Intuitively operable interfaces for space automation systems by means of projective virtual reality

Eckhard Freund, Jürgen Rossmann

Institute of Robotics Research, University of Dortmund

Otto-Hahn-Str. 8, D-44227 Dortmund, Germany

Email: rossmann@irf.de

www.irf.de

ABSTRACT

The general aim of the development of virtual reality technology for automation applications at the IRF is to provide the framework for *Projective Virtual Reality* which allows users to "project" their actions in the virtual world into the real world primarily by means of robots but also by other means of automation. The framework is based on a new task-oriented approach which builds on the "task deduction" capabilities of a newly developed virtual reality system and a task planning component. The advantage of this new approach is that robots which work at great distances from the control station can be controlled as easily and intuitively as robots that work right next to the control station. Robot control technology now provides the user in the virtual world with a "prolonged arm" into the physical environment, thus paving the way for a new quality of user-friendly man machine interfaces for automation applications. Lately, this work has been enhanced by a new structure that allows to distribute the virtual reality application over multiple computers. With this new step, it is now possible for multiple users to work together in the same virtual room, although they may physically be thousands of miles apart. They only need an Internet or ISDN connection to share this new experience. Last but not least, the distribution technology has been further developed to not just allow users to cooperate but to be able to run the virtual world on many synchronized PCs so that a panorama projection or even a cave can be run with 10 synchronized PCs instead of high-end workstations, thus cutting down the costs for such a visualization environment drastically and allowing for a new range of applications.

Keywords: Decentralized Computation, Distributed Simulation, Projective Virtual Reality, Forest Machine Simulator, Stereoscopic Viewing

INTRODUCTION

New virtual reality techniques offer the chance to convey information about an automation system in an intuitive manner and can combine supervisory capabilities with new, intuitive approaches to the control of the system. Our basic idea of an intuitively operable man-machine-interface is to provide a virtual reality system that automatically translates actions carried out by a user in the virtual, graphically animated world into physical changes in the real world, e.g. by means of robots or other automation components. The less the user in the virtual world needs to know about the means of automation which carry out the task physically, the better the design of the man machine interface. This idea actually forms the background for the new "Projective Virtual Reality"-methodology that is proposed in this paper: ***with the help of robots, changes made in the virtual world are "projected" into the physical world***.

Based on this new concept, a man machine interface has been developed in order to bridge the gap between virtual reality and robotics [4]. The idea here is not to develop just another VR system and to connect it to just another robot control system, but to develop a generic framework which embraces the latest VR system — or even one of the latest VRML browsers — and to connect it to an available robot control system. Only this approach allows the user to take advantage of the latest advances in the two fields, where enormous, yet almost independent, research is being conducted.

In this paper, it will be explained how the applicability of IRF's virtual reality system COSIMIR/VR was further enhanced by providing the capability to share the virtual world with multiple users, display the worlds on multiple screens and to automatically distribute the computational burden over multiple computers.

A SHORT HISTORY OF PROJECTIVE VIRTUAL REALITY

The Projective Virtual Reality System to be described here, is currently being used in different applications ranging from space laboratory servicing over industrial assembly applications to virtual reality based training. The current VR system used for all applications is based on IRF's robot simulation and virtual reality system **COSIMIR** (**C**ell **O**riented **Sim**ulation of **I**ndustrial **R**obots —http://www.cosimir.com).

1.1. Control of a multi-robot testbed for space laboratory servicing

The very first application of the new VR techniques, realized already in 1993, was to develop a man machine interface to control and supervise the **CIROS** (**C**ontrol of **I**ntelligent **Ro**bots in **S**pace) multi-robot testbed at the IRF. The aim of the CIROS project was to develop a versatile multi-robot control system capable of coordinating multiple robots in such a way that the robots would be able to support or even substitute for astronauts in routine tasks for experiment servicing and repair in a space laboratory environment.

Figure. 1: The CIROS multi-robot testbed

It was this testbed that in 1994 was controlled by means of projective virtual reality methods from California via INTERNET. The CIROS multi-robot testbed of the IRF in Germany was fully controlled by colleagues at the University of Southern California over a distance of more than 10000 km.

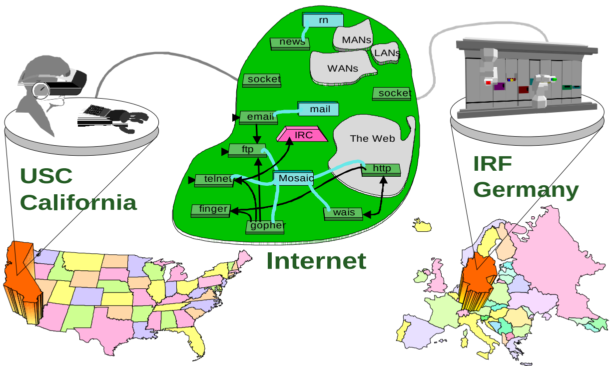

Figure 2: Control of the CIROS-testbed by means of "Projective Virtual Reality" over a long distance via INTERNET

The basic idea behind projective virtual reality is to let the user immerse into the virtual world and let him work in the virtual world in the same way as in the physical world. The developed virtual reality system COSIMIR VR is capable to "deduce" the user's intention from his actions in the virtual world using a petri-net based action tracking principle (see [4] for a detailed explanation). After the user's intention has been deduced, a corresponding task description is sent to a planning system which in turn orders a robot or another appropriate means of automation — in general terms: an agent — to carry out the task in the physical world. Thus the same task that has been carried out in the virtual world by means of a virtual hand is "projected" into the physical world by means of an automation agent. Projective virtual reality is a great help to make even the most complex automation systems easy to supervise and to command. Fig. 3 shows how simple it is to for example command the relocation of a sample container in the CIROS space laboratory environment: The user just grasps the container and takes it to the desired location.

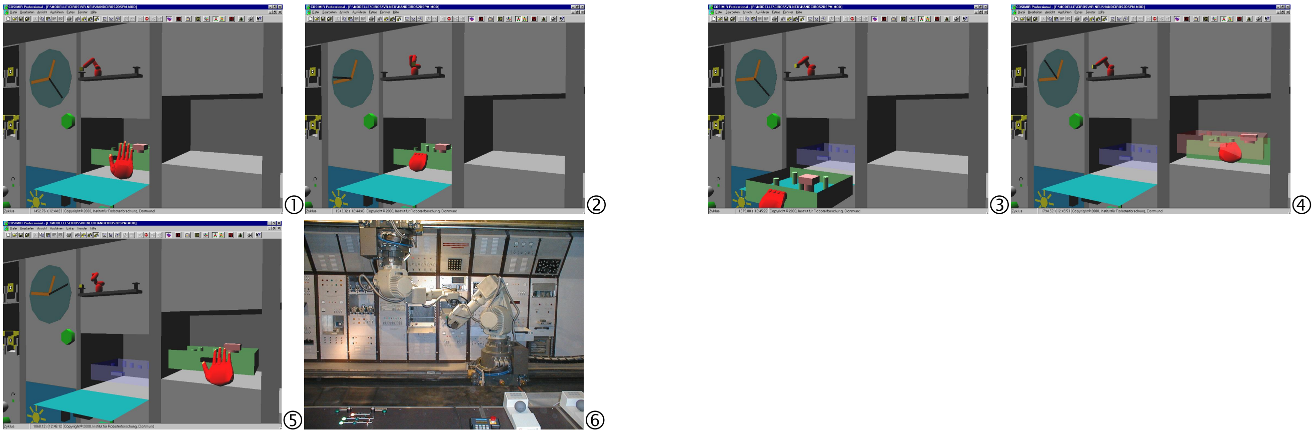

Figure 3: Sequence of user actions to move a sample container

Only picture ⑥ in fig. 3 indicates that the execution by means of the robots is far more difficult than the commanding in the virtual world: For this task two robots have to be employed to be able to safely carry and guide the container to its target position. This distribution of the task on the two agents is automatically taken care of by the resource based action planning component developed for CIROS.

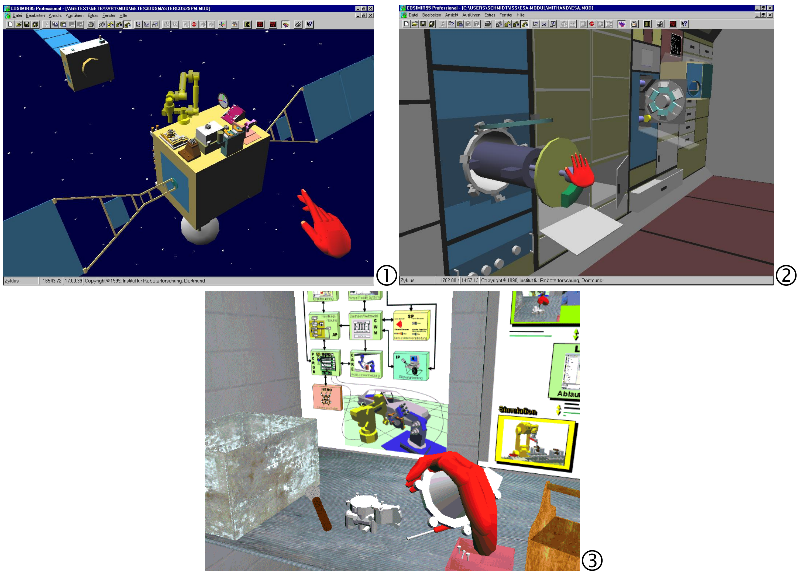

Figure 4: Projective Virtual Reality in outer space (①), inside the Columbus Orbital Facility (②), and in dismantling applications (③) on earth.

Fig. 4 shows that the development of projective virtual reality has already left the state of laboratory experiments. In April 1999 it was used to realize the ground control station for the robot ERA on board the Japanese satellite ETS VII. This mission to control the first free flying robot in space together with our colleagues from Japan was a great success and is described in greater detail in [8]. In a current project, projective virtual reality technology is being implemented to provide a similarly easy to operate interface (②) for the Columbus Orbital Facility (COF), the European Contribution to the International Space Station (ISS). Picture ③ of fig. 4 shows how the same technology is being used in an industrial application. Here it is employed to supervise and command the automatic disassembly of used car parts for recycling purposes. With this virtual reality based interface, the user is able to simultaneously command 4 robots and to "advise" them how to disassemble car parts.

2. DECENTRALIZED COMMANDING AND SUPERVISION

When training systems and simulators are considered in *production- and space environments, it is very important to provide not just a close-to-reality simulation, but also to provide an ergonomically suitable, multimedia capable environment which — at best — also allows the active cooperation of multiple users in the same virtual room. The first realization of the projective virtual reality system for CIROS (see section 2) which used a *Head Mounted Display (HMD)* or a single projection screen to provide a stereoscopic image of the virtual space laboratory module, suffered from the "tunnel view" the users get. Experiments showed that a field of view of only 60 to 70 degree was by far not sufficient to provide an ergonomic display of the virtual world. This experience and the wish to provide larger projection screens and to run multi-screen 3D workbenches and even CAVES^{TM} with COSIMIR VR led to the first ideas concerning the distribution of COSIMIR VR over multiple PCs to provide the necessary computational power in a decentralized simulation approach. This approach had to address to following issues:

1. The simulation software COSIMIR VR must be able to distribute itself over multiple PCs and to make sure that the states of the virtual worlds are synchronized. This allows to share the computational burden between multiple PCs and provides the required computational power to employ latest multimedia technologies e.g. for artificially generated surround-sound.
2. The graphical views generated by the different instances of COSIMIR VR must be synchronized in order to be able to arrange the screens in e.g. different panorama or cave configurations.
3. Besides being able to display the same virtual world on multiple screens, COSIMIR VR should also allow for different active users to share the same virtual world. For the ISS example, this makes sure that different scientists who are in charge of the experiments can work together in the virtual world as they would in the physical world — without having to travel.

The functionalities described above were realized by the approach depicted in fig. 5. This approach was implemented in COSIMIR VR in order to be able to visualize and to cooperate in a virtual world in a decentralized manner:

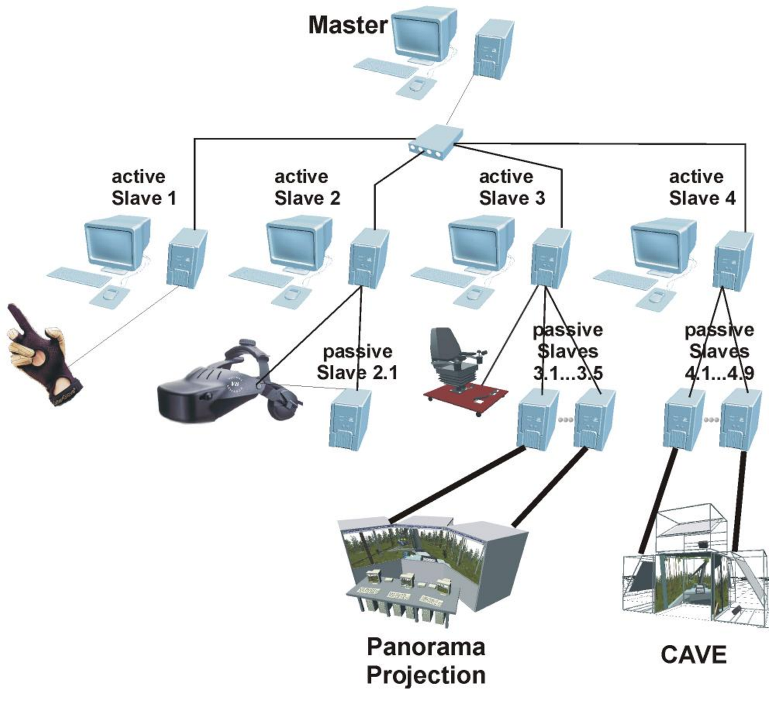

Figure 5: The COSIMIR VR concept for decentralized cooperation and visualization of virtual worlds

1. One (and only one) PC is determined to be the master in the virtual world. The master has the task to communicate with active slaves, to detect changed state variables and to communicate the changes to all slave computers. The master in general is in charge of making sure that all active slaves are keeping a consistent database of the state of the world.
2. For the slave PCs, two groups are distinguished:
   a) *Active slaves:* Active slaves are always associated with a user — they are each user's door into the cooperatively used virtual world. Active slaves communicate changes made by the user in the virtual world to the master which in turn updates the virtual worlds of the other active slaves. Thus, changes made to the virtual world by means of active slaves are reflected in all databases of the other users working in the same virtual world. In the example of the harvester simulator, the driver seats (fig. 5) are attached to active slaves which interpret the user's actions and actively transmit changes of state variables to the master computer which in turn further distributes the data to the other slave machines making sure that no transmission cycles and deadlocks are generated
   b) *Passive slaves:* Passive Slaves are used as plain rendering computers that are just used to generate a corresponding view of a scene whose viewpoint is determined by their active slave. Nevertheless, passive slaves are a very important part of the visualization concept of COSIMIR VR because they take the computational burden of the rendering process for large panorama projections or CAVES™.

The concept behind COSIMIR's decentralized cooperation and visualization capabilities shown in fig. 5 emphasizes the hierarchical concept by showing that the major coordination aspects between the different active users are taken care of by the master computer. Each active slave can have zero or more passive slaves to share the computational burden for the rendering of the images. The active slave on the left side of fig. 5 does not have a passive slave, because its user watches the virtual world just on a monitor — the required view is generated by the active slave itself. The next user is watching the same world through a data-helmet, so the generation of a right eye view and a left eye view is required to provide the stereoscopic representation of the virtual world. For this user, one view (left-eye) is generated by the active slave that also handles the user's input, the second (right-eye) view is generated by the attached passive slave. On the right side of fig. 5 it is shown how the same principle is applied to a 5-screen CAVE™. The stereoscopic views for each of the 5 screens of the cave are generated by two PCs (one PC for each eye-view), so altogether 10 PCs (1 active slave, 9 passive slaves) are required to run the cave — implying that the computer costs to run the caves do not exceed 15.000 $US thus paving the way for a whole new range of modern training and simulation applications.

3. THE ARCHITECTURE OF THE VIRTUAL REALITY SYSTEM

In order to support the desired capabilities for multi-user access and multi-screen projection, the architecture of the VR system has to reflect these issues directly. Furthermore COSIMIR VR is designed to serve as a suitable framework for a variety of realizations with a variety of interaction tools like the dataglove and sensorball and different types of head mounted displays, shutterglasses and mutli-screen projection equipment. Last but not least, the framework must also be able to embrace the action planning component and a VR system of the user's choice in order to really connect robotics and virtual reality in a general way.

3.1. The software-framework to realize Projective Virtual Reality

For the realization of the VR system the client/server approach shown in fig. 6 proved to be useful, because the multi-user and multi-screen aspect is inherent to this approach. The interaction components and their software-drivers in the corresponding interaction clients provide the input for the VR-Server, the central component of the VR system. With the help of this information, the VR-server then keeps current its internal world model, a central database, which contains different kinds of information related to the interaction of objects in the VR and which provide the basis for the simulation of different "real world effects" in the virtual world as explained below.

The main task of the VR-management system, the central part of the VR-server shown in fig. 6, is to coordinate the work of the different components of the world model. The VR-management system collects information from the different interaction clients like the dataglove or sensorball, detects changes in the states of these components and instructs a corresponding part of the world model to react on a specific change accordingly. It is the VR-management system which classifies the detected change and generates an appropriate message for the corresponding world model components to react on the change. If e.g. the gripping of an object is detected a corresponding message is generated and sent to the change-reaction-model which contains the task deduction petri nets and thus can deduce a task for the physical robots [4]. It is this part which makes the important difference between a Projective Virtual-Reality System, and a Virtual Reality System that is "closed" in the way that is performs a graphical display of an environment and allows an interaction with the user in this graphic world only without making the connection to the physical world.

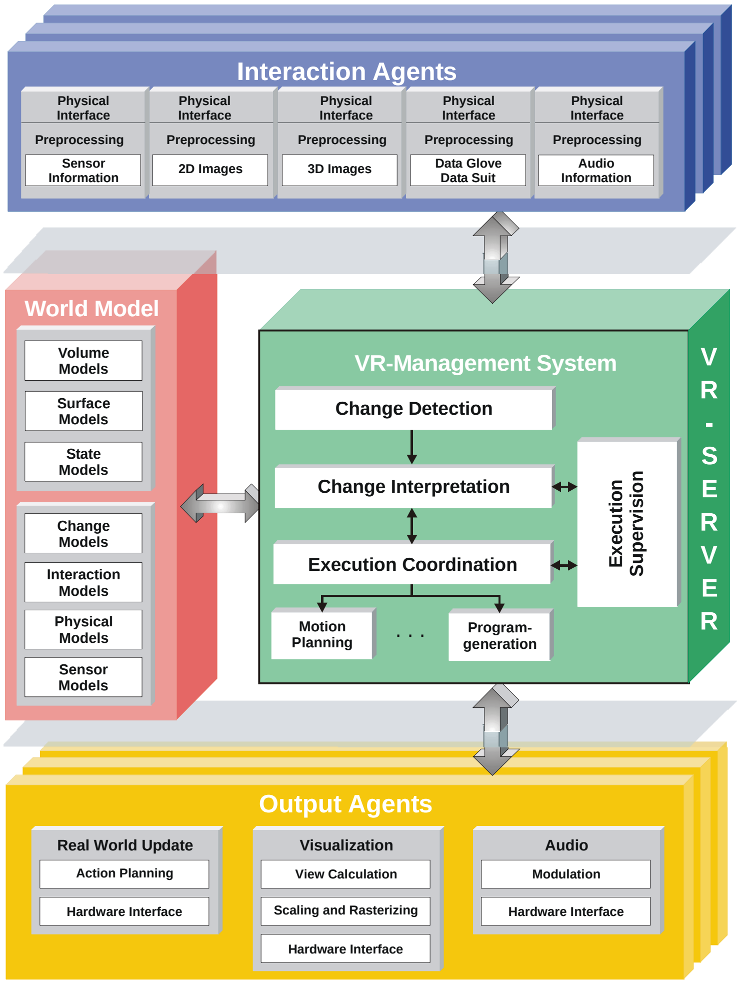

Fig. 6: The Client-Server Architecture of the realized Projective Virtual Reality System

Apart from supporting the key idea of projective virtual reality, the “change-detection” approach of the VR managment system also proved to be an excellent basis for the distribution of the virtual world over multiple PCs. In order to maintain the impression for the users to share a virtual world, it is exactly the detected changes that have to be communicated (— via the master computer depicted in fig. 5) to the active slaves in order to keep synchronized the states of the different world databases of the virtual worlds running on the slave PCs.

3.2. The VR world model components

Besides the VR-server whose main task is the detection of changes and the (re)direction of corresponding messages, the world model is one of the key elements in the projective VR architecture of fig. 6, because it supports the modular implementation of the necessary functionalities. The key idea is to understand the components of the world model as active objects, which communicate and cooperate with each other using well defined interfaces. The world model components are informed of new events in the outside world — or in the other users’ worlds — by the VR-management system and they only take care of their specific aspects of the modeled world.

The parts “volume models” and “surface models”, shown in fig. 6, contain the volume-descriptions in the form of BREPS (boundary representations) and surface descriptions by means of color-properties and texture-elements. The physical models enhance the geometric representation of the different objects in the simulated environment by physical properties for different purposes, like mass, friction, a restitution coefficient for momentum-based contact simulations. Further properties that are considered are conductivity, permeability, acoustic-reflection coefficients etc. for the simulation of different types of sensors. The state models store the different state-variables of the objects in the environment and also do the "bookkeeping", if an object is gripped by another. The state variables comprise dynamic parameters such as joint angles, -velocities, and -accelerations.

The group of four models below the group of three (fig. 6) described before contains additional application-specific information like the physical measurement principle and the measurement range of sensors in the *sensor models* or the petri-net representation of the task-deduction nets [4] in the *change-reaction model*. In contrast to the first group, the algorithms implemented for these world model components are much more elaborate as they all realize complex behavior of the objects in the virtual environment. For example, the sensor models realize the behavior of different sensors considering their physic al measurement principle, so that an inductive sensor cannot be used to detect an object whose physical parameters like conductivity and permeability are set to those of wood in its physical model.

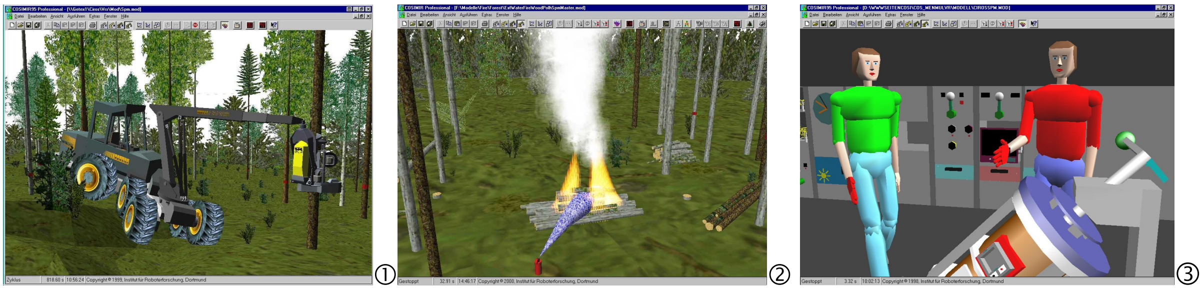

Fig. 7: Simulation of physical effects: A harvester cutting down a tree, a burning fire and two walking humans

The next class of models in the VR-architecture is the interaction model which handles the interaction between the different objects in the simulated environment. These models must assure that objects may be gripped and transported, that they are transported together with an object that they are placed on, and that a drawer slides open, when the user pulls the handle. In a more complex interaction, it is even possible to simulate the human gait as the interaction between a simulated human with the ground (fig.7 ③). Last but not least, the algorithms of an interaction-model dedicated to the simulation of physical effects are able to simulate the effects of gravity and of contact between different objects in the environment. This was exploited by the realization of a VR based harvester (wood-cutting) simulator and a fire-fighting training system as shown in fig 7.

The advantages of the proposed structure are its modularity and its portability to multiple hardware platforms: High-end systems based on graphics-workstation and high-end — and high priced — VR interaction components can be configured when the emphasis is on the perfect presentation of the virtual world.

3.3. Multi-Screen Displays

Whereas the discussion above mainly focussed on the aspects of connecting automation systems and virtual reality systems as well as providing multi-user access to virtual world, an impressive "by-product" of the realized structure is that it does not only support multiple users, but also multiple screens for each user as indicated before in fig. 5. Fig. 8 shows examples of different configurations that have been or are about to be realized with COSIMIR VR.

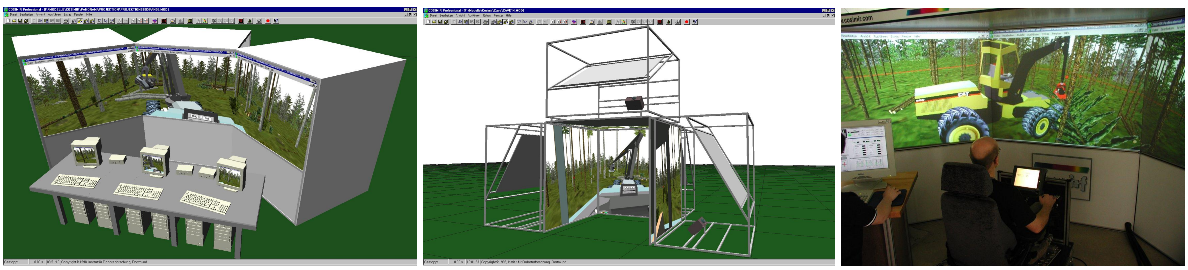

Figure 8: Multi-Screen Displays: Stereoscopic Panorama Projection, Stereoscopic CAVE configuration

4. CONCLUSION

With the latest developments at the IRF, Projective Virtual Reality Technology can make the next leap forward as an intuitive and ergonomic man machine interface. Projective VR efficiently makes the connection between VR- and robot control technology in a way, that the robots serve as the “prolonged arm” of the user handling objects in the virtual world. With the task deduction and action planning components it is possible to “project” the user’s actions from the virtual world into the physical world by means of robots, which means that robots physically carry out the task that the user conducted in the virtual world. The feasibility of this approach has been thoroughly tested and the implementation is being used in several applications today. These ideas, combined with latest development in the field of the automatic distribution of virtual worlds over multiple computers will further contribute to new application fields to be conquered with this modern and appealing visualization, supervision and commanding technology.

REFERENCES

1. A.K. Bejczy: *Virtual Reality in Telerobotics*, Barcelona, Spain, August 1995. Available over the Web under HTTP: http://techreports.jpl.nasa.gov Directory: 1995 File: 95-0954.pdf
2. A. Cai, T. Fukuda, F. Arai, K. Yamada and S. Matsuura, *Path Planning and Environment Understanding Based on Distributed Sensing in Distributed Autonomous Robotic System*, in Proc. of the 4th Int'l Workshop on Advanced Motion Control (AMC'95-MIE) Vol.2, pp.699-704, 1996..
3. E. Freund, M. Krämer, J. Rossmann.: *Towards Realistic Forest Machine Simulators*; AIAA-2000-4095
4. Freund. E.; Roßmann, J.: *"Projective Virtual Reality: Bridging the Gap between Virtual Reality and Robotics"*, IEEE Transaction on Robotics and Automation; Special Section on Virtual Reality in Robotics and Automation; pp. 411-422, Vol 15, No. 3, June 1999.
5. E. Freund, J. Roßmann, J. Uthoff, U. van der Valk: *Towards realistic Simulation of Robotic Workcells*, Proceedings of the IEEE/RSJ/GI Intelligent Robots and Systems, IROS, München, Deutschland, September 1994.
6. E. Freund, J. Roßmann, K. Hoffmann.:" Automatic Action Planning as a Key Technique for Virtual Reality based Man Machine Interfaces", Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI’96), pp. 273-280, Washington D.C., USA, December 1996.
7. J. Rossmann, T. Wagner: *Forstmaschinenführer in der dritten Dimension*, Forst & Technik, Heft 2/99, BLV Verlag GmbH, München.
8. E. Freund, J. Rossmann, M. Schluse: *Projective Virtual Reality in Space Applications: A Telerobotic Ground Station for a Space Mission*, Sensor Fusion and Decentralized Control in Robotic Systems III, part of SPIE’s Intelligent Systems and Smart Manufacturing, Nov. 2000, Boston, MA, USA