A multimedia environment for supporting the teaching of robotics systems

José C. Freire Jr.
Electrical Department
São Paulo State University
CP 205, 12516-410, Guaratinguetá, SP, Brazil
Jose-Celso.Freire@feg.unesp.br

José V. de Lima
Computer Science Department
Federal University of Rio Grande do Sul
CP 15064, 91501-970, Porto Alegre, RS, Brazil
valdeni@inf.ufrgs.br

Rodrigo A. Neves
Electrical Department
São Paulo State University
CP 205, 12516-410, Guaratinguetá, SP, Brazil
Rodrigo.Neves@ifrance.com

Galeno J. de Sena*
Mathematics Department
São Paulo State University
CP 205, 12516-410, Guaratinguetá, SP, Brazil
gsena@feg.unesp.br

Abstract

This paper aims at describing an educational system for teaching and learning robotic systems. Multimedia resources were used to construct a virtual laboratory where users are able to use functionalities of a virtual robotic arm, by moving and clicking the mouse without caring about the detailed internal robot operation. Moreover, through the multimedia system the user can interact with a real robot arm. The engineering students are the target public of the developed system. With its contents and interactive capabilities, it has been used as a support to the traditional face-to-face classes on the subject of robotics.. In the paper it is first introduced the metaphor of Virtual Laboratory used in the system. Next, it is described the Graphical and Multimedia Environment approach: an interactive graphic user interface with a 3D environment for simulation. Design and implementation issues of the real-time interactive multimedia learning system, which supports the W3C SMIL standard for presenting the real-time multimedia teaching material, are described. Finally, some preliminary conclusions and possible future works from this research are presented.

1. Introduction

The engineering teaching must obligatorily base the theoretical concepts presented in classroom using experimen-

---

* Visiting researcher at the National Institute of Multimedia Education, Japan, in the fiscal year of 2003.

tation and practical demonstration. As defined by Donald [3] "robotics is the science that seeks to forge an intelligent, computational connection between perception and action". We believe that through the robotics we can effectively base the theoretical concepts in a practical way.

However, the use of robots can become difficulty due to the high costs that they normally present. Opposing to that point, the recent hardware and software development made possible the outcome of diverse didactic robots under low costs [9, 13] what turned out the education and research in the area of didactic robots very popular. Besides, the robotic education was introduced in several engineering and computer science curriculum. One of the reasons for that is presented by Gallagher, Perretta and Drushel [5], who underline that "didactic robots related courses and materials enables the student to participate in a number of practical areas, including engineering, computer programming, artificial intelligence, project management and technical writing".

In this way, the main objective of this project is to develop a multimedia environment able to communicate and interact with a real robot arm for didactic use in laboratory activities of engineering students. It presents a multimedia system, developed with the Java language [20], which provides a dynamic environment for teaching and learning concepts of robotic systems and communication of a PC microcomputer with external devices. Such system, to have increased their qualities, should be used with an electro-mechanical robotic arm also developed by the authors and presented in [8]. Concisely, this electro-mechanical robotic arm (nowadays almost on plain function) is composed of

step motors and its respective drivers and sensors, assembled in a plate and connected to the PC parallel port. Through the parallel port, the system dynamically interacts with the robotic system under student control, following a preprogrammed content or by means of an application developed by the students. To do this, the system must activate devices like step motors, and read the status of sensors connected to the parallel port.

In developing the project, it was also of our concern the use of Educational Technologies [2] in Engineering Courses, considering the importance that investigations related to the use of new technologies in Education have nowadays. In the work [18], for instance, it was emphasized the importance of the development of Hypermedia tools to support the teaching activities in subjects of the Engineering. Other examples of tools related to the teaching of Engineering subjects were presented in [16, 17].

2. Multimedia environment

2.1. The graphical presentation

In this project we present a virtual robotic arm accessible through a computational system. A view of this arm on a computer screen, as a 2D object, could be inappropriate to the correct visualization of the arm itself as well as to the interactivity related to its use. That problem, related to the interface, is very common in simulation applications where real objects are represented on a virtual world. To solve this problem several 3D graphic tools have been developed and, although nowadays does not exist a unique tool able to solve all the problems, there are some classes of tools dedicated to solve this specific problem.

The work [1] mentions the following 3D graphic tools: ACAD, SolidWorks, VRML, OpenGL and Java 3D API. According to that work, the first two ones are mainly used in "high precision and dedicated modelling". The last three ones, cost-free on the web, are more oriented to graphical applications.

With respect to the cost-free tools, the work [1] enumerates several of their characteristics. For instance, VRML applications are independent of the platform. Usually plug-ins are required in the client's Web browser to run VRML applications. Two well-known plug-ins are Cosmoplayer (http://www.cosmosoftware.com/) and Blaxxun (http://www.blaxxun.com/). OpenGL corresponds to a C++ graphics library and, as so, an OpenGL application can be platform-dependent. As for Java 3D API, it corresponds to a set of classes that can be used in writing 3D graphical applications and 3D applets.

As it has been decided to use the Java language to develop the Multimedia System, the Java API 3D showed to be the natural solution to implement the virtual robotic arm.

Besides, as pointed out in [7], programs developed with Java 3D are executable on several platforms and over the web. Notwithstanding the fact that the interface provided is simple, "good applications and animations" can be developed using the library resources, and objects developed using other 3D modelling tools, like VRML, can be imported into Java 3D.

2.2. The development language

At present Java reached a distinguished place in the panorama of the programming languages, not only in the academic as also in the industrial environments. Even though it does not integrate the latest advances in the theory of programming languages, it constitutes a language able to generate robust and stable programs. Another advantage is related to its similarities to C++, which, before the appearing of Java, had been maybe the most used language in the development of object oriented applications.

In addition, Java eliminated several of the problems linked to C++, as for example the one related to the treatment of threads and runtime exceptions. Another important feature is that, differently from C++, Java is fully object-oriented and not hybrid like C++.

Another remarkable point refers to Memory Management. Java programs do not care about memory. The Java Virtual Machine (JVM) provides applications with all the memory they need. The JVM handles the nuances of obtaining more memory from the underlying host operating system. Java programmers do not even need to concern themselves with returning memory they no longer need to the system, since the JVM's automatic garbage collection mechanism automatically deallocate objects with no more references.

One of the main concerns in the development of this system was to minimize its implementation cost. Taking into account this concern and the excellent technical characteristics of the Java programming language, this language was chosen to develop the multimedia system. This choice led to the utilization of the API Java 3D for building the graphical simulation environment (the virtual robotic arm).

2.3. The developed multimedia system

As it can be noted in Figure 1 (a detail of Figure 2), the developed tool has 4 tabs that offer different utilization possibilities. The *Robot Arm Simulation Tab* permits access to the 3D arm simulator. The *Dynamic Presentation Tab* gives a presentation about the robotics developed under the SMIL technology (described below). In the *Static Presentation Tab* the user can find topics related to robotics in a static way. Finally the *Notepad Tab* is a simplified text ed-

itor that could be used as a notebook by the students during the use of the multimedia system.
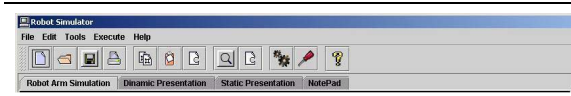

Figure 1. Functionalities of the System

2.3.1. Robot Arm Simulation From this tab the user has access to the virtual robotic arm developed with the Java 3D API [19]. In Figure 2 we present a snapshot with the Robot Arm Simulation, selected in the multimedia tool.
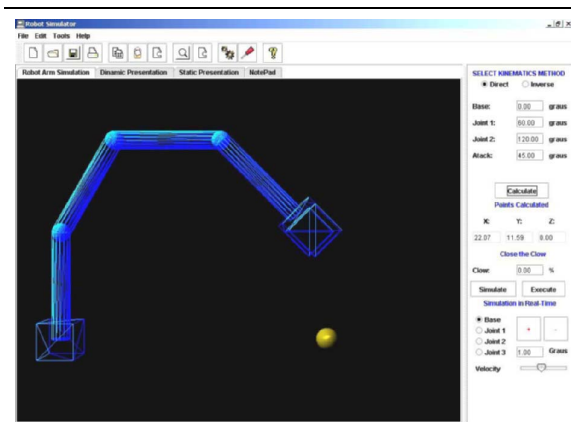

Figure 2. 3D Virtual Robotic Arm

In the multimedia tool we can move the robot arm using the Forward (angles to position) and Inverse Kinematics (position to angles) options. To do so, when the *Robot Arm Simulation Tab* is active, the tool presents a panel to the right of the window so that the user can supply the parameters that will define the movement of the robotic arm.

Figure 3 presents in details the panels through the ones one can specify the parameters to move the arm. The choice between direct kinematics and forward is made through the radio buttons on the upper part of the panel.

Through these panels the user initially supplies in the text fields the parameters that define the desired movement. Then, through the Calculate button, the user commands the system to calculate the final position of the virtual arm corresponding to the given parameters. Through the Simulate button the user then commands the virtual arm to move to the calculated position. Through the Execute button the user can command the real robotic arm, connected to the computer by the parallel port, to execute the same movement performed by the virtual arm (e.g.

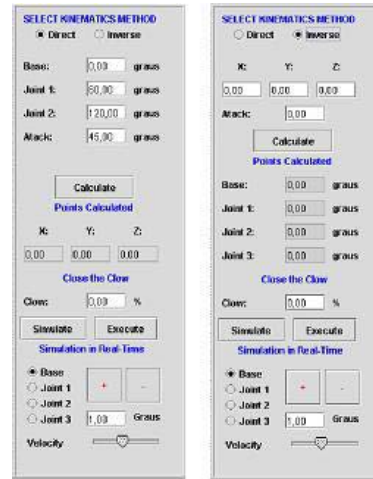section 2.3.4). The buttons on the bottom of the panel allow moving the virtual arm in discrete steps.

Figure 3. Forward and Inverse Kinematics

Thus the user can obtain a direct visual feeling about the behavior of the robotic arm and the relations between the different parts. Furthermore, to have assistance in building simulations, the user can access the "Static Presentation tab" where he or she can find a lot of information about robotic systems.

2.3.2. Robotics System Dynamic Presentation The multimedia term generally refers to data such as video, sound, image and text. The use of data of these types has been generalized, becoming even ubiquitous. And it is accomplished at present either through the use of patterns or languages, to guaranty the change and the portability of these applications, as through the integration of these specific data types into the current systems as for instance in the new versions of the Database Management Systems.

The World Wide Web Consortium (W3C) proposed to the integration of multimedia content in a presentation, the SMIL language, an acronym for Synchronized Multimedia Integration Language that, according to the W3C "enables simple authoring of interactive audiovisual presentations being used for "enables simple authoring of interactive audiovisual presentations being used for 'rich media'/multimedia presentations which integrates streaming audio and video with images, text or any other media type" [21]. A presentation written in SMIL is similar to a HTML document, being structured using Tags, what makes its learning ease.

In relation to the desired features for the Multimedia System, the SMIL language presents some important characteristics, as for instance the possibility "to play" a multimedia

presentation several times. In addition, as in the building of the presentation one can define synchronization points among the medias (with respect to a time line), and since these points can be reached from the hyperlinks defined in the presentation itself, one can reach any point of the presentation randomly, what allows that the presentation "be played" in any order.

So, the multimedia environment integrates a dynamic presentation (e.g. Figure 4) that shows lessons about robotic system topics created using the SMIL language. This presentation can be started in the 'Dynamic Presentation' tab. Actually this presentation is made by using the RealOne Player [12]. We are working in integrate a SMIL presentation player inside our Multimedia System using the X-Smiles API [22] developed at Helsinki University of Technology.
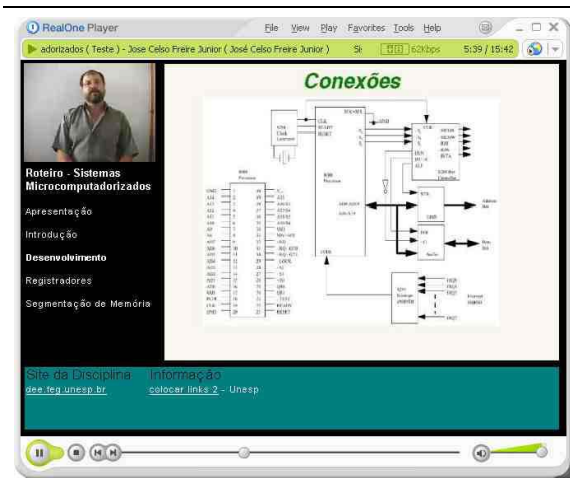

Figure 4. Smil Presentation on Robotics

2.3.3. Robotics System Static Presentation The multimedia environment also includes a set of HTML pages, namely the *Static Presentation Tab* that presents the robotic system topics, like in the SMIL presentation but in a static way. These set of pages are presented in an integrated browser that can be reached by the 'Static Presentation' tab. Besides the robotic system content, in these pages the students can access some practical exercises either to run in the 3D simulator or to activate the real arm.

2.3.4. Interaction with the real Robotic Arm The Multimedia environment will interact through the parallel port with a real robotic arm [8]. Dealing directly with hardware is always a delicate task, knowing the vast quantity of possible problems, which varies from simple software requisition conflicts to system memory violation.

To protect the system and increase its stability, Microsoft implemented in Windows 2000, a privilege level scheme inherited from Windows NT. This scheme has only two I/O privilege levels, level 0 and level 3. User mode programs will run in privilege level 3, while device drivers and the kernel will run in privilege level 0, commonly referred to as ring 0. As a result, the I/O can be touched only by the operation system and drivers, not any user being allowed to read or write it. Therefore, all user mode programs should talk to a device driver, which arbitrates access.

In our project to implement the 'multimedia environment/real arm' communication, the simulator module must activate the registers of the Parallel Port Interface (PPI) of the computer. Due to the problems presented above, we chose to implement this communication using a direct interaction with the firmware. Knowing that the objective was to send and extract data from PPI, our efforts were oriented in this line, not adding any other type of functionality on the driver. A windows library able to establish communication with the driver and return the useful parameters was developed, with a view to an union with Java applications (e.g. Figure 5). The library uses the driver capabilities to interact with the firmware, manipulating it to obtain the desired information.
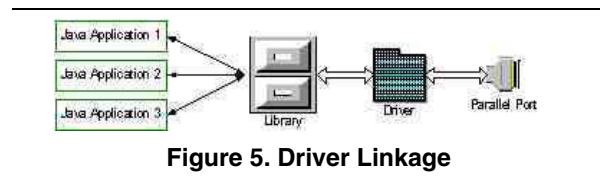

Figure 5. Driver Linkage

3. Discussion and Related Work

In this section we present some considerations about works similar to the one discussed in this article. Initially, we make some considerations about works related to multimedia systems and, in the sequence, about works on robotics and education.

Much work has been done in the domain of construction of synchronized didactic materials. Rousseau et al. have defined one generic solution for user adaptation of synchronized multimedia presentations [15]. The adaptation was considered as a transformation problem in which the user specifies a predicate that being applied to a generic multimedia presentation yields a customized view of presentation. Holtman et al. have defined extensions to HTTP to handle transparent content negotiation [6]. The extensions allow the client to specify which version of a given object a server should deliver. Fox et al. have developed a distillation proxy that degrades several media types (images,

video, postscript) to decrease the download time [4]. Our work aims at synchronizing the medias as didactic material, without worry with the media adaptable problem. However, we are studying these problems for a future work development. Thus, we have proposed here a virtual laboratory capable to show the synchronized operations with a robotic arm. Also, the presentation could be made by synchronizing videos, slides and operations with the robotic arm functions.

Several works approach the use of robotics in the engineering or computing science curriculum. Some works concentrate on demonstrating the interest and the importance of these topics in the curricula. Nagai [11] for example presents an approach developed and applied in a course on robotics at Ritsumeikan University, denominated "learning while doing". Rosenblatt and Closet [14] from Carnegie Mellon University present an experience developed in a course on robotics where the students learn to build themselves their own robots. Our work is still in its initial stage. Therefore, we intend to analyze experiences as the ones mentioned above in order to base the use of our environment as a tool in courses on robotics.

Several works also approach the development of web based robotics education. MaKee, of the University of Reading for instance, presents the development of an "on line environment supporting a network based robotics educational project" [10]. This environment allows that the students to control a "toy device" through the net. At the development of our work we intend to follow an approach similar to this one.

4. Conclusions and Future Plans

This work presents a multimedia system that can be used as didactic tool for teaching and learning robotics. The tool integrates in a unique environment a virtual robotic arm that can be operated through direct or forward kinematics, dynamic or static multimedia presentations that can be used to base the learning, and a notebook that can be used by the student to make notes about the accomplished study. Furthermore, one can use the multimedia system to control a real robotic arm connected to the parallel port of the computer, so that this arm perform the same movement performed by the virtual arm.

At the development, we chose to use an open architecture based on tools and technologies that can be obtained free from the Web. Besides the question of costs, this choice was made to ease a future development of the tool, through the use of plug-ins. The accomplished work confirms the viability of the development of a refined tool using only free technologies. Preliminary evaluation results from its use (the system has been used to support teaching activities of the Electrical Engineering Course at São Paulo State Univer-

sity, campus of Guaratinguetá) evidence its utility and the interest that such tool originates in the students, in this way being supportive to the teaching and learning process. Logically, a more elaborated evaluation instrument need to be developed and applied, so that one can have a better feedback concerning the impact of the use of the tool. Our intention also, connected to another research on visualization techniques in the teaching of Engineering and virtual reality, is to investigate the effects of such systems, in special those effects related to cognition, on their users.

The future development of the tool should be concentrated on the development of a web-based environment that allows a user to control, through the Web, a robotic arm connected to another computer. For that, the user should install our multimedia system as a client system, accomplish simulations in the client, and observe the behavior of the real arm connected to a server.

We believe that this web-based environment can be made true by two initiatives. The first one refers to the development of a video server that makes possible the capture, by a camera, of images of the real arm, and their transmission through the Web. The second functionality can be implemented to allow that, at the client, the user can save a series of movements of the virtual robotic arm and send this sequence to the server, to which the real arm will be connected, so that the server commands the accomplishment of the simulated movements. An initial study is being done to define the way through which this functionality will be implemented.

And we also intend to change the system into a development platform, which can be used in the development of other multimedia courses about engineering topics where the interaction with a simulator can be profitable. We imagine that it can be done by adding plugins to the future platform. And in this way we are beginning the development of a module about electric machines for future integration.

References

[1] K. E. Afshari and S. Payandeh. Toward implementation of Java/VRML environment for planning, training and teleoperation of robotic systems. In *Proceedings of the 5th Americas Conference on Information Systems - AMCIS '99*, pages 277–278, Milwaukee, Wisconsin, August 1999.

[2] J. I. Akamatsu and G. J. Sena. Tecnologias de aprendizagem para o ensino de engenharia in loco e à distância. In *Procedings of the XXVIII Congresso Brasileiro de Ensino de Engenharia*, Ouro Preto, MG, Brazil, 2000. in CD-ROM.

[3] B. R. Donald. On information invariants in robotics. In *Artificial Intelligence*, volume 72, pages 217–304. january 1995.

[4] A. Fox and A. Brewer. Reducing WWW latency and bandwidth requirements by real time destiollation. In *Proc. 5th International WWW Conference*, 1996.

[5] J. C. Gallagher, S. Perretta, and R. F. Drushel. *Teaching Autonomous Robotics over the World Wide Web: an Online Computer Engineering Practicum*. Interactive Multimedia Electronic Journal of Computer-Enhanced Learning, http://imej.wfu.edu/articles/2002/2/03/index.asp. Access in 04/11/2003.

[6] K. Holtman and A. Mutz. Transparent content negotiation. In IETF RFC 2295, ftp://ftp.isi.edu/in-notes/rfc2295.txt.

[7] G. Hopkins. *Teaching Autonomous Robotics over the World Wide Web: an Online Computer Engineering Practicum*. Resplendent Technology Ltd., http://www.java3d.org/tutorial/.

[8] R. B. Kina. Desenvolvimento de um braço robótico para utilização didática. Technical report, UNESP/FEG/DEE, 2003.

[9] Lynxmotion - Lynx 5 Robotic Arm Kit. http://www.lynxmotion.com/.

[10] G. T. McKee. The development of internet based laboratory environments for teaching robotics and artificial intelligence. In *Proceedings of the IEEE International Conference on Robotics & Automation*, pages 2696–2700, Washington DC, 2002.

[11] K. Nagai. Learning while doing - practical robotics education. In *Robotics & Automation Magazine*, pages 39–43. IEEE, June 2001.

[12] RealNetworks. RealOne Player. http://www.real.com/realone/index.html.

[13] Robix RCS-6: Robot Construction Set. http://www.robix.com/.

[14] M. Rosenblatt and H. Choset. Designing and implementing hands-on robotics labs. In *Intelligent Systems*, pages 32–39. IEEE, November/December 2000.

[15] F. Rousseau, J. A. Garca-Macas, J. V. Lima, and A. Duda. User adaptable multimedia presentations for WWW. In *Proc. 8th International WWW Conference*, 1999.

[16] G. J. Sena and M. N. Alves. An induction heating software to simulate special heat treatments. In *Proceedings of the International Seminar on Heating by Internal Sources*, pages 215–218, Padua, Italy, 2001.

[17] G. J. Sena, R. M. Bittencourt, O. P. B. Teixeira, A. D. Velasco, M. A. R. F. Gonalves, and J. I. Akamatsu. Desenvolvimento de um software multimídia para o ensino de tópicos de eletrônica básica. In *Proceedings of the XXVII Congresso Brasileiro de Ensino de Engenharia*, pages 318–325, Natal, RN, Brazil, 1999.

[18] G. J. Sena, R. A. C. Silva, O. C. Winter, F. T. Mitsuyassu, E. L. Moura, and M. A. F. Barbosa. Hypermedia-based tools: Development and usage to support teaching activities. In *Proceedings of International Conference on Engineering and Computer Education*, pages 330–334, São Paulo, SP, Brazil, 1999.

[19] Sun Microsystems, Inc. Java 3D API. http://java.sun.com/products/java-media/3D/.

[20] Sun Microsystems, Inc. The Source for Java Technology. http://java.sun.com/.

[21] World Wide Web Consortium (W3C). *Synchronized Multimedia Integration Language (SMIL 2.0) Specification'- W3C Working Draft March 2001*. http://www.w3.org/TR/smil20/.

[22] X-Smiles. http://www.xsmiles.org/index.html. An Open XML Browser for Exotic Devices.

COMPUTER SOCIETY