

A Distributed Architecture for Collaborative Teleoperation using Virtual Reality and Web Platforms

Christophe Domingues, Samir Otmame, Frederic Davesne, Malik Mallem and Laredj Benchikh
IBISC Laboratory – CNRS FRE 3190 – University of Evry, France
{firstname.lastname}@ibisc.univ-evry.fr

Abstract — Augmented Reality (AR) can provide to a Human Operator (HO) a real help to achieve complex tasks, such as remote control of robots and cooperative teleassistance. Using appropriate augmentations, the HO can interact faster, safer and easier with the remote real world. In this paper, we present an extension of an existing distributed software and network architecture for collaborative teleoperation based on networked human-scaled mixed reality and mobile platform. The first teleoperation system was composed by a VR application and a Web application. However the 2 systems cannot be used together and it is impossible to control a distant robot simultaneously. Our goal is to update the teleoperation system to permit a heterogeneous collaborative teleoperation between the 2 platforms. An important feature of this interface is based on the use of different Virtual Reality platforms and different Mobile platforms to control one or many robots. The first aim with this work is to develop collaborative robot teaching with the use of virtual or real robot.

Keywords — Collaboration, Virtual Reality, Augmented Reality, Robot Teaching, Distributed Architecture, Teleoperation.

I. INTRODUCTION

Early Computer Assisted Teleoperation architectures provided to the human operator many kinds of assistances [1][2]. However diverse assistance strategies and information's concerning tasks are proposed and the robot commands where a function of operator actions and the adopted strategy (e.g Freezing some robot Degrees of freedom). The problem with this approach is that the current solutions are used in closed platforms and do not allow changes of functionalities [3].

Virtual Reality (VR) and Augmented Reality can benefit from teleoperation systems to improve human robot interaction. VR/AR has been used to solve time delay problems using predictive display concept where a virtual robot is surrounding the real robot on the video feedback [4][5][6][7].

In our laboratory, we have worked several years on human machine interaction model for teleoperation systems that led to the ARITI system. ARITI permits to control a 6 DoF robot with the use of Virtual Reality and

Augmented Reality. Two teleoperation platforms exist. The first using a Virtual Reality/Augmented Reality semi immersive platform [8][9] and the second is an online version accessible through a Web browser [6].

Recently, collaboration functionalities have been integrated on the VR [9] platform and Web platform [10]. However, technical problems and utilization limitations appears that are described on this paper. The major problem is that the 2 platforms cannot be used at the same time. Consequently, collaboration is not possible between the 2 platforms. Collaborative teleoperation has been motivated by several situations: physical limitation of manipulator's workspace or grasping large objects. However as pointed in [11], one of the current challenges in robotics is the joint execution of tasks involving human beings and robots at the same workspace, also called telepresence. With this work, our final goal is to permit collaborative teleoperation (users-robots) with multiple interfaces and multiples devices. So we want to make possible collaboration between VR/AR users, Mobile users and distant robots in a robot teaching application.

In the second section of the paper, the previous work of ARITI interface is presented with the different kinds of assistance. The third section is dedicated to the software and network architecture. A collaborative robot teaching application using this new architecture is presented in the fourth part.

II. PREVIOUS WORK

The ARITI project or Augmented Reality Interface for Teleoperation via Internet has started several years ago. In fact, it is a Client/Server application, which allows visualizing and controlling a 6 DoF robot (Fanuc Lr Mate, see table 1 for characteristics) by using any remote computer. It is aimed at enhancing HO capabilities for achieving complex telerobotic operations.

In this part, we present the ARITI teleoperation system with its problems and limitations, where answers trying to solve the problems have permitted us to develop the work presented in this paper.

TABLE 1: MAIN CHARACTERISTICS OF THE FANUC LR MATE 200

Dof : 6	J1: $\pm 320^\circ$	J4: $\pm 380^\circ$
Payload: 3 KG	J2: $\pm 185^\circ$	J5: $\pm 240^\circ$
Repeatability: ± 0.04 mm	J3: $\pm 365^\circ$	J6: $\pm 720^\circ$

In order to help the HO to control the robot and achieve a complex teleoperation task, an interactive assistance is given to him, which is a Virtual Fixtures (VF) intervention in the operation area. These VF appear and disappear as the robot's peg comes closer or away from the objects to be manipulated [12]. This type of virtual fixtures appears according to the kind of tasks that had already been achieved. Figure 1 presents ARITI versions' history. The first application created, is ARITI Web mono user. After that, we wanted to integrate collaborative functions on the application and in parallel we wanted to develop the VR platform to improve user experience using 3D vision and natural interaction.

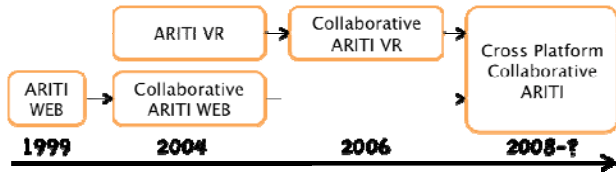


Fig. 1: The ARITI versions history and the final goal of our work.

A. ARITI Web

ARITI Web is based on a Java Applet where the user interface is divided in 4 parts. First, the augmented video feedback of the distance robot with the virtual robot superposed on. Second, there are 2 virtual views of the virtual robot. Finally, the last part presents the way to control the robot by clicking adequate buttons. At the beginning, collaboration was not possible therefore, the application has been updated to integrate these concepts where users form a group and each user is affected to a mission that they perform independently.

B. ARITI on VR Platform

1) Presentation of the VR Platform

The VR platform has been built on 2 conditions: Free and natural manipulation. A big screen with active stereoscopic capabilities composes the VR platform. Many interaction devices are present as 5DT Data Gloves to retrieve hand configuration, an ART optical tracking (Flystick, Hand/Head Tracking) and a SPIDAR to produce a haptic feedback. In figure 2, an operator is manipulating a distant robot with a Flystick and wears active stereo glasses to get 3D view.

2) ARITI VR

At the beginning, ARITI VR was used with a 6 DoF with only mono user support, where the collaborative functions have been integrated later. We have tested a collaborative function with another VR platform [9]. A Web interface was used to display the videoconference of the 2 distant operators while the second platform used the SPIDAR as an interaction device to control a virtual robot. The first platform used a Flystick as interaction device to control the virtual robot but this virtual robot was superposed on the real robot by using a video feedback.



Fig. 2: The operator controls the virtual phantom robot to take virtual objects. After the validation, the distant real robot will follow the phantom and take the real objects

C. Limitations and Problems

The major limitation of the actual system is that the 2 versions are not compatible together. To use the Web version, we must stop the VR Platform. Our goal is to have a global teleoperation system that permits to teleoperate a robot with the desired platform. The video stream of the distant robot can only be diffused for one user only because the protocol used to diffuse the stream is a homemade protocol. Consequently, when the second user connects to the system, he can only view the virtual robot. The VR platform application is realized in C++ with the use of OpenGL library. Moreover, the interaction part is limited to the devices available on our VR platform (ART tracking, SPIDAR). The Web application is realized in JAVA in an applet format. Collaboration is available but users must form a group and selected pre defined missions, as they cannot free control the robot together. Table 1 (columns 1 and 2) summarizes many limitations and problems encountered with the actual system.

III. THE NEW TELEOPERATION SYSTEM

The proposed teleoperation system tries to remedy the limitations and problems. In this part we present the proposed corrections and the technical solutions. Our intention is to support multiple interfaces and multiple devices.

A. Solving Problems and Limitations

We have integrated the Virtools solution to develop VR applications discussed in the previous section. Virtools is an application to easily develop VR applications for commercial or educational activities. Virtools natively integrates VRPN, an open source solution to use a multiple VR devices. The first version of ARITI Web uses Java Standard Applet. This solution needs the Virtual Machine installed on the client. Using the J2EE based on a Tomcat server permits to deport and execute the code on the server and not in the client. However, we could choose PHP or ASP as the portability is easier (J2SE -> JEE) and JAVA offering more possibilities due to the existence of JAVA 3D JAVA VLC (JVLC) libraries.

TABLE 2: SUMMARY OF SOME LIMITATIONS AND PROBLEMS AND THEIR CORRECTIONS

Problems and Limitations	What we want	Proposed Corrections
One camera is recording the distant robot Only one robot can be teleopered in the same time Video stream is only diffused for a maximum of one user Robot command are sequentially applied according to their reception by the system The Java Applet version is not compatible with mobile phones	Multi view of one or more robots Control many real robots Diffusing the stream for all connected users Combination of commands from the different users A teleoperation system which can be used on any kinds of devices Delete this limitation	Multi cameras streaming solution needed Changing the actual server Using Multicast or Unicast streaming solutions Integration of new command laws (e.g. combine commands from different users) Using JEE solution instead to have HTML in output Centralizing data as virtual shared objects into a server connected to a database
Impossible to teleoperate the robot with the Web and VR version at the same time		

B. The new Network Architecture

To remedy the limitation of the non-use of the 2 platforms together, we have added the multi user server. This server permits to link VR and Mobile applications. Figure 3 shows the network architecture with the integration of the multi user server. This server is connected to a MySQL database storing position and orientation of all shareable objects but also the position and orientation of the users' virtual robots. The multi user server is connected to the robot server to transmit the robot commands from the users.

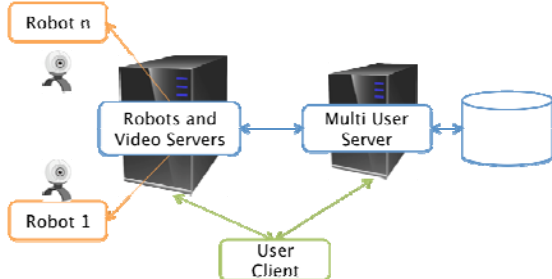


Fig. 3: The network architecture for the new ARITI teleoperation system. The user can indifferent used a Web or VR application and can do collaborative teleoperation with others Web or VR users.

For the Video component, we have used Windows Media Services based on Windows Server 2003 Enterprise Edition on the relay server. The relay server receives streams from different servers coder and diffuses (Multicast or Unicast) to the connected clients. The

encoder servers used Windows Media Coder 9. This solution has been chosen instead of ffmpeg streaming solution because streaming with ffmpeg does not work correctly with Virtools. Figure 4 shows the proposed streaming architecture. The media coder can encode video and audio from different sources only the sources are DxShow compatible (e.g. Webcams or Firewire cameras). We have used two types of cameras: Logitech Quickcam Pro 4000 and PixelLink Firewire cameras and have performed many tests (latency, network charge) to select the best compression format.

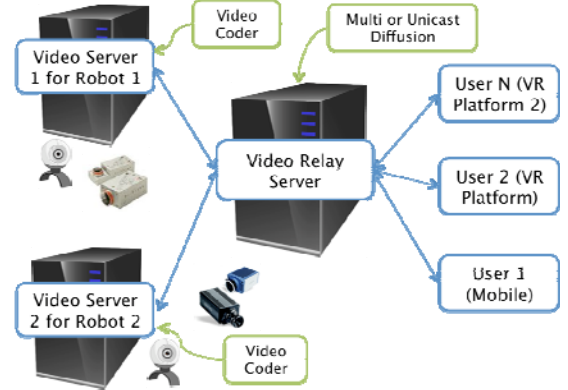


Fig. 4: Zoom on the video/audio streaming network architecture

C. The new Software Architecture

With this work, we wanted to integrate on the software architecture a system that will permit the personalization of the application. To achieve this objective, we have considered all the functionalities as modules where each module is designed with the same conventions and is available in 2 versions (Mobile and classic). A module has 2 ways of working (Safe or classic mode) and communicates with the core. There are 2 cores version, the mobile and the VR Core. The VR core is implemented in Virtools scripting language while the second is implemented in JAVA. Despite of the existence of 2 core versions, each core has the same structure, functions, inputs and outputs. The core sends to a module synchronization signals as "LOAD" or "UNLOAD". These signals are used to load the module or unload the module when the application is running. Figure 5 illustrates the communication between the core and different modules.



Fig. 5: The communication between the VR/Mobile core and the modules. The cores have the same architecture just modules are different.

At the beginning, modules are loaded by the module loader (component of the core). When the load is complete, the module gives a state report to the core that permits to determine the working mode (Safe or classic). A safe mode permits to deactivate some possibilities of the module by sending an adequate signal ("SAFE"). The core also sends with the signal an argument. This argument is a degree of degradation.

For example, the user has selected 5 cameras to get 5 point of view of the distant robot but the latency is too high. So this core will automatically use the maximum number of cameras possible to have a good latency in order to preserve a good experience for the user, the module receives “SAFE 4” with the argument “4” for 4 cameras. Figure 6 shows the software architecture with the use of 3 modules.

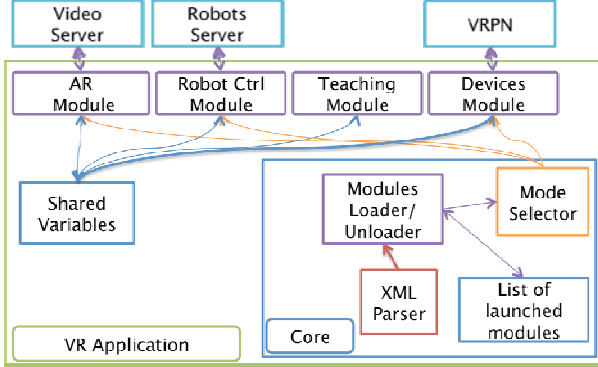


Fig. 6: Software architecture example with 3 modules for VR application. The core is the same for both platforms but modules are adapted to the support.

The main advantage of this architecture is that the user can dynamically add modules in the application without stopping it. For example, during a teleoperation task the user has initially chosen to free control the robot, however he needs to program a trajectory and stored in it. In this case, the system is available to add the trajectory module when the application is running. The prototyping system uses the web server of the mobile applications.

The user will use the Prototyping System to select all the elements he wants to use in the teleoperation application. In fact, a module is a package, which contains the module and an XML description of the module, which is containing methods and arguments. A final XML file is created which is containing all modules description and the user option. Figure 7 shows the path of the XML file.

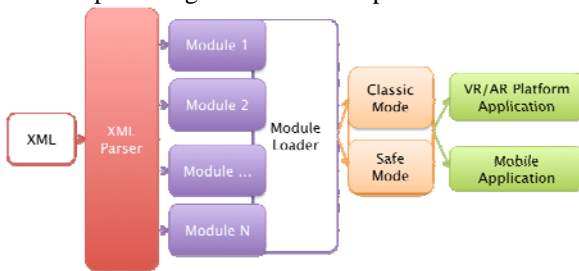


Fig. 7: The XML file is an input parameter of the Core. At the end, the result is the VR or Web application.

On the Prototyping System, a list of modules is displayed. This list is stored on a MySQL database. Clicking on the “more” caption permits to configure the options of the modules (number of cameras, interaction device to used...). Messages are displayed when the user passes the cursor on the element.

The user can hence click on 2 links: “Examples of use” and “Applications using the module” that permits to have more details on the module.

D. Examples of Modules

The Prototyping System (PS) permits to create a personalized application. In this part, we present 4 important modules used in the robot teaching application that is based on our architecture.

1) Collaborative Module

This module permits collaborative teleoperation tasks. However, the user is not forced to use the collaborative module. The non-use of this module will permit mono user teleoperation task. For the moment, this module includes text messages for communication, a basic coordination process and 2 types of robot control:

- Each user controls a virtual robot and the result of all movements will be transmitted to the real robot;
- Each user controls the virtual robot independently;
- Text messages for communication on the VR platform are not possible because users do not use a keyboard.

Therefore we are integrating audio and video communication.

2) Robot Control Module

This module is necessary to teleoperate the robot and must be configured. In fact, many ways of interaction are available as a mouse, keyboard, Flystick, Data Gloves and others. However, Flystick, Data Gloves can only be used on VR platform. On the VR application case, this module used the VR Device Module. The combination of the 2 modules will permit the teleoperation of the robot with a VR platform. The VR Device Module used VRPN and supports a huge list of devices. VR Devices needs VRPN to retrieve their data; keyboard and mouse do not need VRPN. VR Devices modules can activate audio and video on the platform also to get a 3D view.

3) Augmented Reality Module

The robot-teaching module permits to create real trajectories with the use of the real or virtual robot. The trajectory is stored directly on the robot while the use of AR module and this one will permit to create distant trajectory on virtual objects with the real robot. The trajectory is directly stored on the robot by using an appropriate communication protocol. Figure 10 illustrates the use of AR and Robot Teaching modules.

IV. FIRST APPLICATION USING THE UPDATED ARCHITECTURE

We have used our novel software and network architecture to develop a collaborative robot teaching teleoperation system. Two possibilities of control can be used. First, controlling virtual robots to produce a global command and then transmitting the result to the real robot (for a multi users case). A virtual robot is associated to each user. Each user can see the position of the user virtual robot effector. Figure 8 shows an example with the use of a web browser, an iPhone and a Virtools application.

Secondly, controlling the real robot to in this case

produce a command (for a mono user case). The user directly controls the robot with the robot teach (if the user is near the robot) or in a teleoperation mode (if distant). Figure 9 shows the screen where a HO is controlling the distant robot with a reproduction of the Fanuc Teach on the right. The user has 3 points of views corresponding to the 3 axis (X, Y, Z). The user can select the point of view by clicking on the “little” view. When clicking, the application displays the view and uses augmented reality to track the markers. The user can create a trajectory on the virtual object (here a car) by using the panel. The robot can be teleoperated in joint or tool mode.

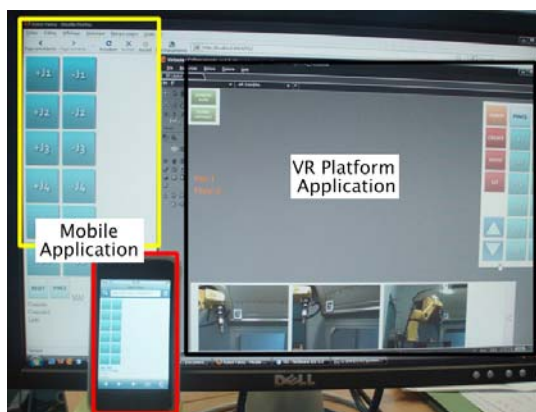


Fig. 8: Three clients (a Virtools, a Firefox and an Iphone client) using the collaborative Teleoperation system together.

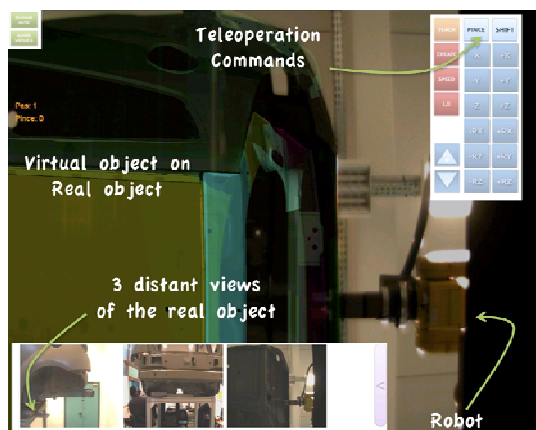


Fig. 9: View of the teleoperation system with the use of Augmented Reality and Teaching modules. The trajectory is made and stored with real robot.

V. CONCLUSION

We presented a new collaborative Man Machine Interface (MMI) allowing high-level teleworking activities and collaborative robot teleoperation using Augmented and Virtual Reality technologies. We started this work by identifying problems and limitations on actual systems. In fact, we wanted to integrate collaborative robot-teaching aspects with the support of multi interfaces and multi devices. Consequently, we have built our architecture with this constraint. One of the major problems encountered was the video streaming to permit a synchronous diffusion of the distant robot video stream. The second condition is to permit to quickly and easily add/delete/update modules on the system. To answer these conditions we have

proposed a Prototyping System, which permit to personalize an application by clicking on desired modules. PS permits to choose between a VR application compatible with the platform and a Web application usable on a Web Browser or any mobile device. We have tested the system with an iPhone, a Web browser and a VR application but no quantitative results have been gathered. An example of teleworking application is presented as a teleoperation of the remote 6 DoF Fanuc robot for robot teaching in a mixed environment. An important feature of this application is based on the use of different technological heterogeneous platforms to control the robot such TCP/IP. Future work will focus on many aspects as:

- Completing the transformation of the old ARITI system by developing missing modules;
- Performing evaluations on the interaction aspects (Mobile versus VR; Virtual versus Real, VR versus VR) with the use of an evaluation system [13] to improve the system;
- A new model of Virtual Fixtures as it exists on ARITI but for a collaborative case where users want to grasp the same object at the moment.

REFERENCES

- [1] B. Espiau. Téléopération avancée: Le projet de teleoperation avancée. Technical report, Saclay (France), December 1986.
- [2] T. Hasegawa, T. Suehiro, T. Ogasawara, T. Matsui, K. Kitagaki, and K. Takase. Model-based telerobot system manipulation skill. In 91 International Symposium on Advanced Robot Technology, pages 499-506, March 1991.
- [3] J Martins Jr, L. Camolesi and G. Caurin. Scara3D: 3-Dimensional HRI integrated to distributed control architecture for remote and cooperative actuation. In Proceedings of the 2008 ACM Symposium on Applied Computing, pages 1597-1601. Fortaleza, Ceara, (Brazil), March 16-20, 2008
- [4] M. Mallem, F. Chavand, and E. Colle. Computer Assited visual perception in teleoperated robotics. In J. Rose, editor, Robotica, volume 10, pages 93–103. Cambridge University Press, Cambridge University (England), 1992.
- [5] P. Milgram, S. Zhai, and D. Drasic. Application of augmented reality for human-robot communication. In IROS: Int. Conf. on Intelligent Robot and Systems, pages 1467-1476. Yokohama (Japan), July 1993.
- [6] S. Otmane, M. Mallem, A. Kheddar, and F. Chavand. Ariti: an augmented reality interface for teleopeation on the Internet. In Advanced Simulation Technologies Conf. (ASTC2000), pages 254-261. Wyndham City Center Hotel, Washington, D.C. (USA), April 12-20 2000
- [7] S. A. Greena, M. Billinghurstb, X. Qi Chena and J. G. Chase. Human-Robot Collaboration: A Literature Review and Augmented Reality Approach in Design, 2008
- [8] P. Boudoin, C. Domingues, S. Otmane, F. Davesne, N. Ourmadane and M. Mallem. Towards multimodal human-robot interaction in large-scale virtual environment. Proceedings of the 3rd ACM/IEEE Int. Conf. on Human Robot Interaction, Amsterdam (Holland), March 2008
- [9] V. Leligeour, S Otmane, P. Richard, and M. Mallem - Distributed Software Architecture for Collaborative Teleoperation based on Networked Mixed Reality Platforms. In 2nd IEEE Int. Conf. on Information & Communication Technologies: From Theory to Applications, pages 3498-3503, Damascus (Syria), 24-28 April 2006.
- [10] N. Khezami, S. Otmane and M. Mallem. A New Interface for Collaborative Teleoperation. In Int. Federation of Automatic Control. Phraha (Czech Republic), July 3-8, 2005.
- [11] C. Kemp, A. Edsinger, E. Torres-Jara. Challenges for robot manipulation in human environments. In IEEE Robotics &

Automation Magazine, New York (USA), v. 14, pages 20-29, March 2007.

- [12] N. Ouramdane, F. Davesne, S. Otmane and M. Mallem. 3D interaction technique to enhance telemanipulation tasks using virtual environment. In IEEE/RJS Int Conf. on Intelligent Robots and Systems, pages 5201-5207, Beijing, (Chine), October 9-15, 2006.
- [13] C. Domingues, S. Otmane, F. Davesne and M. Mallem, Creating 3D Interaction Technique Empirical Evaluation with the use of a Knowledge Database of Interaction Experiments. In IEEE Int Conf. on Human System Interaction 2008, Krakow (Poland), 25-27 May 2008.