VIRTUAL CENTER FOR ROBOT STUDIES

eng. Ciprian Comsa, eng. Robert Mitrica, eng. Camelia Cojocaru, Drd eng. Gabriel Vladut
– IPA CIFATT Craiova, Romania, office@ipacv.ro
Prof Dr. eng. Oleg Cernian, Prof. Dr. eng. Mircea Nitulescu – University of Craiova
Prof. Dr. eng. Axel Hunger, Diplomat eng. Christian Schuetz
- University of Duisburg-Essen, Germany,
Prof. Dr. eng. Timothy Hall - University of Limerick, UK
Prof. Dr. Eng. Alexandru Soceanu - University of Applied Science Regensburg (FHR), Germany
Dr. Ileana Hamburg - Institut Arbeit und Technik Gelsenkirchen, Germany

Abstract

In the technological field, Virtual Reality (VR) technology has been widely proposed and recognized as a major technological advance for supporting life-long education to individuals along with a flexible workforce.

One of the unique capabilities of the VR technology is the successful translation of abstract concepts into visualized events and the interaction of students with them, that in real life could be limited due to distance, time, and safety factors.

Introduction

The wide expansion of the World Wide Web and the Internet has formed all the necessary preconditions for adopting this powerful means for purposes such as delivery of e-learning content, collaboration, and distance learning both in the industrial and the educational field.

E-learning can also be very useful in the case of a potential public for learning that could be described as "itinerant or that is widely scattered over a vast area. E-learning may be a way of overcoming geographical isolation and physical distances from resources or learning centres. In these situations, the use of the Internet and networking takes on its full meaning.

Robotics is well suited as a problem domain because it intrinsically requires multidisciplinary knowledge. Robotics encompasses subjects such as mechanical engineering, electronics, control, communication, vision, real-time parallel computing and systems design. By choosing a problem domain with a high multi-disciplinary requirement, we reduce the chance that any one person on the team will ‘cheat’ and do all the work. As the literature supporting the positive educational effects of robotics accumulates, it becomes increasingly clear that:

- robotics is very absorbing and enjoyable for many children and adults
- robotics events and competitions motivate the study of technical subjects
- robotics provides hands on examples for teaching science, technology, engineering
- robotics creates excellent possibilities for learning teamworking, especially through competitions.

Cumulative experience suggests that robotics is an excellent domain for introducing students to a range of technological and scientific disciplines. Robotics provides leverage on two key pedagogic aims of e-learning: problem-based learning and engaging students in discourse (i.e. creating a community of learning).

Robotics possesses a number of advantages that make it particularly well suited to unsupervised project work at a distance.

Distance learners often have little opportunity for face-to-face contact, although some courses offer up to two hour-long monthly tutorials or one-off day schools.

However, attendance at such tutorials is not compulsory and there are many reasons why distance education students may be unable to attend tutorial meetings.

Motivations and design

This section presents the main motivation factors that inspired the design and the implementation of the Virtual Robot Laboratory. The goal that should be achieved by this work is threefold:

1. To provide an e-learning environment for students as well as to overcome communication difficulties among them.
2. To overcome limitations of current e-learning applications.
3. To avoid limitations met in most Virtual Robot applications so far.

At a first level we plan to support the community of Virtual Laboratory users as follows:

- To overcome the problem of interaction and communication among them.
- To support the students to construct their knowledge in a "learning by doing" situation.
- To provide remote support to students by mentors.

System implementation and architecture

This section is dedicated to describing the structure of the system that supports the Virtual Robot. The term structure is used for referring to the logical view of the static structure of the architecture in terms of its components, their interconnections, and the interfaces and operations offered by these components.

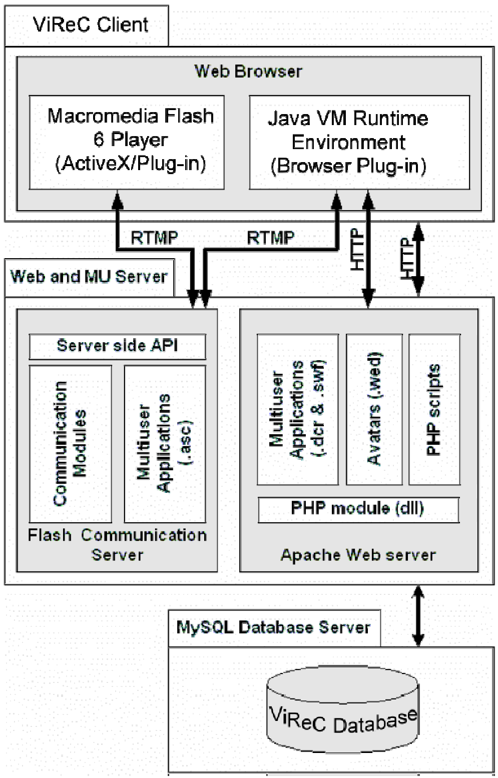
The step that follows the definition of the main design principles of the Virtual Robot is the assessment of the technologies that will facilitate the implementation of the system. system should be an easily accessible web-based system, used by a worldwide community, taking into account bandwidth, and client-side system constraints. Therefore one of the major targets apart from implementing the Virtual Robot with the appropriate functionality is to satisfy the following prerequisites.

First, there is a need to minimize the client-side system requirements and the cost of client-side system set-up. This means that the end-user should be able to access the laboratory using a typical personal computer (PC) without excessive requirements either in hardware or in software. Second,the system should have the ability to support a maximum number of simultaneous users.

This requirement should in particular be considered in the multi-user applications. Finally, using technologies, which do not require excessive hardware requirements, should also minimize the cost of the server side set-up.

As far as it concerns the web server, it is used for storing the client-side files of the Virtual Robot, voice chat and text chat. Furthermore, the web server stores and executes the PHP scripts to obtain the users' data from a database. The web server that satisfies these prerequisites is the Apache web server, which is free of charge, runs on almost all operating systems; supports PHP scripts; can host Director and Flash movies; and can interoperate with MySQL and Macromedia Flash Communication Server MX and java.

The structure of Virtual Robot is:



TextBooks

In this section we presented information about the robot programming language, communication protocols of the robot's controller.

The information in this chapter is enough in order to offer support to a user with medium skills in this domain. It is necessary for the user, before using the compiler, to enrich his knowledge by reading about how to make a module, a routine and how to combine them in a program, how to read and analyze the programming instructions and the examples presented in this chapter.

Because the use and the programming is limited by the space of operation the user has to know more information about the coordinates of movement, about the position of the table on which the pneumatic manipulators are placed facing the robot, information about the pieces type and their position on the table, information about the buffers where the raw pieces are gathered and the way the pneumatic manipulators act.

We also presented in this chapter the most useful instructions of programming together with the examples of program: instructions of moving, instructions of activating and deactivating sensors, logical, arithmetic, assigning, conditioning, looping and selection instructions.

“RAPID Summary”

This chapter briefly describes all instructions, functions and data types grouped in accordance with the instruction pick-lists using when programming. It also includes a summary of the syntax, which is particularly useful when programming off-line. **RAPID Summary** content:

“Communication Interface”

This chapter describes equipment, which the robots can communicate with computer, or other equipment via RS232/RS422 serial channels or via Ethernet.

“Communication protocol”

This chapter shows the protocols use in the S4 Controller. All protocols except the Robot Application Protocol (RAP) are standard protocols: SLIP, Ethernet, IP, ICMP, TCP, UDP, RPC/XDR, TFTP, Bootp, RAP, NFS, FTP.

-**“Instruction Set”**

Present instruction details for programming ABB robots

-**“Operator interface”**

This chapter describes how to programming a robots using Teach Pendant and offered possibility to test a Virtual Teach Pendant.

- “Sample”

This chapter describes sample with robots programming including useful instruction for a better understanding of working laboratories. This chapter contend sample at Motion application, cylindrical pieces palletizing, Paralelipedicall pieces palletizing, assembling pieces, Welding operation. This textbox content program source and simulated and interactivity application for a better understanding.

Tutorials

Simulate and interactivity applications

This section presents simulated and interactive applications with basic operations: palletizing of two types of pieces (a parallelepipedical piece and a cylindrical piece), the assembling of these two types of pieces and the welding-line and welding-point operations.in an attractive and explicit form for the reader.

The user can analyze the instructions in the program during its execution and the way of activating and deactivating the sensors used in the run process. The user can stop the execution and even to run step by step the execution in order to understand how the program is created.

In these tutorials we can also find interactive applications with the user where he can choose the type of the piece that needs to be transformed and the place on the pallet where the piece is going to be placed. He can also analyze the instructions in the program and the activating and deactivating the sensors during its execution.

At last, this section presents viewing tutorials of the axes of the robot in a 3d graphic, a tutorial dealing with the buttons of the Teach Pendant of programming of the robot in an attractive and explicit form and an interactive presentation of the user with the joystick on the Teach Pendant.

In order to understand the aim and the steps in lab assignments, the user must read carefully the sections: text books and all tutorials.

To offer useful information this section presents a film with the execution in the real lab and with information about the application components.

We can also learn about how to integrate a module in a program and about the set of useful instructions in an explicit presentation with sound, images and examples of execution of linear and circular movement instructions in a 3d form and the assignment of values in a variable.

The lab assignments are presented explicitly with sound and images for the user to understand the theme.

The lab assignments are:

-Movement operation

(by moving the tool of robot from Home position to a specified point)

-Output signals

(Set and reset output signals using RAPID instructions)

-Simple palletizing operation

(by grabbing the piece and move to a specified position)

-Palletizing operation

(by grabing the piece and move to a specified position)

-the assembling operations (by assembling these two types of pieces we will obtain a new product. Flowing from the parallelepipedical pieces is supplied by the first two workstations and flowing of the cylindrical pieces from the last two workstations.

All the pieces must be placed in specific position known by the robot. This two types of pieces are taken one by one by the robot and placed into the assembling place. Because of position's errors it's necessary for the robot to do two movements to arrange all the pieces in assembling place. After that, the robot puts the new product in a matricial storehouse with four locations.);

-the palletizing operation of parallelepipedical pieces (All the pieces must be placed in a matricial storehouse with four locations.);

-palletizing cylindrical pieces (Flowing of the cylindrical pieces is supplied by the workstation number three and workstation number four. All the pieces must be placed in a matricial storehouse with eight locations.);

- the welding operation (To do this operation we use storehouses (STATION) 1 and 2 which are supplied with parallelepipedical pieces. The robot take these pieces one by one and puts them together in the assembling place (AP). After that, with a welding device that is taken from AS (support for the welding device)the robot simulates welding-line and welding-point operations.

When the operation is finished the robot puts in support (AS) the welding device and the piece in the matricial storehouse with four locations in order 1, 2, 3, 4.).

There are also presented programming examples of the applications of palletizing, assembling and welding. They differ from the lab assignments by the positioning of the pieces in another order.

Robot movies

In this section, user can see movies with real application with: Cylindrical Palletizing, Paralelipipedical Palletizing, Welding, Assembling)

Exercises

The user opens this window in the same time with the compiler and can edit a program.

Once the program edited the user opens the option "run" of the compiler to see if the program has errors or not. If errors are observed in the program, the compiler shows in a bottom window the number of the line in the program where the error is and helping explanations to make it right. The running of the program can be made slowly or step by step.

The user can stop the execution and can pass to the next line or before the initial line, or can continue the execution from the initial line, if wanted.

During the running of the program the user can visualize the stocked values in registers**.**

It can be monitoring the activating and deactivating of some inputs and outputs pressing the number of the output that needs to be visualized.

After the program being checked for errors, the execution can be seen by pressing the button View Result. After pushing this button a window is opened in which the execution is visualized in 2d.

The execution can be visualize from any angle by a simple rotating of the mouse.

Robot's Workplace

Robot workplace is intended to provide a simulation tool in which the user can preview his written programs. After successfully loading the operating environment and the program to execute, just pushing the RUN button will start the program simulation, current instruction being shown on screen.

For greater customization, the user can define it's own environment space, containing various objects defined inside the editor.

Those objects can have different properties, like a piston that makes a push operation, or a parallelepiped piece with/without holes.

Technical Description

-DataBase

MySQL is an open source relational database management system (RDBMS) that uses Structured Query Language (SQL), the most popular language for adding, accessing, and processing data in a database.

-Compiler

The compiler is developed using java technology.

An applet is a small Internet-based program written in Java, a programming language for the Web, which can be downloaded by any computer. The applet is also able to run in HTML. The applet is usually embedded in an HTML page on a Web site and can be executed from within a browser.

Each user can save his work in database with public and private access.

Software offer possibility to load components in environment to develop more applications.

Robot Workplace short technical description

The code behind simulation is made as follows:

- initialization code;
- run-time code;
- clean-up code;

In the initialization phase, there are two different operations. First, the corresponding operating environment is read from the robot database. This is done through a service based on Apache Web server and PHP scripting language with MySql Support. Below is shown how the operating environment is read using a sample script (getenv.php).

In this way the objects from the environment are passed to the simulator, using the requested environment '$env' defined by user '$user'.

Then, the simulator decodes object data and interprets it for use.

This is done in the _root.onLoad function, which is called when the simulator loads.

The program is read in a similar way, with according modifications.

When the user presses the run button, the simulator will take the first instruction and execute it. Depending on the instruction mnemonic, the simulator runs the specific function in order to complete the program. Considering a "movel" instruction, the simulator will call the movel function.

When the function ends its operation, the simulator extracts the next instruction and fetches it, and so on. After the last instruction is executed, the simulator stops and waits for another simulation to be run.

Conclusions

As the Internet is turning into a truly multi-service network with a steady increase in bandwidth and decrease in response time, the environment becomes more suitable for implementations such as Virtual Laboratory.

Furthermore, it offers various communication channels such as gestures, voice, and text chat that help learners to interact and cooperate with each other.

References

Hamburg, Ileana, 2003: Verteilt und doch gemeinsam lernen. In: Wissenschaftszentrum Nordrhein-Westfalen: Das Magazin 14, H. 1, S. 34

Hamburg, Ileana / Cernian, Oleg / Thij, Herbert ten, 2003: Blended learning and distributed learning environments. In: 5th International Conference on New Educational Environments: Lucerne, Switzerland - may 26-28, 2003; the know-how hub for blended learning. Berne: net4net, p. 197-202

Hamburg, Ileana / Cernian, Oleg / Thij, Herbert ten, 2003: Lernen und Kooperieren in verteilten Umgebungen: die Chance für die betriebliche Weiterbildung! In: Engert, Steffi / Hamburg, Ileana (Hrsg.): IT-basierte Lernformen für die betriebliche Weiterbildung. Gelsenkirchen: Inst. Arbeit und Technik, S. 45-55

JC Waller, N Foster: Training via the web: a virtual instrument, Computers and Education, 35, March 2000, 161-167

news.bbc.co.uk 2001: BBC News Article – Virtual Lab brings Science To Life, http://news.bbc.co.uk/1/hi/sci/tech/1111654.stm

physicsweb.org: Virtual Interaktive Experiments,

http://physicsweb.org/resources/Education/Interactive_experiments/

www.sci.brooklyn.cuny.edu/-marciano Virtual Multi-Media Internet Laboratories,