Proceedings of Medicine Meets Virtual Reality 2002 Conference, IOS Press Newport Beach CA, pp. 362-368, January 23-26 2002

Shared Virtual Environments for Telerehabilitation

George V. Popescu[1], Grigore Burdea and Rares Boian

Center for Advanced Information Processing,
Rutgers University, Piscataway, N.J. 08854, USA.
http://www.caip.rutgers.edu/vrlab

Abstract.
Current VR telerehabilitation systems use offline remote monitoring from the clinic and patient-therapist videoconferencing. Such "store and forward" and video-based systems cannot implement medical services involving patient therapist direct interaction. Real-time telerehabilitation applications (including remote therapy) can be developed using a shared Virtual Environment (VE) architecture. We developed a two-user shared VE for hand telerehabilitation. Each site has a telerehabilitation workstation with a videocamera and a Rutgers Master II (RMII) force feedback glove. Each user can control a virtual hand and interact hapticly with virtual objects. Simulated physical interactions between therapist and patient are implemented using hand force feedback. The therapist's graphic interface contains several virtual panels, which allow control over the rehabilitation process. These controls start a videoconferencing session, collect patient data, or apply therapy. Several experimental telerehabilitation scenarios were successfully tested on a LAN. A Web-based approach to "real-time" patient telemonitoring - the monitoring portal for hand telerehabilitation – was also developed. The therapist interface is implemented as a Java3D applet that monitors patient hand movement. The monitoring portal gives real-time performance on off-the-shelf desktop workstations.

1. Introduction

Recent research investigates Virtual Environments use for home-based orthopedic rehabilitation [2,3,4,7]. Haptics increase patient immersion and participation and could potentially lead to faster recovery. Several networking solutions were proposed for home-based telerehabilitation [6,7,9]. These systems use a "store and forward" architecture for data collection and videoconferencing for patient-therapist interaction.

The VR-based telerehabilitation architecture presented in [7] is the reference point for our current work. The architecture supports offline interaction between therapist and a VR-enabled patient site. The telerehabilitation system contains a PC workstation, a novel Multipurpose Haptic Control Interface, the Rutgers Master II (RMII) force feedback glove, and videoconferencing hardware. The Client/Server software architecture implements a

[1] Currently with IBM T.J. Watson Research Center

“store and forward” type of system. The Client (patient home) runs VR simulations and collects real-time patient data. The Server (clinic site) stores patient medical records and runs data analysis and visualization software. Pilot clinical trials were performed in 1999 at Stanford Medical School (client site), with rehabilitation progress being monitored remotely from Rutgers University (Server site) [2].

The “store and forward” system described above is insufficient for implementing telerehabilitation services involving patient-therapist direct interaction. Shared Virtual Environments technology can be used to implement real-time interactions between physician and patient. This technology would require high-speed, low-latency communication networks in addition to advance human-machine interfaces. Currently such network infrastructure (e.g. Internet 2 [5]) is only available for research prototyping. In the future, these networks along with broadband consumer access (DSL, cable modem) will have the potential of supporting real-time telemedicine services.

This paper proposes a shared Virtual Environment telerehabilitation architecture. Simulated physical interactions between therapist and patient are implemented using force feedback. Data transmitted between the two sites includes audio, video, images, scene graph information, force, and control commands. The shared VE telerehabilitation system is presented in Section 2. Section 3 describes several “real-time” hand telerehabilitation experiments implemented on the shared VE platform. A Web-based approach to “real-time” patient telemonitoring - the monitoring portal - is presented in Section 4. Section 5 concludes this paper.

2. The Shared VE Telerehabilitation System

The shared VE telerehabilitation system is a software platform for real-time patient-therapist interactions. Its two sites are each equipped with a workstation, a video camera and an RMII force feedback glove [1]. The VE allow user control of virtual hands and haptic interaction with virtual objects. The shared VE uses a replicated database of virtual objects. All static content (3D objects, textures, sounds, etc.) is stored locally at each site. Therefore only dynamic updates (3D positions and rotation angles) are transmitted during the simulation. Additional data transmitted between the two sites include audio, video, forces, images, graphs and control commands.

Hand positions and finger joint angles are continuously sent over the network from each site, in order to animate the corresponding remote hand. Additionally, the forces displayed to the patient’s hand are sent over the network to the clinic site. This information is used to display at both sites force and effort - calculated as an integral of forces displayed at patient’s hand - visual feedback. Simulated physical interactions between therapist and patient are implemented using hand force feedback. This is achieved by replicating on the remote user the forces felt by the local user. Force data read from a local haptic glove is sent over the network and displayed on the remote glove.

Another component of the shared VE telerehabilitation system is the videoconferencing window. CuSeeMe videoconferencing software [12] is installed at the clinic and patient sites and runs as a separate program. The VE contains graphic elements (push buttons) which allow either the therapist or the patient to open a video consultation channel.

The architecture of the shared VE telerehabilitation system is presented in Fig. 1 [8]. The application was developed using WorldToolKit [10] graphics library. In addition to the graphics loop, separate threads run the database update, force feedback loop and the network communication loop. Both sites include an application controller, which implements the control logic of the simulation. Patient force and motion data are continuously available at the therapist site and can be stored on demand in the clinical database. The haptic thread displays the interaction forces on the RMII glove. The network protocol threads are responsible for formatting/parsing the messages and sending them to the remote site at specified update rates. The position and patient force data are sent at the graphics frame rate speed. The forces can be sent at either graphics or haptics thread update rates, depending on the application control mode.

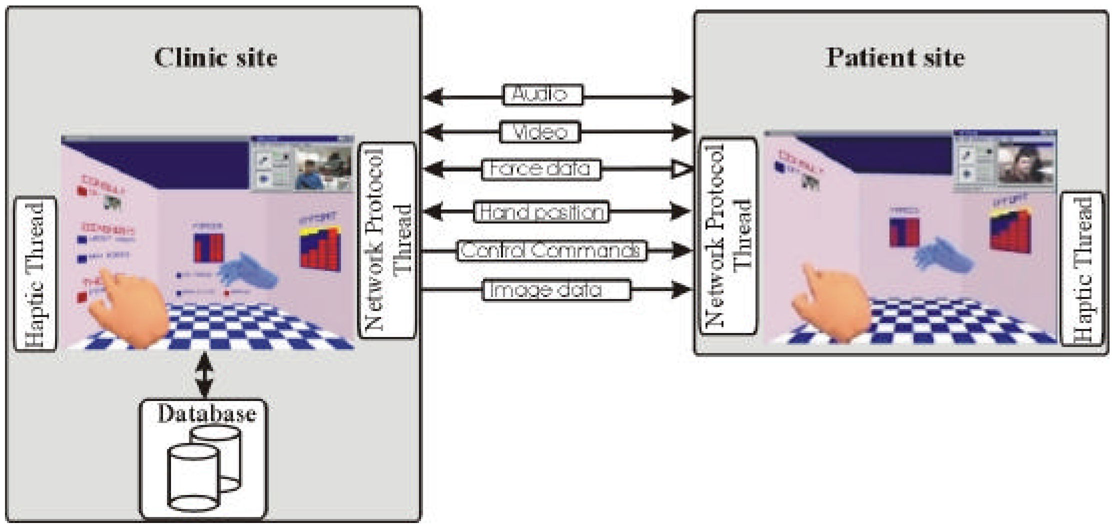

Figure 1: The Shared Virtual Environment Telerehabilitation System [8]. © 2001 Rutgers University.

2.1 The Shared Virtual Rehabilitation Room

The *Shared Virtual Rehabilitation Room* (SVRR) is a VE application implementing the above telerehabilitation architecture (Fig. 2). The environment contains virtual displays, user hands (local and remote) and control panels. The virtual displays are used to visualize the force and effort exerted by the patient’s hand. Each VE shows two virtual hands, one controlled by the local user and the other one controlled by the remote user. At each site, the remote hand is displayed in transparent blue (50% transparency), so that it doesn’t obstruct the local user view of the 3D buttons. The transparency is also an intuitive way to identify the remote objects.

In addition to the shared objects, the *SVRR* contains customized control elements: master control panels at the therapist site and a slave panel at the patient site. The master system virtual panels allow the therapist to control the rehabilitation process. The therapist can start a videoconferencing session (“consult” panel), collect patient data (“diagnosis” panel) or apply therapy (“therapy” panel). The “consult” panel allows the therapist or the patient to start the videoconferencing window by pushing the on/off button on the wall. The “diagnosis” panel has two buttons: one for measuring finger joint angles and the other for measuring the maximum forces exerted by the patient’s hand. The “therapy” panel has a single entry. When this is activated, the therapist can send a control command at the patient site to indicate the force level. Possible options are: “no forces”; “constant” - set constant level forces; “spring” - set forces proportional with finger displacement; “replicated” – send the target forces applied to therapist hand to be replicated at the patient site.

In addition to these panels, a “graphic board” switch is used to display a virtual whiteboard. The whiteboard can display a sequence of images, similar to a slide projector. The images - representing X-ray, patient reports, graphs, drawings, etc. - are displayed as textures mapped on the whiteboard.

The slave panel at the patient site has limited interaction modalities, its primary task being to feedback force and visual information. The forces rendered here are either the result of patient's hand interaction or commanded by the therapist from the remote site. The only control items in the Virtual Environment are the "consult" and whiteboard switches. These allow the patient to start a videoconsultation session and to select the desired image to be displayed on the whiteboard.

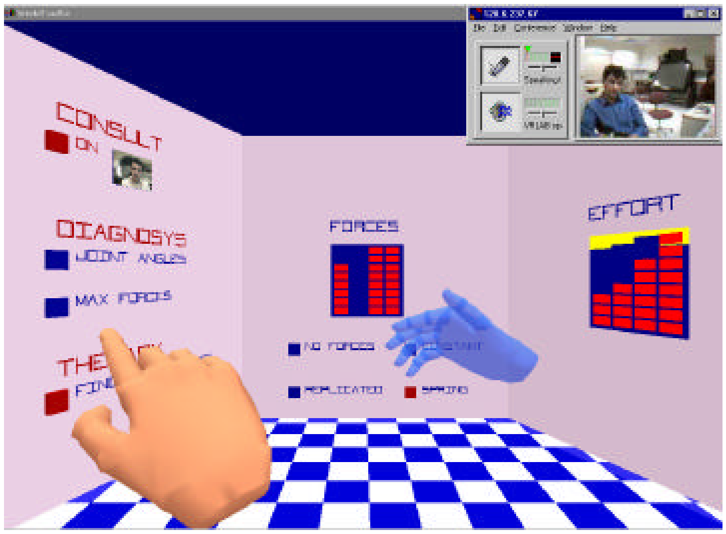

Figure 2: The Shared Virtual Rehabilitation Room (clinic site) [8]. © 2001 Rutgers University.

2.2 Application Control Logic

The session control diagram of the clinic site applications is shown in Fig. 3. A similar state machine is used to control the client application [8]. Synchronization of the two sites is based on a command protocol, which sets the application state in one of several modes: *diagnosis*, *therapy*, *graphic board* and *neutral*.

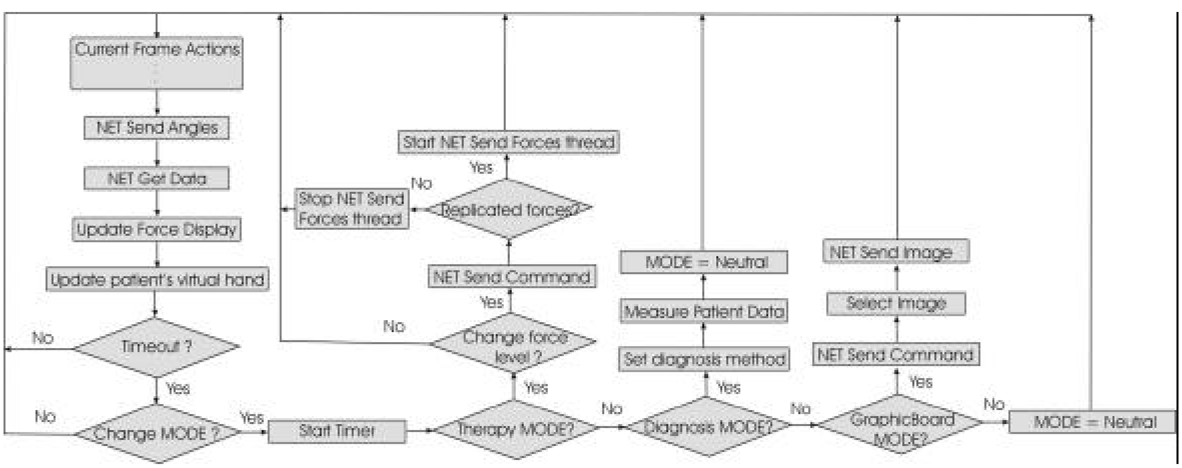

Figure 3: *SVRR* Session Management and Communication Diagram - Clinic site.[8]

The *diagnosis* mode allows patient force and range measurements. The *therapy* mode sends finger forces from the therapist site to the patient site similarly to a robotics telemanipulation application. Since these forces are used as targets for force feedback control the transmission rate should match the force feedback loop update rate (about 200 updates per second for the RMII glove [1]). Therefore when this mode is selected, the applications start an additional network thread, which runs at the above-mentioned speed. In *graphic board* mode images are sent remotely from clinic site for display on the patient's whiteboard.

The shared VE application described above uses a custom communication protocol. There are five types of message: angles, positions, forces, images and commands. The packet format of the network protocol is <*tag, data, length*>. The tag is used to identify the message type. Each type of message is parsed and processed differently by the network protocol thread (e.g. when *tag*=NET_ANGLES the *data* package contains 20 hand joint angles). The command messages used for session management synchronize the application modes between the two shared-VE sites. Thus when the therapist site changes to replicated force mode, the patient site is switched to the same mode. Table 1 shows the message formats used by the protocol.

Table 1: *SVRR* message formats

Tag	Message type	Bytes
NET_ANGLES	Hand Joint Angles	80
NET_POSITION	Hand Position	24
NET_FORCES	Finger Forces	16
NET_IMAGE	Image	Variable
NET_COMMAND	Command	1

3. Shared Virtual Rehabilitation Room Experiments

The *SVRR* application was tested on a LAN at the Center for Advanced Information Processing, Rutgers University. Two PCs with high performance graphics cards, videoconferencing capabilities and RMII haptic interfaces were used in the experiment. The shared VE contains about 8000 Gouraud shaded polygons. The two systems had different processing power, and therefore ran the graphic and network loops at different speeds. In order to prevent the overflow of the slower computer, the graphic and network update rate was limited to 20 fps at both sites. Patient force data were logged in the clinical database during the tests. Force data were sampled one time per graphic frame. This experimental setup was used for several telerehabilitation experiments:

a) Teletherapy: The teletherapy experiment allows the therapist to control the patient's hand movements. The force control method using replicated forces from clinic site was used to control the forces applied to the patient's hand (position control can also be implemented). This method was used to implement a standard hand rehabilitation exercise: *finger stretching*. In this exercise the therapist is virtually molding the patient's hand. The current RMII glove could only be used to open up patient's hand, as it provides one degree of force feedback per finger (opening the hand). In addition to *replicated* forces, *spring* and *constant* forces of different intensities were applied to patient's hand.

b) Telediagnosis: The *SVRR* allows the therapist to collect data from the patient during a real-time session. The videoconference application allows the therapist to give instructions to the patient ("open hand", "close hand", "grasp", etc.) and asks the patient to execute standard tests for evaluation purposes. The therapist measures joint angles, finger mobility, maximum force, and hand range of motion, which are automatically entered in the clinical database.

c) Telemonitoring: The therapist monitors patient hand movement, forces and mechanical effort applied by the patient's hand. Since both virtual hands (local and remote) are displayed in the shared virtual environment, the therapist can observe unusual hand movements and incorrect routine execution. The graphic panels mounted on the walls of SVRR provide force and effort information feedback.

To characterize *SVRR* traffic requirements, several network traffic sessions were recorded. As expected, the network traffic (in a LAN) produced during therapy and diagnosis modes is almost constant. The application uses only 45 Kbps on average in these modes. Switching to *graphic board* mode creates bursty traffic. An average bandwidth of 1.2 Mbps was needed in order to transmit high quality images during a graphic frame (the image transmission interval can be set higher in order to accommodate lower bandwidth). Videoconferencing creates network traffic of about 300 Kbps, depending on the quality (size, compression) of the transmitted video. The critical parameter for force replication is the network round trip time (RTT). Haptic data requires a maximum time delay of less than 100 ms for stable control. This requirement is easily met in a LAN setup (RTT in the order of couple of milliseconds), but can also be satisfied by Internet2 connections (measured RTT for Rutgers-Stanford connection is about 80 ms).

4. Web-based telemonitoring portal

A web-based telemonitoring portal was implemented as an extension of the shared VE telerehabilitation platform. The portal running at the clinic site allows telemonitoring of hand motion and patient data collection. The patient site runs several VR rehabilitation exercises [7]. The data flow in this case is unidirectional (from patient to clinic). The server application at the clinic receives sets of sensor data and exercise monitoring parameters. Similarly to the architecture presented in Fig. 1 the server continuously samples and stores patient data in the clinical database.

The web-portal is implemented as a Java3D [11] applet that displays a virtual hand model controlled by the data read from the monitoring server (Fig. 4). A simplified virtual hand model was chosen over a more realistic one to make the finger angles more obvious. Also, the simplified model contains fewer polygons, which is important for the performance of Java3D applets running in a browser. The monitored patient could be easily chosen from a select list in the applet window.

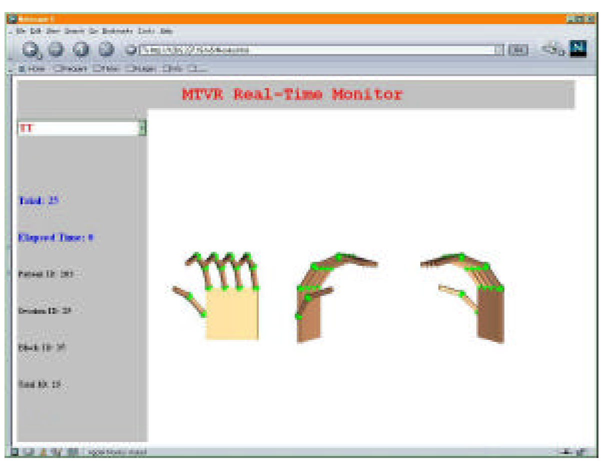

Figure 4: Web monitoring applet. © 2001 Rutgers University.

The main advantage brought by the web-based monitoring portal is the flexibility offered to the therapist. Any computer with Internet access can serve as monitoring station. The therapist can also monitor multiple patients simultaneously by either switching between them in one applet window or by opening a separate window for each.

To test the performance of the web monitor we used three computers with T1 Internet connection. The location of the monitoring server was in New Brunswick, New Jersey. The client workstations were located in Newark New Jersey. The results obtained are presented in Table 2.

Table 2 - Monitoring applet performance

Computer	Frame Rate
PentiumII 400MHz, 256MB RAM	6 fps
PentiumII 700MHz, 256 MB RAM	13 fps
Dual PentiumIII 833MHz, 256 MB RAM	27 fps

5. Conclusions

A shared VE telerehabilitation system was designed to support real-time communication and remote interaction between patient and therapist. Each site has a telerehabilitation workstation with a videocamera and an RMII force feedback glove. Both users can control a virtual hand and interact hapticly with virtual objects. The shared sense of space and presence allow patient-therapist interactions mediated by haptic devices. The system allows the therapist to apply remote physical therapy and collect patient data. Several experimental telerehabilitation scenarios were successfully tested in a LAN.

A Web-based approach to "real-time" patient telemonitoring - the monitoring portal for hand telerehabilitation – was also developed. The therapist interface is implemented as a Java3D applet that monitors patient hand movement. The web-based portal has less functionality than *SVRR* but offers portability and flexibility advantages. The therapist can monitor multiple patients simultaneously.

Acknowledgments

Research reported here was supported by grants from the National Science Foundation and from the New Jersey Commission on Science and Technology.

References

[1] Bouzit M., G. Burdea, G. V. Popescu, and R. Boian. 2000. "The Rutgers Master II-ND Force Feedback Glove", submitted to *IEEE Transactions on Mechatronics*, July.

[2] Burdea, G., Popescu, V., Hentz, V., Colbert, K. 2000. "Virtual Reality-Based Orthopedic Telerehabilitation", *IEEE Transactions on Rehabilitation Engineering*, Vol. 8, No. 3, pp. 430-432.

[3] Deutsch, J., J. Latonio, G. Burdea and R. Boian. 2001. "Post-Stroke Rehabilitation with the Rutgers Ankle System - A case study," *Presence*, MIT Press, Vol. 10(4), pp. 416-430, August 2001.

[4] Girone, M. J., G. C. Burdea, M. Bouzit, V. Popescu, and J. E. Deutsch. 2000. "Othopedic Rehabilitation Using the 'Rutgers Ankle' Interface," In *The Proc. of Medicine Meets Virtual Reality 2000*.

[5] Internet2. 2000. [Online]. Available: http://www.internet2.edu. [2001, October 15].

[6] Peifer, J., A. Hooper and B. Sudduth. 1998. "A Patient-Centric Approach to Telemedicine Database Development," *Proceedings of Medicine Meets Virtual Reality 6*, IOS Press, Amsterdam, pp. 67-73, 1998.

[7] Popescu, V., G. Burdea, M. Bouzit and V. Hentz MD. 2000. "A Virtual Reality-based Telerehabilitation system with Force Feedback," *IEEE Transactions on Information Technology in Biomedicine*, vol. 4, No. 1, pp. 45-51.

[8] Popescu G. V. 2001. *Design and Performance Analysis of a Virtual Reality-based Telerehabilitation System*, Ph.D. Thesis, Rutgers University, January.

[9] Trepagnier, C., M. Rosen, C. Lathan. 1999. "Telerehabilitation and Virtual Reality Technology for Rehabilitation: Preliminary Results." *CSUN's Conference Proceedings*. [Online]. Available: http://www.dinf.org/csun_99/session0241.html [2001, October 15].

[10] Sense8 Co., WorldToolKit User's Manual, Sausalito, CA, 1994.

[11] Sowizral, H., K. Rushforth, M. Deering. 1997. *The Java 3D API Specification.* Addison-Wesley.
[12] WhitePine, CuSeeMe User manual, 1998.