

Haptic Issues in Virtual Environments

Grigore C. Burdea

CAIP Center, Rutgers University,

96 Frelinghuysen Rd., Piscataway, New Jersey USA.

E-mail: burdea@vr.rutgers.edu

Abstract

Haptics is a recent enhancement to virtual environments allowing users to “touch” and feel the simulated objects they interact with. Current commercial products allow tactile feedback through desktop interfaces (such as the FEELItä mouse or the PHANToMä arm) and dextrous tactile and force feedback at the fingertips through haptic gloves (such as the CyberTouchä and the CyberGraspä).

Haptics VR programming requires good physical modeling of user interactions, primarily through collision detection, and of object response, such as surface deformation, hard-contact simulation, slippage, etc. It is at present difficult to simulate complex virtual environments that have a realistic behavior. This task is added by the recent introduction of haptic toolkits (such as Ghostä or VPS).

Current technology suffers from a number of limitations, which go beyond the higher product cost of haptic interfaces. These technical drawbacks include the limited workspace of desktop interfaces, the large weight of force feedback gloves, the lack of force feedback to the body, safety concerns, etc. Not to be neglected is the high bandwidth requirement of haptics that is not met by current Internet technology. As a result it is not possible at present to have a large number of remote participants interacting haptically in a shared virtual space.

1. Introduction

Haptics is a recent enhancement to virtual environments allowing users to “touch” and feel the simulated objects they interact with. This sensorial channel complements the usual visual (graphics) and sound feedback modalities used in current VR simulations [2]. The resulting complex system is more expensive than present PCs, but simulations with haptics are more realistic and more useful.

The haptic feedback modality groups tactile feedback and force feedback. Tactile feedback allows users to feel

the rugosity of virtual surfaces, their edges, temperature, or slippage. Force feedback reproduces the weight of grasped virtual objects, their mechanical compliance, inertia, as well as motion constraints. As opposed to tactile feedback, force feedback can actively prevent a user from moving into a restricted simulation space. However, force feedback lacks the rich contact surface information associated with tactile feedback. Certain high-bandwidth haptic interfaces, such as the PHANToM [9] combine tactile and force feedback, better addressing the requirements of realistic virtual environments [8].

Section 2 describes the haptic interfaces that mediate communication between the computer and the user. A review of the major components of physical modeling and available programming toolkits is included in Section 3. Section 4 is an analysis of the various limitations of current haptic feedback technology. Concluding remarks are given in Section 5.

2. Commercial Haptic Interfaces

The block diagram in Figure 1 [2] depicts a VR system which includes a haptic interface. The interface mediates a bi-directional energy exchange between the user and the computer (unlike visual feedback which is unidirectional). The user applies forces on, or changes the position of the haptic interface. This data is transmitted to the interface controller at very high rates. Here it is processed and sent to the host computer running the graphics rendering program. In response to changes in the virtual environment the host computer sends position and feedback forces to the interface. Eventually the forces are felt by the user.

It is important to understand that the human sensorial characteristics impose much faster refresh rates for the haptic feedback loop than for visual feedback. Computer graphics has for many years contended itself with low scene refresh rates of 20 to 30 frames/sec. In contrast, tactile sensors in the skin respond best to vibrations of 200 to 300 Hz or higher. This order-of-magnitude difference between haptics and vision bandwidths requires that the haptic interface incorporate a dedicated controller, such as

an embedded PC. The host computer processing the graphics high-level data is usually a multiprocessor machine (either a PC or a workstation). One of its processors is dedicated to the input/output and modeling needs of virtual environments with haptics.

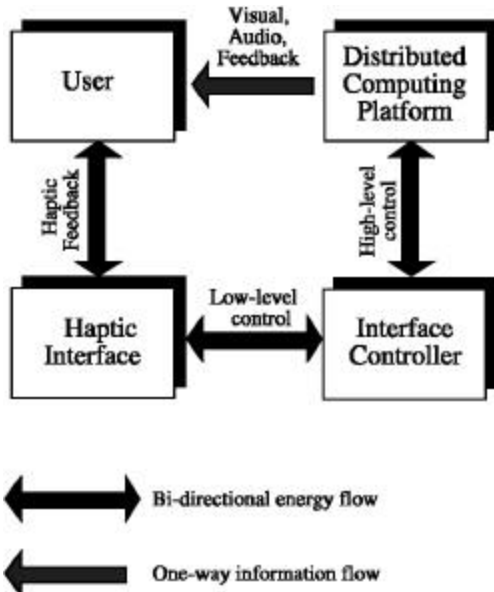


Figure 1. VR system with haptic feedback [2]. © Wiley 1996. Reprinted by permission.

There are many ways to classify a haptic interface depending on the type of actuators used, the application area, or the grounding arrangement. Today's haptic interfaces use mostly electrical actuators, because of their ease of installation and cleanliness, with hydraulic and pneumatic interfaces being less common. This paper will discuss general-purpose haptic interfaces, as opposed to those developed for a restricted application domain (such as for surgical training).

Since force feedback interfaces apply forces to the user, they need to be attached or "grounded" to an immovable support. This allows the interface to resist the actions of the user, even stopping him through large feedback forces. General-purpose commercial interfaces used today are either grounded to a desk (desktop) or to parts of the body (tactile and force feedback gloves).

2.1. Desktop haptic interfaces

An example of a desktop tactile feedback interfaces is the FEELit™ Mouse produced by Immersion Corporation (San Jose CA), shown in Figure 2 [6]. This is a two degrees-of-freedom mouse that has a 2.5×1.9 cm² workspace in which the user can feel simulated objects. For example, if the arrow mapping the mouse position to the screen traverses a surface, then the user can feel its roughness. Similarly the user can push with the arrow into

curves and determine their elasticity. Interaction through menus and buttons can be improved by making graphical symbols "magnetic," thus attracting the mouse arrow.



Figure 2. The FEELit™ mouse tactile interface [6]. Reprinted by permission.

An example of desktop force feedback interface is the PHANToM™ arm produced by SensAble Technologies (Cambridge MA), and illustrated in Figure 3 [9]. This is a weight counterbalanced and back-driveable arm that has the workspace of the user's wrist. It can apply 10 N maximum force (1.5 N continuous force) at an attachment where the user places his index finger. The arm low inertia and static friction, combined with its very high control bandwidth (1,000 Hz) result in an extremely crisp haptic feedback.

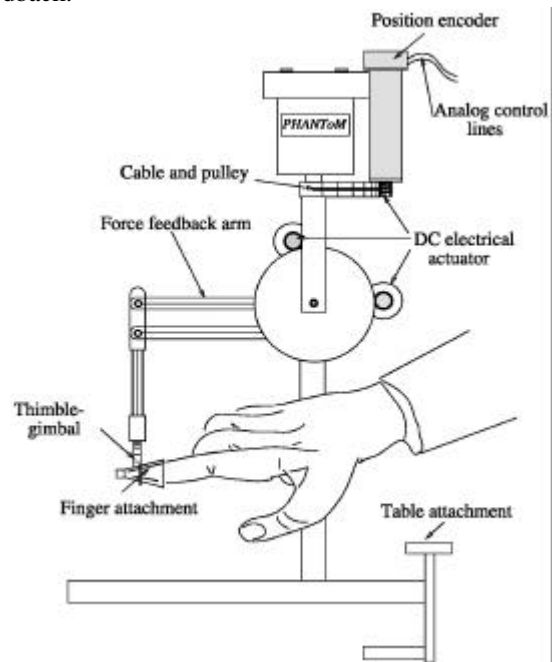


Figure 3. The PHANToM™ arm force feedback interface [9]. © ASME 1994. Reprinted by permission.

2.2. Wearable haptic interfaces

Wearable interfaces are those grounded to the user's body, giving him a much larger work volume than desktop haptic interfaces do. These are haptic gloves that allow users to interact with virtual environments through natural hand gestures while receiving feedback at the finger level. Haptic gloves, whether for tactile or force feedback, are most useful in simulations involving dexterity.

An example of a tactile feedback glove is the CyberTouch™ produced by Virtual Technologies (Palo Alto CA), shown in Figure 4-a [21]. It consists of a 18-sensor position measuring CyberGlove™ on which are mounted six small vibro-tactile actuators (on the back of each finger and in the palm). This array of tactile actuators can vibrate at frequencies of up to 125 Hz, being controlled by an electronic unit, which in turn communicates with the host computer.



a)



b)

Figure 4. a) The CyberTouch™ glove; b) The CyberGrasp™ glove [21]. Reprinted by permission.

The CyberTouch™, which weights about 200 grams is lighter than the CyberGrasp™ force feedback glove, which weighs 350 grams. As shown in Figure 4-b, the interface uses the same CyberGlove™ to measure the user's hand gesture, but the vibro-tactile actuators are replaced by an exoskeleton. This complex structure on the back of the hand guides tendons transmitting forces produced by electrical motors in a control box. Each finger can thus

feel up to 12 N continuous forces during the closing of the hand, but not during the hand opening.

3. Physical Modeling

The control of haptic interfaces is based on the modeling of interactions taking place in the virtual environment. As illustrated in Figure 5 [2], the first step in physical modeling of virtual objects is collision detection. This is followed by surface deformation and compliance, hard contact simulation and motion constraints.

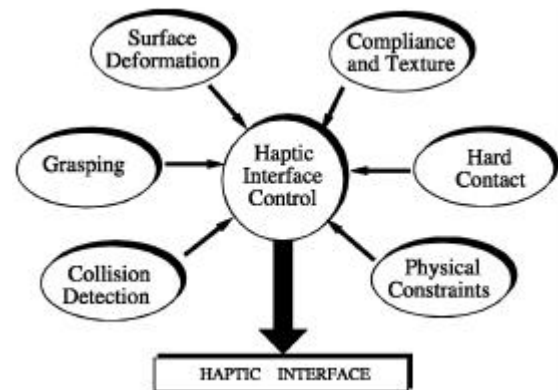


Figure 5. The elements of physical modeling of virtual objects [2]. © Wiley 1996. Reprinted by permission.

3.1. Collision detection

Collision detection is the first step in the haptic rendering loop. While bounding box algorithms are fast, their lack of accuracy poses problems when detailed contact information is necessary. In the case of dextrous manipulation of virtual objects by a virtual hand mapped to the user's haptic glove, the simulation needs to provide information on individual fingers contact. In such cases, as well as for large numbers of virtual objects, one approach is to prune the pairs of interacting objects with a fast bounding box algorithm, followed by a more accurate (and slower) local collision detection. Several algorithms exist, using Voronoi volumes [15], oriented bounded boxes and reconstruction [3], or voxel maps [11].

It is at present difficult to simulate complex virtual environments that have a realistic behavior. The approach taken by McNeely and colleagues at Boeing Co. is to extract a point shell model from the voxel map of dynamic objects. This corresponds to one point for each surface voxel of that object, as illustrated in Figure 6. In this way both speed and accuracy are maintained as long as the voxel has the appropriate size, and only one dynamic object (in this case a tea pot) exists.

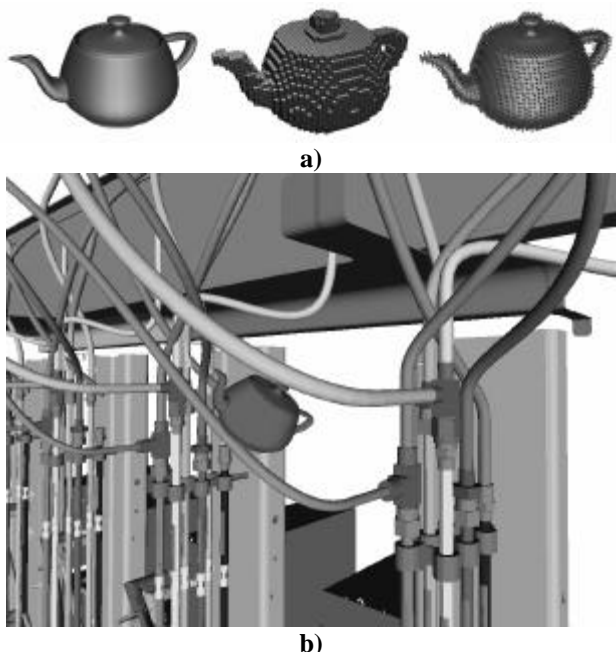


Figure 6. Voxmap and point shell collision detection: a) the point shell model; b) the complex virtual environment [11]. © ACM. Reprinted by permission.

3.2. Surface deformation

In the above example both the teapot and the pipes are considered rigid. In many instances virtual objects that need to be simulated have either elastic or plastic surface deformation. In this case algorithms can apply either a vertex-based method, if surfaces are polygonal meshes, or spline-based methods, for objects with parametric surfaces.

In a vortex-based surface deformation approach the user can interactively change the location of a vertex, thus changing the shape of the polygons sharing it. Certain connectivity laws can be implemented, such that the change in position for one vertex is transmitted to a certain degree to its neighbors. Figure 7 [16] illustrates how the above method is applied for a virtual hand squeezing an elastic virtual ball. A reference shape is first defined, consisting of the undeformed ball, followed by a global and a local deformation steps. The global deformation during grasping and squeezing is implemented using a morphing technique. Local deformation is controlled by the distances between the fingertip mesh and the closest vertices within the fingertip bounding box.

In a spline-based surface deformation method, surfaces are considered extensions of parametric three-dimensional curves (such as bi-cubic surfaces) [5]. This method uses less storage, and provides increased surface smoothness compared to polygonal meshes. Users do not interact directly with the surface, rather they move control points,

which in turn results in a change in surface shape. The Direct Free Form Deformation approach developed by Yamashita and colleagues for VR [23] allows users to select a point on the object surface and then indicate the desired change in surface shape. Control lattice candidates are selected based on a least-squares solution, from the many possible candidate sets.

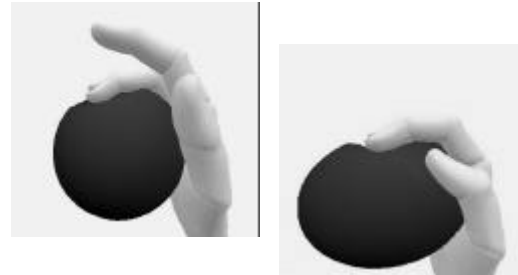


Figure 7. Virtual ball squeezing using global and local deformation [16]. © 1999 IEEE. Reprinted by permission.

3.3. Force feedback modeling

In order to assure simulation realism it is necessary to compute the contact forces present during object deformation and then apply these forces on the user. Furthermore, the contact forces have to be computed at very high rates, in order to satisfy the human tactile sensing requirements. This is a significant computational load, solved through distributed computing and simplified modeling laws. The most common approach is to use the Hooke's law, as illustrated in Figure 8 [1].

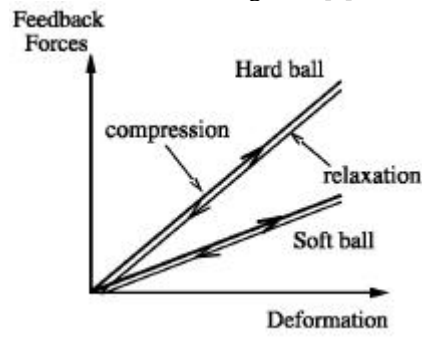


Figure 8. Contact force modeling for elastic deformation [1]. © Etitions Hermes. Reprinted by permission.

The applied force grows linearly with the amount of surface deformation, up to the maximum output of the haptic interface. The slope of the force curve is proportional with the stiffness of the modeled object, in this case of soft and hard elastic virtual balls. As seen in this graph, forces are present during both compression and relaxation. If the grasped object is plastic, then contact

forces drop to zero as soon as the finger loses contact with the deformed surface, i.e. forces are present only during compression.

Sometimes a modeled object has a more complex structure, such as a softer outer volume and a harder interior. In this case the initial compression follows a smaller slope, with a discontinuity and steeper forces when the interior kernel is felt. Another example is the force pattern associated with haptic push buttons, as illustrated in Figure 9 [18]. Here forces grow linearly until a threshold is reached, after which they drop to zero. When the limit of translation motion allowed by the button mechanical stops is reached, forces grow very fast, simulating contact with the hard back wall. This results in a haptic “click.”

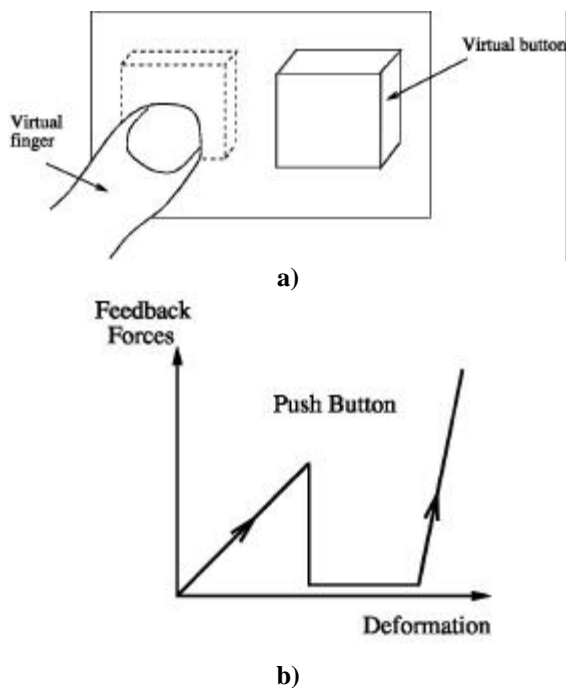


Figure 9. Contact force modeling for a virtual pushbutton: a) the virtual environment; b) the corresponding force pattern [18]. Reprinted by permission.

Surface texture is another important part of physical modeling, allowing virtual objects to be characterized as smooth, rough, slippery, etc. One approach to model the surface mechanical bumps is to use local surface gradients in the direction of the surface normal. The small forces felt by the user are proportional to the height of the hills shown in Figure 10 [12]. Another approach is to use high-frequency sinusoidal functions, as vibrations superimposed over the force-feedback signal. This approach is taken by SensAble Technologies for the PHANToM™ arm.

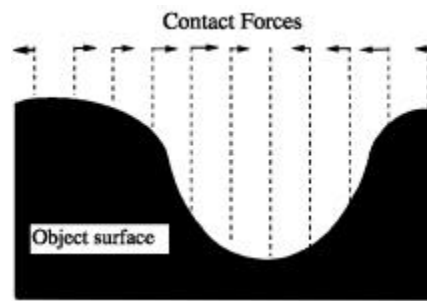


Figure 10. Surface smoothness modeling [12]. © Etitions Hermes. Reprinted by permission.

3.4. Haptics toolkits

Haptic toolkits have been developed to help the programmer integrate this important sensorial channel in the VR simulation. These toolkits are either extensions of existing VR libraries, or stand-alone. In the first category falls the haptic extension of WorldToolKit™, a 1,000 function library designed for VR developers [4]. This provides support for collision detection, dynamic object behavior and drivers for the CyberGlove™ and other i/o devices.

The GHOST™ SDK is a specialized library developed by SensAble Technologies for the PHANToM™ haptic interface [19]. This library allows programmers to specify object geometry, properties, and global haptic effects, using a haptic scene graph. GHOST™ automatically computes the interaction forces between a haptic point (mapped to the user’s index fingertip), and objects or effects within the virtual environment. It can also simulate object compliance and friction, as well as springs, impulses and vibrations. Unlike WorldToolKit™ however, it does not generate a graphics scene, which needs to be developed separately by the programmer, and coupled with the force feedback loop controlled by the GHOST™ commands.

Virtual Technologies has recently developed the VirtualHand® Toolkit which gets position measurements from the CyberGlove® and generates the graphical image of a hand mapped to the user’s gestures. It provides a high-level API for the CyberGrasp including effects such as jolts and pulses, as well as tactile feedback control for the CyberTouch glove.

4. Limitations of Current Haptic Technology

Current technology suffers from a number of limitations, which go beyond the higher price of haptic interfaces. These technical drawbacks include the limited workspace of desktop interfaces, the large weight of force feedback gloves, the lack of force feedback to the body, safety concerns, etc.

4.1. Large haptic interface weight

One of the major drawbacks of haptic gloves is their large weight. The CyberGrasp at almost 400 grams is tiring for the user during prolonged simulations. The large weight is due to metal necessary to support the cables at the back of the hand, and to electrical actuators which have very low power/weight ratio. A new version of the glove, called the GraspPack™ features a backpack allowing the user to carry up to two electronic interfaces (for gloves worn on the right and left hands) [22]. While this arrangement, shown in Figure 11, gives users more freedom of motion, it also increases the weight they have to carry.



**Figure 11. The GraspPack™ extension to the CyberGrasp haptic glove [22].
Reprinted by permission**

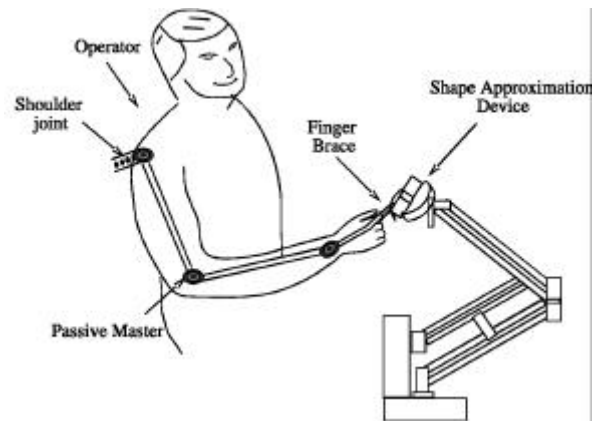
A solution to the above problem requires actuators that have much larger power output, at much lower mass. Such actuators should also be more compact, allowing a “direct drive arrangement” in the hand. A haptic glove prototype using these kind of actuators is the Rutgers Master II-ND shown in Figure 12 [17]. Unfortunately, pneumatic actuators such as those of the RMII-ND glove have low bandwidth (tens of Hz), which may be problematic in certain applications. Furthermore, some portion of the hand workspace is lost due to the placement of the actuators in the palm.

Another way to address the interface weight problem is to use desktop, as opposed to wearable interfaces. In this case the weight of the interface is supported by the desk. The drawback in this case is a significant reduction in the user’s work volume.



Figure 12. The Rutgers Master II-New Design haptic glove [17].

A way to provide force feedback without tiring the user is to apply forces to the hand using a robot, an approach pioneered by McNeely at Boeing Co. [10] and Tachi and colleagues at the University of Tokyo [20]. As illustrated in Figure 13, the user wears a passive position measuring exoskeleton, while the robot moves a turret called a “shape approximation device.” Contact is made only when there is contact in the virtual environment, and the turret changes configuration to present to the user a flat surface, an edge, or other contact surface information. Several drawbacks exist, namely cost, limited dexterity (only one contact point exists), and user’s safety.



**Figure 13. Haptic feedback using a robot [20].
© IEEE 1994. Reprinted by permission.**

In the above systems force feedback is provided at the hand. In locomotion simulations it is necessary to provide force feedback to the user’s feet, for such applications as walking, running, driving simulators, etc. The approach used initially was that of motorized treadmills allowing walking in place. More recently Noma and his colleagues at the Advanced Technology Research Laboratory (Japan) developed the ground surface simulator shown in Figure

14 [13]. It consists of vertically moving small tiling panels and actuator units. The motion of the user is tracked by a vision system that transmits the data to a host computer running the simulation. The host communicates with the actuator controllers which change the orientation of the active tiles to simulate smooth or bumpy surfaces.

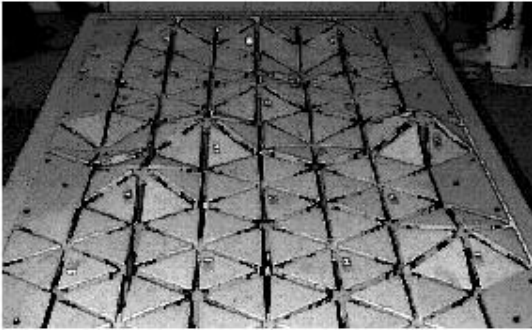


Figure 14. The ground surface simulator [13].
© IEEE 2000. Reprinted by permission.

In view of the above problems with heavy and bulky actuators, there is a lack of haptic suits on the market. Users can either have force feedback to the hand, or to the feet, but not to the whole body. While graphics libraries exist to simulate avatars in virtual environments, it is not possible to feel contact forces the virtual body experiences in other areas besides the hands or feet.

4.2. Tracker errors in large workspaces

A problem with vision-based tracking is the requirement for direct line of sight between the camera and the object it is tracking. If the simulation does not require precision, then a vision-based approach may suffice. However, if the simulation involves interaction at the finger level, then vision tracking is not the best approach. An alternative is to use long-range magnetic trackers, such as the Polhemus Long Ranger™ system.

An instance of tracking over large volumes is the interaction with graphics projected by large displays, as illustrated in Figure 15. Here the user wearing a haptic glove views stereo images on a BARCO Baron 3D. The display has metal which interferes with the accuracy of the Polhemus tracker worn on the user's wrist. Further errors are introduced by the metal present in the laboratory walls, or ceiling! It is thus very difficult to interact with the virtual objects directly, due to calibration errors. Instead the user's hand is mapped to a virtual hand, which in turn interacts with the virtual objects (in this case virtual balls of different color and compliance). The user adapts to the unintended offsets in position and orientation of the virtual hand, but the naturalness of the interaction is diminished. The solution is to either develop a very

complex calibration scheme, or to opt for a different type of trackers, such as the IS600 Mark2 recently introduced by InterSense [7]. It measures motion through a combination of solid-state accelerometers for orientations and ultrasonic sensors for translations. Neither of these techniques are subject to metallic interference, however ultrasonic sensors have other problems such as sensitivity to certain sounds, a smaller sampling rate than magnetic trackers and direct line of sight requirements.

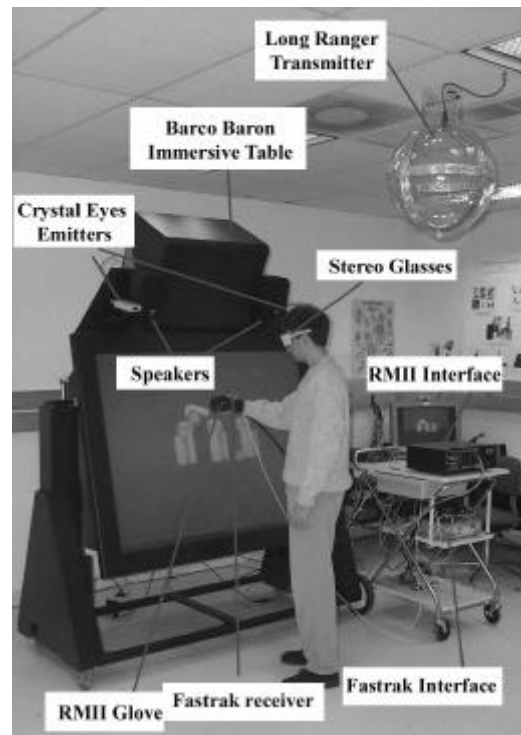


Figure 15. Large-volume haptic simulation

4.3. Bandwidth limitations

Not to be neglected is the high bandwidth requirement of haptics that is not met by current Internet technology. As a result it is not possible at present to have a large number of remote participants interacting haptically in a shared virtual space. The situation should improve with faster communication modalities such as Internet2, as well as embedded computing in all haptic interfaces which reduces the bandwidth requirements of real-time distributed haptics.

5. Conclusions

The present review is by necessity limited. It is hoped that the reader now understands the state of the art in haptic interfaces, its potential to improve VR simulations, and the many technical challenges that still exist.

Acknowledgements

The author's research reported here was supported in part by grants from the National Science Foundation (BES-9708020), from Office of Naval Research (N14-98-1-0264) and from Rutgers University (CAIP and SROA grants).

References

- [1] G. Burdea, "Virtual Reality Systems and Applications," *Electro'93 International Conference*, Short Course, Edison NJ, April 1993.
- [2] G. Burdea. *Force and Touch Feedback for Virtual Reality*, John Wiley & Sons, New York, 1996.
- [3] Z.L. Cai, J. Dill, and S. Payandeh, "Haptic Rendering: Practical Modeling and Collision Detection," *Proc. of ASME WAM*, DSC-Vol. 67, ASME, New York, 1999, pp. 81-86.
- [4] Engineering Animation Inc. "The Sense8 Product Line," www.sense8.com.
- [5] J. Foley, A. van Dam, S. Feiner, and J. Hughes, *Introduction to Computer Graphics*, Addison-Wesley Publishing, Menlo Park CA, 1994.
- [6] Immersion Corporation. "FEELit™ Mouse," Technical document, San Jose CA, 12 pp., October 1, 1997. Electronic version: www.immerse.com.
- [7] InterSense Co "IS600 Mark 2 Overview," Burlington MA. Electronic version: www.isense.com/is600.htm.
- [8] D. Kontarinis and R. Howe, "Tactile Display of Vibratory Information in Teleoperation and Virtual Environments", *Presence – Teleoperators and Virtual Environments*, Cambridge, 1995, pp. 387-402.
- [9] T. Massie and K. Salisbury, "The PHANToM Haptic Interface: A Device for Probing Virtual Objects," *Proceedings of ASME WAM*, DSC-Vol. 55-1, 1994, pp. 295-300.
- [10] W. McNeely, "Robotic Graphics: A new approach to force feedback for virtual reality," *Proceedings of IEEE VRAIS'93 Symposium*, Seattle WA, 1993, pp. 336-341.
- [11] W. McNeely, K. Puterbaugh and J. Troy, "Six Degree-of-Freedom Haptic Rendering Using Voxel Sampling," *Computer Graphics Proceedings*, ACM, August 1999, pp. 401-408.
- [12] M. Minsky, M. Ouh-young, O. Steele, F. Brooks Jr. and M. Behensky, "Feeling and Seeing: Issues in Force Display," *Computer Graphics*, ACM, 24(2), 1990, pp. 235-243.
- [13] H. Noma, T. Sugihara and T. Miyasato, "Development of Ground Surface Simulator for Tel-E-Merge System", *Proceedings of IEEE Virtual Reality 2000*, IEEE, 2000, pp. 217-224.
- [14] Polhemus Inc., "Long Ranger," Colchester VT, www.polhemus.com/lrangeds.htm
- [15] M. Ponamgi, D. Manocha, and M. Lin, "Incremental Algorithms for Collision Detection between Solid Models," *Proceedings of ACM/SIGGRAPH Symposium on Solid Modeling*, ACM, New York, Date, pp. 293-304.
- [16] V. Popescu, G. Burdea, and M. Bouzit, "Virtual Reality Simulation Modeling for a Haptic Glove," *Computer Animation'99*, IEEE, May 1999, pp. 195-200.
- [17] Rutgers University, CAIP Center, "The Rutgers Master II- ND," Piscataway NJ, 1999. Electronic version: www.caip.rutgers.edu/vrlab/rm2.html.
- [18] SensAble Technologies, "PHANToM Master User's Manual," Cambridge MA, 1994. Electronic version: www.sensable.com.
- [19] SensAble Technologies, "GHOST SDK Programmer's Guide," Cambridge MA, 1998.
- [20] S. Tachi, T. Maeda, R. Hirata, and H. Hashino, "A construction method of virtual haptics space," *Proceedings ICAT'94*, Tokyo, Japan, 1994, pp. 131-138.
- [21] Virtual Technologies, "CyberTouch, and CyberGrasp" Company brochures, Palo Alto CA, 1998. Electronic version: www.virtex.com.
- [22] Virtual Technologies, "GraspPack," Company brochure, Palo Alto CA, 1999. Electronic version: www.virtex.com/products/hw_products/grasppack.html.
- [23] J. Yamashita, H. Yokoi, Y. Fukui, and M. Shimojo, "development of Virtual Shape Manipulation System," *Proceedings of ICAT'94*, Tokyo Japan, 1994, pp. 35-42.