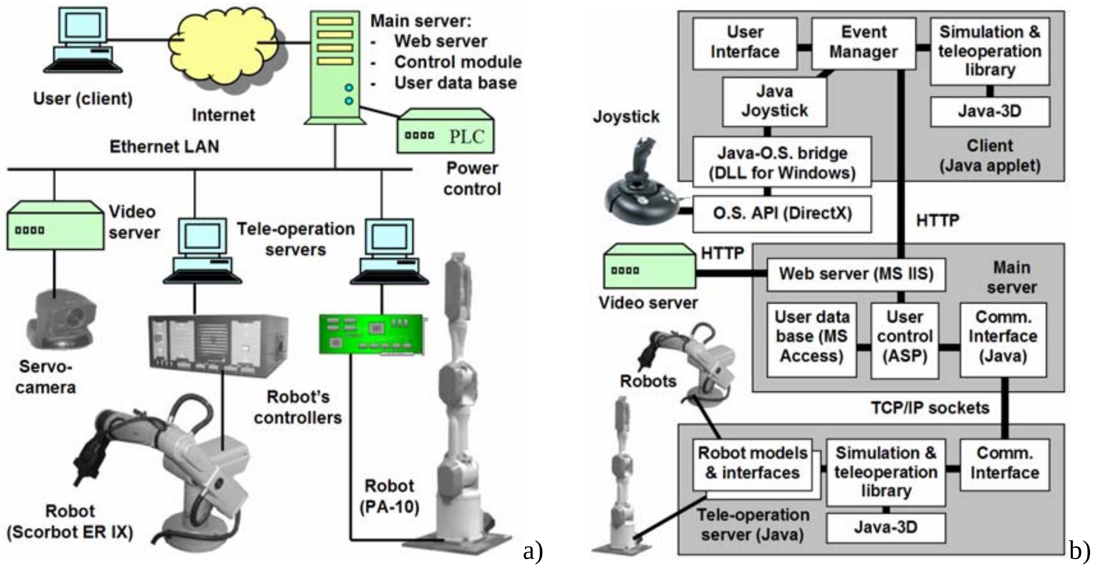

Fig. 1 Equipment (a) and software architecture (b) of Robolab 2 system.

The main achievement of Robolab 2 is the flexibility to define new robot models and include them in the client-applet and the teleoperation server (Figure 1-b). In this way, new robots of different characteristics can also be incorporated into the laboratory equipment to be teleoperated [10]. To model a robot, a library with a set of different Java classes has been created from the Java 3D API [11]. The main advantage of using this API, in contrast to others for Java, is that Java 3D is a standard extension of Sun’s Java. Since all the system is based on Java, the obtained software is very portable.

Other important feature of Robolab 2 is that the communication between a user interface applet and the web server is done with high-level protocols, which make possible the use of the teleoperation capabilities from any Internet access, over firewalls or other security systems configured in the networks [10].

2.2 User Interface

The only equipment that the student requires is a computer connected to the Internet, a Web client program and the Java and Java 3D runtime libraries. With these devices, the student can access to the “main server” and get the page in which includes the Java client-applet with the user interface. Figure 2 shows two views of this interface: the first one displays the simulation of a Mitsubitshi PA-10 robot, and the second one represents a Scorbot ER-IX robot (Intelitek). In addition to the robot, it is possible to define models for scenarios and passive objects in the simulation in run-time, from a local file or from an URL. The interface also displays the transformation matrix with the position and the orientation of the end tool, the values of the robot joints, and the list of tested movement commands entered by the user, in addition to other options as to modify the point of view of the graphic representation [10].

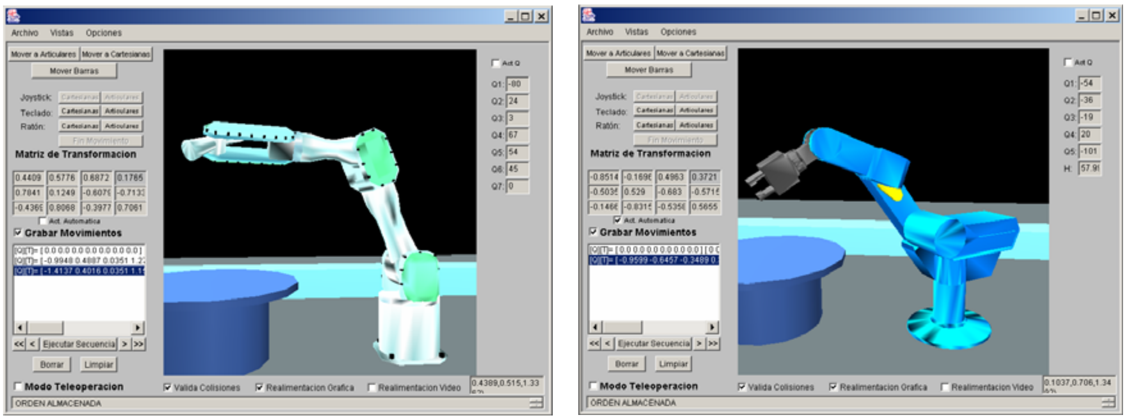

Fig. 2 Two views of the user interface of Robolab 2 with different robots.

The applet is able to use different devices to define a position or movement for the robot. Thus, it is possible to use either, a keyboard, a mouse or a commercial joystick for games. Furthermore, the user can specify the exact joint values, the transformation matrix that indicates the cartesian coordinates of the end tool and the duration of the movements (path planning).

Among the many devices that may be used as a user interface for controlling the remote robot arm in Tele-robotics, the force-feedback joystick is the most commonly employed, not only because it is a very intuitive way of specifying the movements, but also because it transfers the robot arm’s sense of touch to the human operator’s arm. However, force-feedback joysticks that are made specifically for Robotics are sophisticated and very costly devices. Taking this into consideration, we have improved the user interface applet of Robolab 2 with the capability of using a joystick for games in addition to the typical keyboard or mouse devices. There are many models which have simple force-feedback features. However, the majority of these joysticks are designed for specific operating systems and require specific APIs what contrasts with the main aim of Robolab: providing a portable and flexible user interface [12].

To access the joystick’s functions, we use the MS. DirectX API. The problem arises when a Java application has to use DirectX functions, since Java can not access this API directly. So, a “bridge” between DirectX and a Java library that manages the joystick has been developed. With this library, a commercial joystick for games can be used to move the simulated robot easily and directly (Figure 1-b).

Moreover, if a force-feedback joystick is used, the system can transmit the contacts or collisions of the robot-arm with its environment to the user, through a sensation of resistance. Robolab 2 can use two different sources of information to identify the collisions. The first one is the simulation engine in the client applet, which detects the collisions of the robot-arm with their environments (passive objects) or with itself. The second one is an optional force sensor which can be incorporated into the robot’s end tool to obtain the real force read, while manipulating objects, and transmit it to the user interface through the Internet. The client applet evaluates the data of the two sources to control the joystick’s movements.

Finally, the system permits two options for performing the feedback to the user during a teleoperation: an online video stream from the robot workspace or a graphical updating of the 3D simulation in the client-applet based on information about the robot’s current state received from a teleoperation server [5, 10]. The second option allows a fluid tele-operation with a lower bandwidth requirement the first one.

3. Using Robolab

3.1 Practical exercises

Robolab is used in several courses in the University of Alicante (Spain) by the authors of this work. In particular, it has been used since the 1999/2000 academic year in practice groups of the course "Robots and Sensorial Systems", which is an optional course in the degrees in Computer Engineering and Computer Systems. In the practical classes of this course, students have to achieve a set of exercises which combine computer vision algorithms used as sensorial techniques with Robotics [12, 13].

Robolab is used in several experiments. On the one hand, the simulation is used to study the components of different robots, introduce the coordinate systems and homogeneous transformations, and experiment the direct and inverse kinematics of a robot. On the other hand, after students have implemented the appropriate computer vision algorithms to locate an object in the workspace of the robot, Robolab is used to design and evaluate the kinematics control of a robot for catching the object. For this task, students have to apply point-to-point trajectories and a cubic interpolator [12].

The student first does the exercises on the simulated virtual environment and then, after checking that the results are correct, can execute their in the real system by means of the tele-operation option. Thanks to the simulation, the students are able to practice and carry out correct movement sequences. Once a correct simulation has been effected, the student can request the "main server" to remote execute the movement sequences with the real robot. Although the user interface is always available for executing simulations in the virtual environment, the student must identify himself as an authorized user to use the tele-operation capabilities.

3.2 Experiences in teaching

To evaluate the virtual laboratory's acceptability and its effect on learning, several statistical studies have been carried out during several academic years [12, 14].

The main result of the studies is that a great number of students have happily accepted the virtual laboratory and have used it to do the practice experiments remotely (30-40%, depending on the exercise) because the flexible time table, although many students have chosen the option of real laboratory at the university (60-70%) where they can work in coordination their class-mates and have the support of a teacher, although they have also used the virtual laboratory. Thus, it is very interesting to investigate web environments that offer not only virtual tools but also ones that allow the students to share their experiences and their results, while having a virtual teaching support [15]. The studies also have shown that the remote access to costly tools and resources like robots is positive and interesting, since it makes practice more attractive and real in comparison to a mere simulation.

4. Conclusions and current working-lines

The main feature included in the second version of Robolab is the flexibility in changing the robot model used in the simulation or in adding new robots to be tele-operated in the laboratory. The library of classes created for modelling robots, which is based on Java 3D, facilitates the specification of new robot models and their inclusion in the system. In addition, the user interface of Robolab 2 is very user-friendly, and the graphical simulation very realistic.

There is no doubt that a simple interface with a good simulation, like the one performed by Robolab 2, helps the students to save time in learning to use the tool and allows them to concentrate on the more important aspects of the course. In any case, it must be assured that the software required for the student's computers is accessible and easy to install and run. To do so, Java is a very good option.

However, it is also important to evaluate aspects such as the acceptability of such laboratories by the students and the effect they have on teaching, in contrast with the traditional teaching. The studies performed have showed that students considerer the virtual laboratory to be a valuable complement to the teacher and traditional teaching, but they also demand the support of the teacher and the collaboration with their class-mates. Nevertheless, the majority of the virtual laboratories are designed to be used individually, and they do not permit the work in group. Considering the previous conclusions, our research group is currently working in collaborative environments which incorporate shared virtual laboratories. This kind of virtual laboratory allows the teacher and the students of an on-line course through the Internet to use shared simulations in order to experiment practical concepts in a coordinated way [15].

Our research group continues developing the Robolab system, and we are working on including new features such as the 3D recognition of basic objects in the workspace and their inclusion in the simulation as 3D models, the inclusion the virtual objects in the 3D simulation and in the video feedback using augmented reality, and the ability of manipulating this objects with the robot. Also, a new version of the virtual laboratory is being developed using the EJS software (http://www.um.es/fem/Ejs/) [16].

References

[1] F. A. Candelas, J. Sánchez. Recursos Didácticos Basados en Internet para el Apoyo a la Enseñanza de Materias del Área de Ingeniería de Sistemas y Automática. Revista Iberoamericana de Automática e Informática Industrial (especial issue Aplicación de la TIC's a la Educación en Automática) **2,2**, 93 (2005).

[2] F. A. Candelas, S. T. Puente, F. Torres, V. Segarra, J. Navarrete. Flexible system for simulating and teleoperating robots through the Internet. Journal of Robotic Systems **22,3**, 157 (2005).

[3] F. A. Candelas, F. Torres, P. Gil, F. Ortiz, S. T. Puente, J. Pomares. Laboratorio Virtual remoto para Robótica y Evaluación de su Impacto en la Docencia. Revista Iberoamericana de Automática e Informática Industrial (RIAI) **1,2**, 49 (2004).

[4] F. A. Candelas, S. T. Puente, F. Torres, F. G. Ortiz, P. Gil, J. Pomares. A Virtual Laboratory for Teaching Robotics. International Journal of Engineering Education (especial issue Remote Access/Distance Learning Laboratories) **19,3**, 363 (2003).

[5] Ch. Salzmann, P. Saucy, D. Gillet, And F. Mondada. Sharing of Unique or Expensive Equipment for Research and Education. Informatik / Informatique, Magazine of the Swiss Informatics Societies, **4**, 32 (1999).

[6] Gillet, D., Latchman, H. A., Salzmann, Ch., Crisalle, O.D. Hands-On Laboratory Experiments in Flexible and Distance Learning. Journal of Engineering Education, **90**, 187 (2001).

[7] S. Dormido. The role of Interactivity in Control Learning. Proc. 6th IFAC Symposium on Advances in Control Education, Oulu, Finland, 2003, pp. 11-22.

[8] G. T. McKee, An online robot system for projects in robot intelligence, International Journal of Engineering Education - Especial Issue Remote Access/Distance Learning Laboratories, **19,3**, 356 (2003).

[9] R. Marin, P. J. Sanz, A. P. del Pobil. A predictive Interface Based on Virtual and Augmented Reality for task Specification in a Web Telerobotic System. Proc. IEE/RSJ Int. Conference on Intelligent Robots and Systems, Lausane, Switzerland, 2002, pp. 3005-3010.

[10] A. Aditya, B. Riyanto. Implementation of Java 3D Simulation for Internet Telerobotic System. Proc. IASTED International Conference on Modelling and Simulation (MS 2000), Pittsburgh, PA, USA, 2000.

[11] K. Goldberg, R. Siegwart. Beyond Webcams: An Introduction to Online Robots, The MIT Press, 2002.

[12] F. Torres, J. Pomares, P. Gil, S. T. Puente, R. Aracil. Robots y Sistemas Sensoriales. Prentice Hall, 2002.

[13] F. A. Candelas, F. Torres, S. Puente, J. Pomares, V. Segarra, J. Navarrete, A Flexible Java Class Library for Simulating and Teleoperating Robots. Proc. 11th IFAC Symposium on Information Control Problems in Manufacturing (INCOM 2004). Salvador de Bahía, Brasil, 5-7 Abril, 2004.

[14] J. Sánchez, F. Esquembre, C. Martín, S. Dormido, R. Dormido, R. Pastor. Easy Java Simulations: An Open-Source Tool to Develop Interactive Virtual Laboratories Using Matlab/SImulink. Int. Journal of Engineering Education, **21,5**, 798, 2005.

[15] F. A. Candelas, F. Torres, P. Gil, S. Puente, J. Pomares. Including the Virtual Laboratory Concept in an On-Line Collaborative Environment. Proc. Advances in Control Education 2006, Madrid, Spain, 21-23 June 2006.

[16] F. Torres, F. A. Candelas, S. T. Puente, J. Pomares, P. Gil, F. G. Ortiz. Experiences with Virtual Environment and Remote Laboratory for Teaching and Learning Robotics at the University of Alicante.International Journal of Engineering Education (Special Issue on Robotics Education), **22,4**, 766, 2006.