A Virtual Reality System for Network Communications - Multi Party Co-operative Work in Real-time -

Katsuya Shinohara, Ryuichi Hiraike, Mamoru Hashimoto, Nobutatsu Nakamura
C&C Systems Research Laboratories, NEC Corporation
1-1, Miyazaki 4-chome, Miyamae-ku, Kawasaki, Kanagawa 216, JAPAN,
phone: +81-44-856-2175, fax: +81-44-856-2232, email: shino@tsl.cl.nec.co.jp

1 Introduction

Tele-virtuality is an important current research topic of virtual reality technology. In order to realize it, the system reported here links virtual reality technology with network communications. Each workstation in this system has an identical virtual world. Each user interacts with virtual objects in the virtual world at his workstation, and his actions and their results are reflected in real time in the other virtual worlds. The system enables several users, working from different locations, to participate in a common virtual world and to carry out co-operative work in real time, as shown in Fig. 1.

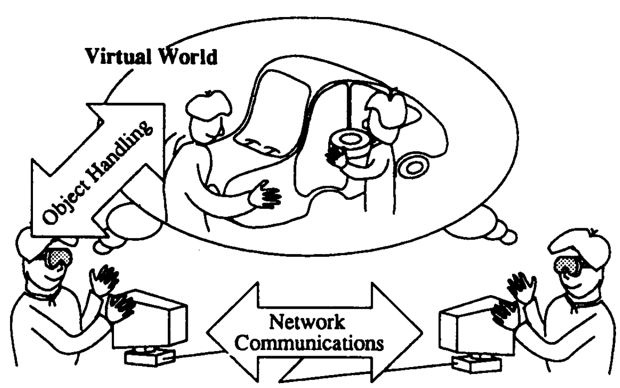


Figure 1: Co-operative work in a virtual world

In Section 2, object handling methods in one workstation are described, while data communication among workstations is described in Section 3.

2 Object Handling

A user of this system wears two VPL DataGloves on his hands and liquid crystal shutter eye-glass for stereoscopic display. Each workstation has its own three-dimensional virtual world description, which consists of shape, location, direction and color data for virtual objects in the virtual world and those for users. Each virtual object shape is represented in a B-spline free form surface model or a polyhedron model. A workstation gets operator's hand location and direction data from DataGloves, and reflects them into the virtual world description. Concurrently, it displays the virtual world to the operator using computer graphics.

Object Grasping

A user can grasp an arbitary object with a virtual hand attached to his virtual arm. In order to grasp an object, the user puts his virtual hand on the object and closes his fist. When a user grasps an object and moves his hand, the object follows the motion.

Photo 1 shows a grasping operation. A virtual right hand, in the figure, grasps the body of an airplane model, while a virtual left hand grasps its right wing.

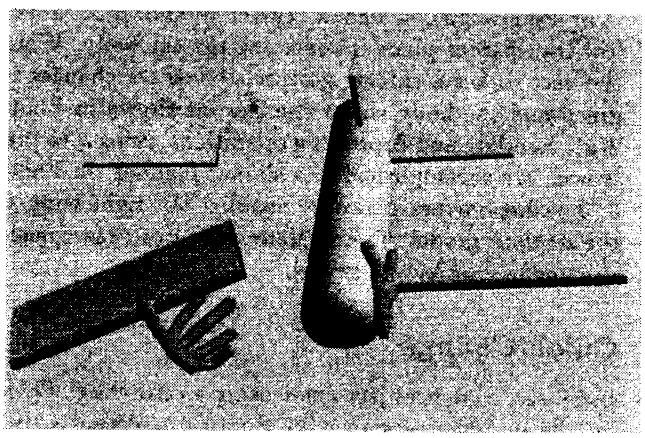


Photo 1: Grasping Operation

Shape Changing

In order to change an object's shape as desired, a user grasps a virtual object, and operates on it directly, using a Plane-cursor[1] attached to his arm in place of a virtual hand. It works as both a virtual cutter and a virtual trowel. With a virtual cutter, cutting operation by infinite-plane is carried out on the object, while a trowel is used for sheet bending or depression. Figures

2 and 3 show cutting operation and bending operation, respectively.

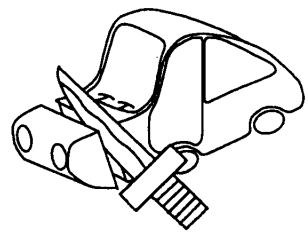


Figure 2: Cutting Operation

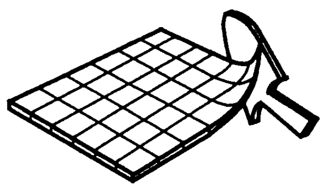


Figure 3: Bending Operation

In cutting operation, a portion of the grasped object at the rear of the infinite-plane is removed, when the user closes his fist. Cutting operation is accomplished when a user grasps an object, puts the cutter plane on his other hand in the object and makes a cutting gesture.

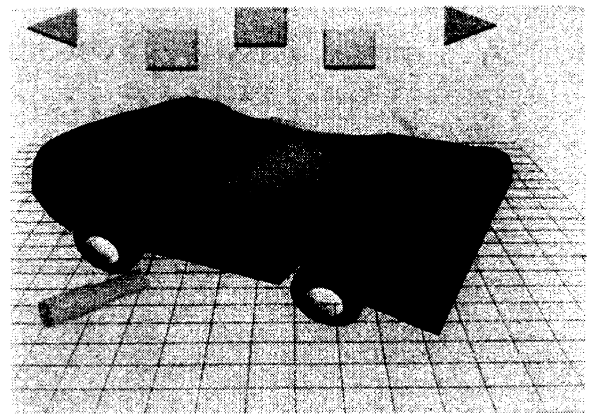
In Photo 2(a), a virtual cutter is shown as a yellow transparent plane intersecting the car body. When the user makes a cutting gesture, the cutter changes to green and the body shape changes, as shown in Photo 2(b). Bending and depressing operations, using a virtual trowel, are accomplished in a similar manner. In Photo 3, a yellow virtual trowel is bending the right wing of the airplane model. The airplane body nose has already been depressed by the trowel.

Color Change

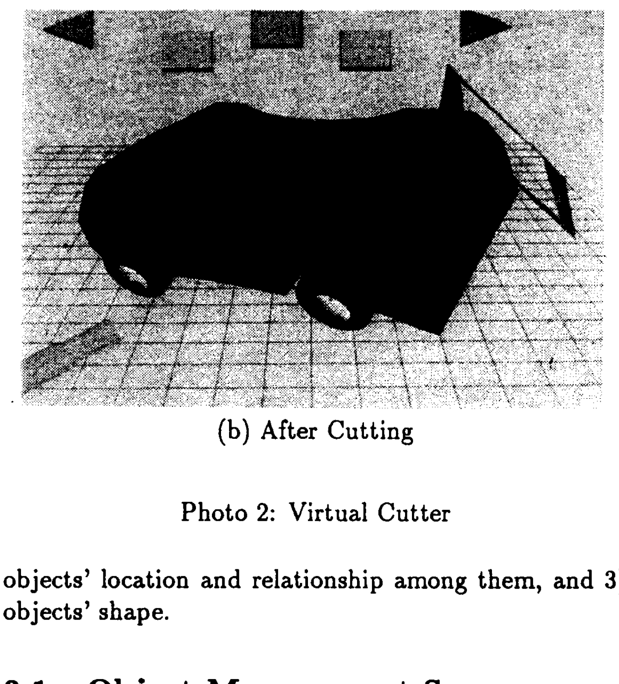
Users can change object color using a color-ring offered by the system, as shown in Photo 4. When a user changes the color of an object, he puts the object at the center of the ring and indicates the desired color on the ring, using a virtual pointer attached to his wrist.

3 Network Communications

In order to share one virtual world among several workstations connected with each other, workstations communicate the world description. For communication time reduction, sent date is limited to the changed portions of the model only. There are three sent data categories, 1) operator's hand location and direction, 2) objects' location and relationship among them, and 3) objects' shape.



(a) Before Cutting



(b) After Cutting

Photo 2: Virtual Cutter

3.1 Object Management Server

If two or more users, at the same time, grasp the same object and move it or change its shape as they desired, the object shows various positions and shapes at different workstations. In order to avoid this conflict, a workstation, called an object management server, connected to the network, controls such simultaneous handling by different users on one object.

The object management server works as follows:

- Initializing Common Virtual Objects

    1. At first, each user informs the object mangement server of information regarding his virtual objects and hands that he intends to put in the common virtual environment. The information consists of initial object shape, position, direction and color.

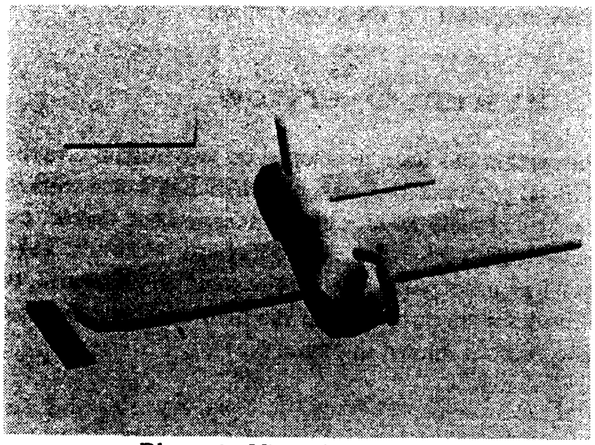


Photo 3: Virtual Trowel

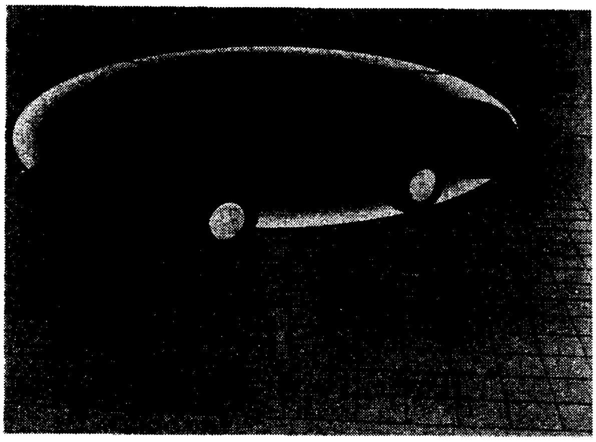


Photo 4: Color Ring

2. The object management server receives the information regarding objects and hands, assigns unique identifiers to them and sends the object information and identifiers to user workstations.

3. The user workstations receive information from the object management server. Each makes its own virtual world description, using this information, so that all workstations have an identical virtual environment.

- Object Handling Privilege control

  1. When a user at a workstation closes his hand near an object, his workstation so notifies the object management server, considering it as request for the privilege to handle the object.
  2. If the object handling privilege has not been assigned to any other user as yet, the object management server grants the implied request and invests the user with the privilege. Otherwise, his request is rejected.
  3. When the privilege is assigned, the user can grasp and move the object, and change its shape as he desires. On the other hand, were the request to be rejected, the object does not follow his hand motion.
  4. When the user releases the object, his workstation so notifies the object management server. The object management server then resets the object's handling privilege.

- Changing Information Announcement

  1. When a user moves his hands, his workstation sends the motion data to the object management server. Additionally, when the user has been assigned the object handling privilege and moves or changes the object shape or color, his workstation modifies the object description in its own virtual world according to users action, and sends the motion or changing information to the object management server.
  2. The object management server sends the information to the other workstations.
  3. The other workstations receive the information and move or modify the object in its own virtual world description, according to this received information.

The object management server acts as the communication center in the network, as explained above. All data regarding the users' hand motion and object modification are sent to the object management server. The object management server sends the information to all workstations, except for the workstation from which the information are originating. So, if the object management server processing speed is relatively low, it becomes a bottleneck in the network communications and causes delays.

In order to speed up the object management server processing and communications, the super server-client method is adopted. At the object management server, multi server-processes are concurrently active. Each server-process is an agent for one workstation. It communicates with its client workstation, using interprocess-communication protocol. Information is exchanged among the server-processes through the common shared memory.

3.2 Communication Process in Workstations

Another probable communication bottleneck would be a workstation with the lowest processing performance.

Main workstation functions are: 1) sensing users action, 2) communication with the object management server, 3) modification of its own virtual world description and 4) displaying the virtual world to its user using three dimensional graphics. World description modification and three dimensional displaying usually require a large amount of processing. So, at a workstation with low performance CPU or graphics facility, modification and displaying probably will not catch up with data inflow from the object management server. In such a case, the workstation becomes a bottleneck in the network communications and causes delays.

In order to resolve this problem, communication is carried out independently from the other jobs. The communication process receives data from the object management server, and updates the objects' position and color table in a shared memory. Other jobs read the latest data from the shared memory. When shape modification data is sent from the object management server, the object's position and color data at that moment are marked to be read from the other jobs. Succeeding data, if any, is heaped upon the marked data.

3.3 Geometry Data Transmission

When an object shape is modified, a large amount of modification data should be sent on the network, because the accurate shape description consists of a large amount of complicated geometry data. In order to realize speedy transmission, a difference shape data protocol has been developed.

After changing the object shape by a user, his workstation sends out on the network the shape data limited to the changed portions of the model only. The other workstations receive this data and reconstruct the object shape, according to this received data. The amount of data sent on the network is relatively small, so that workstations can communicate with little time lag.

The sent data format depends on the geometry model representing the object shape.

In case of B-spline free-form surface model, an object shape is represented with a certain number of three dimensional control points and two knot-vectors. Movement of one control point causes shape change in the portion depending to the control point. When a user changes the shape for a portion of an object represented in B-spline model, motion vectors for some control points in the model are computed, so that all the control points represent the shape of the result object. Instead of all the control points and two knot-vectors, his workstation sends only the motion vectors to the object management server. The other workstations receive them from the object management server and change the control points' position in their own virtual world descriptions.

In case of a polyhedron model, position and direction data for plane-cursor are sent to the object management server, when the user makes a cutting gesture. The other workstations receive the data and calculate the cutting operation result for the object shape.

4 System Overview

The system has been developed on two Silicon Graphics IRIS-4D workstations, plus one NEC EWS4800 workstation, which acts as an object management server, connected with each other via a local area network, Ethernet. Two users participate in the same virtual world, but each views this world from his own spatially disparate viewpoint, as shown in Photo 5.

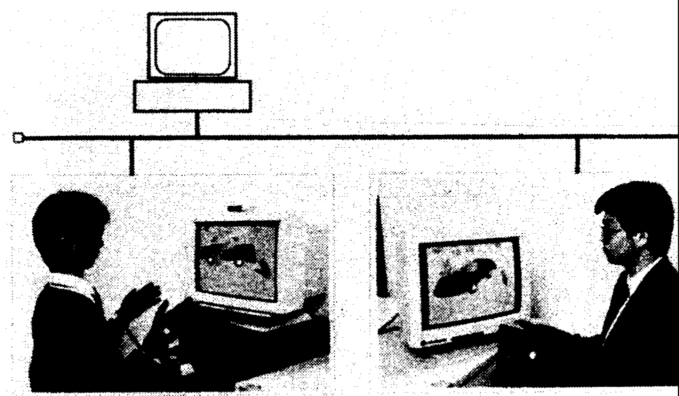


Photo 5: System Overview

5 Conclusion

Virtual reality technologies for network communications have been developed, and a prototype system has been realized using them. Although the prototype system is presently developed on workstations, connected via a local area network, systems using various networks, such as a broad band network, are intended to be developed in the future.

Acknowledgment

The authors express their appreciation to Dr. Ishiguro and Dr. Yamamoto of NEC Corporation for furnishing the opportunity to carry out this research and their encouragement. The authors also thank Dr. Nishitani and Mr. Kawagoe for their helpful advice and significant discussions.

References

[1] H. Sekine and H. Terajima, "A Visual Solid Modeling System using a Plane-cursor", Proc. 13th COMPSAC, 1989