Virtual-Virtual Haptic Feedback and Why it Wasn't Enough

Dru Clark, Mike Bailey[1]
University of California San Diego

ABSTRACT

Traditional interference detection for visualization has taken a virtual-virtual approach, that is, both the intersector and the intersectee are virtual geometries. But, we have learned that there are advantages in combining both physical models and virtual models in the same space. The physical model has many properties that are difficult to mimic in an all-virtual environment. A realistic interaction is achieved by casting the physical model as a 'twin' to the virtual model. The virtual twin has the ability to interact with other virtual models in software. The two combined into a single system allow for a more effective haptic visualization environment than virtual interaction alone.

Keywords: haptic, collision-detection, virtual reality

1. INTRODUCTION

There are many applications in scientific and engineering visualization that need interference, or collision, detection. In chemistry, for example, collision detection is used to see how closely two molecules come to physically docking. In engineering, collision detection is used to see if and how well mechanical parts will fit in an assembly.

Originally, visualization of "collision detection" was strictly a visual operation. Parts were animated on a graphics screen and the user had to very carefully look at the graphics representation to see if one part went inside another. Some simple techniques with clipping planes were introduced, but they were not general enough to solve the whole problem.

Solid modeling systems emerged from the CAD world as a way to exactly solve the collision detection problem. Because solid modeling CSG (constructive solid geometry) systems were made to perform Boolean operations on multiple solid objects, the objects in question could be intersected and the resulting overlap volume could be displayed. This was computationally accurate and general enough, but too slow a solution for interactive visualization. Also, forcing the 3D objects to be legal solids was overly limiting.

Recently, the focus of collision detection has been on fast surface-surface intersections. [GUPTA97] used a unified physically-based model to simulate interactions between surfaces. [JAYARAM99] created VADE, a Virtual Assembly Design Environment, with fast surface-surface comparisons. [SALISBURY95] used fast surface-surface comparisons to drive the PHANTOM force-feedback system.

2. VIRTUAL-VIRTUAL HAPTIC FEEDBACK, AND WHY IT WASN'T ENOUGH

This project started out as a virtual-to-virtual haptic project. Haptic feedback means that something mechanism is taking advantage of the human sense of touch. Haptic does not imply that there is necessarily a force pushing back, it just means that there is some touch-sensitive feedback mechanism. The project's goals were to create insight into 3D dimensional engineering and visualization interference

[1] {dru@sdsc.edu, 858-822-9019}, {mjb@sdsc.edu, 858-534-5142}, San Diego Supercomputer Center, 9500 Gilman Drive, La Jolla, CA 92093-0505

situations by applying haptic feedback. With the aid of the CyberTouch glove's finger vibrators, one would "feel" the contact as an object was "held" in one's hand. This proved to be complicated in development and difficult in application.

We obtained an Ascension Flock of Birds 3D tracker and a Virtual Technologies CyberTouch glove. The idea was to use the glove and a tracker to virtually "grab" a part and virtually "fit" it into, through, or around another part.

This was an interesting exercise in system integration along with some elaborate graphics programming. In the development of the system a number of methods were explored to address virtual interaction. This led to an environment where a contact could be calculated and relayed to the user through the CyberTouch glove's vibrations on the fingers. If an object was gripped, then the vibrators would be driven at a certain frequency. Once contact was established, the object could be repositioned or reoriented within the environment. Contact information has to be layered to represent the initial gripping of the object as well as the part to part interaction. Some illustrations of the process are shown below:

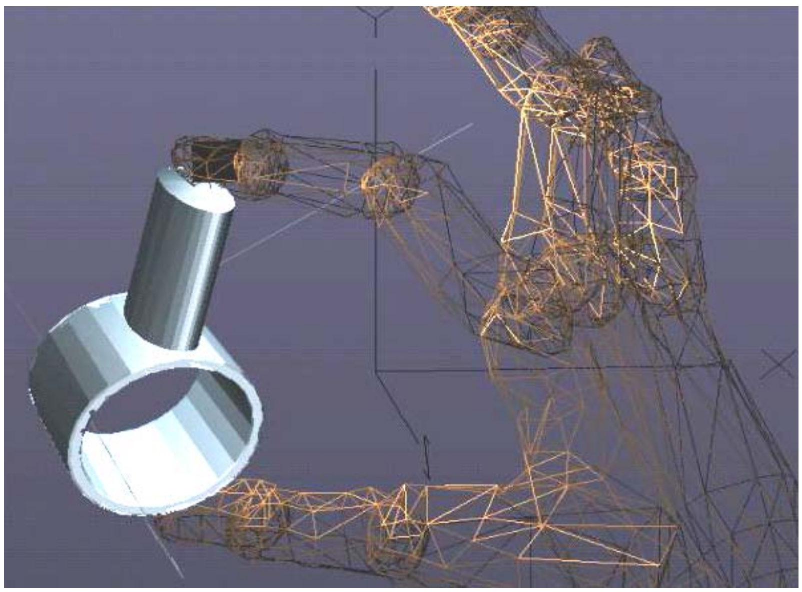

Figure 1: Virtual Hand "Holding" a Virtual Part

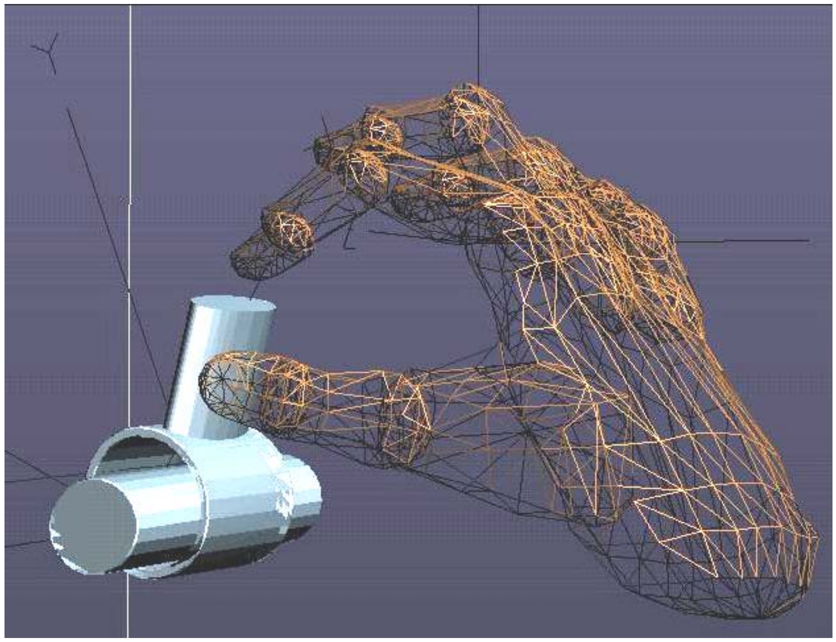

Figure 2: Virtual Hand Fitting Two Parts Together

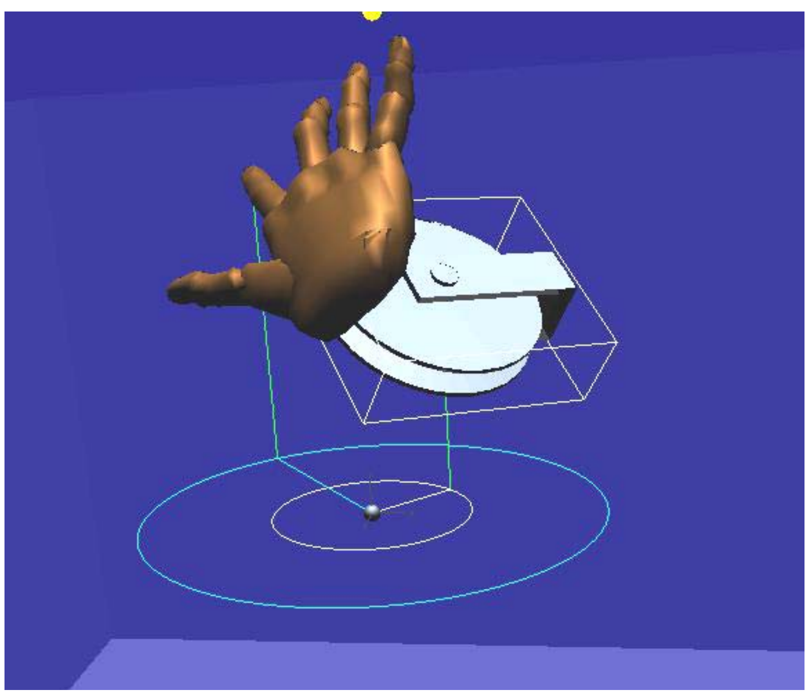

Figure 3: Virtual Hand Attempting to Grab a Pulley Part

Including the rendering of a virtual hand in the scene was needed to understand location.

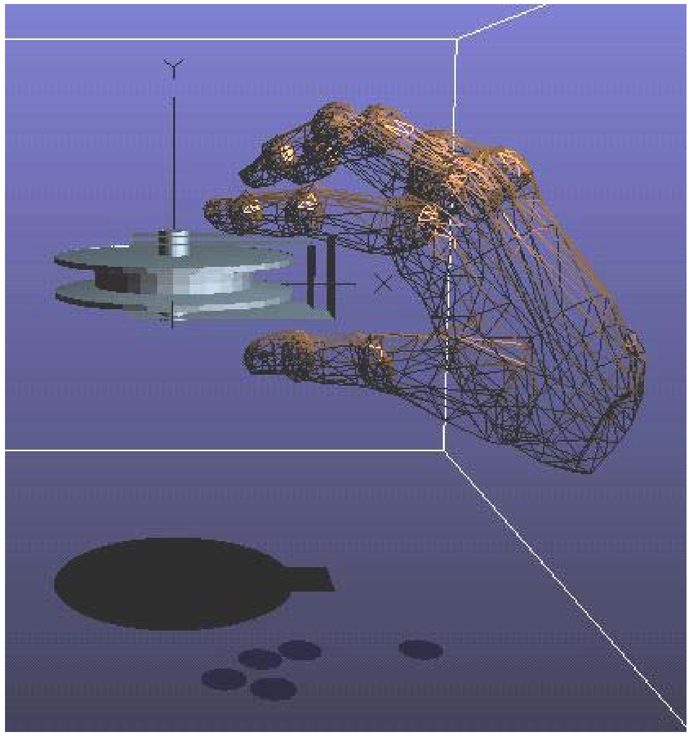

Figure 4: Virtual Hand Close to Grabbing the Pulley

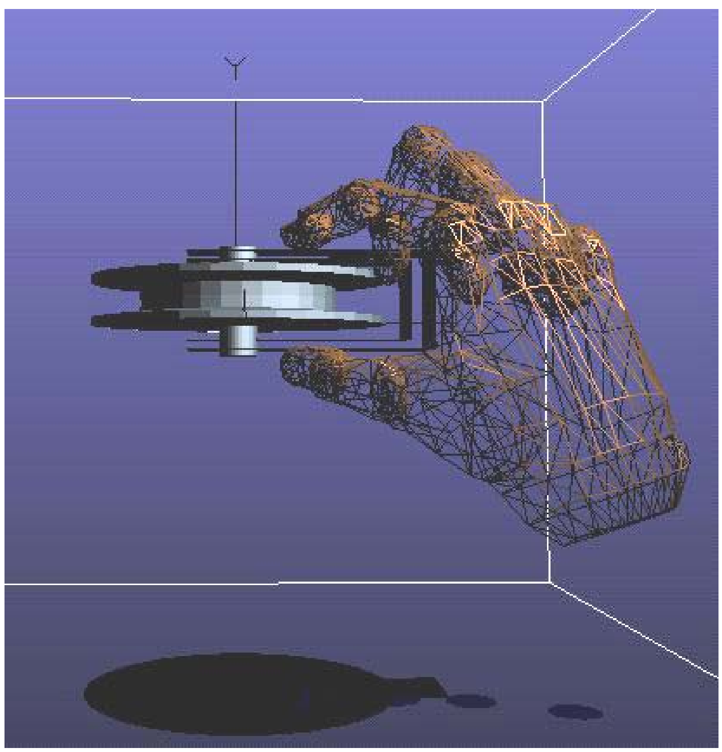

Figure 5: Virtual Hand Holding the Pulley

Including the rendering of a virtual hand in the scene was needed to understand location. Once the part was grabbed, the display of the hand model was less important.

But, while the results were reasonably good, it didn't "feel right". There seemed to be some barriers to good interaction that could not be overcome:

- It was too easy to get lost in 3D. Because everything was virtual, nothing was "grounded" in reality.
- It was too hard to "grab" the part. Even with the glove vibrators, it did not feel right. A user had little cognitive appreciation for how or where the part had been grabbed. This is consistent with other research results and has been shown to affect both the new and advanced users [JAYARAM99].
- The hand often obscured the vision of other parts and added to the complication of the scene.
- Realistic gripping of a virtual part is computationally expensive. The introduction of advanced algorithms has allowed realistic gripping of parts, but at a cost in computation time.
- Interacting with the virtual scene was then a problem because oftentimes we felt we had grabbed the part a certain way when, in fact, we had grabbed it in a different orientation. That made the resulting collision detection vibrations conflict with intuition.
- Virtual parts have none of the inertia or texture feel that we have come to use as motion cues in everyday life.
- It was all too far from reality. If one thinks of real-real intersections as "reality", then virtual-virtual is two steps removed. In short, this was too many levels of abstraction away from our everyday experiences.

The "lost in 3D" problem was the worst of the drawbacks. We conducted several experiments with creating different visual cues about where things were in relation to each other. Some of those tests are shown below:

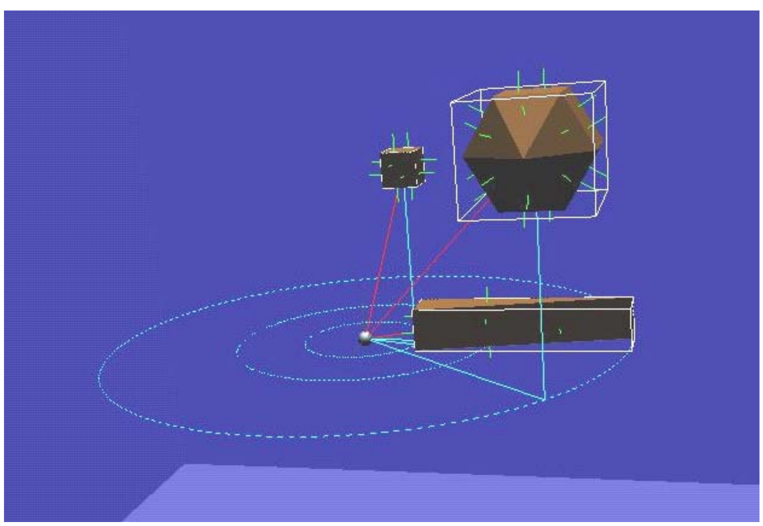

Figure 6: Visualizing Spatial Relationships to Assist Grabbing

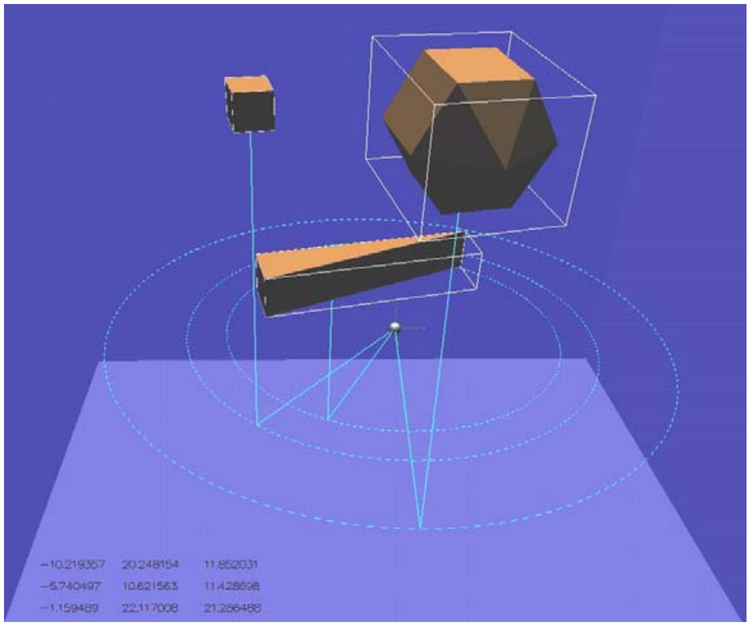

Figure 7: Visualizing Spatial Relationships to Assist Grabbing

While this helped, it still was too far abstracted from reality to be as useful as we had hoped.

3. VISUALIZATION THROUGH THE USE OF PHYSICAL MODELS

In 1995, the SDSC TeleManufacturing Facility (TMF) Project was started as a way to produce physical hardcopy to enhance visualization. This project was more successful than even we had envisioned it would be, as hundreds of visualization models have been made for researchers all over the country. (See: [BAILEY95, BAILEY96, BAILEY98, BAILEY99, CLARK97, SVILTIL98].)

Some of the molecular biologists with whom we had been working had used these real models to develop insight into molecular docking behaviors. [BAILEY98, SVITIL98] By colliding two physical models, one could see how well the molecules geometrically fit together. In effect, it was a custom haptic feedback system. If one thinks of the first attempt as being “virtual-virtual”, then this attempt was “real-real”.

But, this also posed problems that kept this from being a good general-purpose solution:

- It is not always practical to fabricate both components of the intersection. Sometimes, such as in CAD applications, the intersector might be a part, but the intersectee might be a large assembly that is too large for a fabrication machine.

- By losing the computer graphics, we would also lose the ability to display interesting feedback on the collision process. Such information as molecular docking charges and forces are easily represented graphically and are quite valuable in developing insight into what is happening in the virtual scene.

4. REAL-VIRTUAL INTERFERENCE DETECTION

This suggested a new and different approach to this problem: create a real-virtual situation where the intersector is a real physical model and the intersectee is virtual.

A physical model can be built from a CAD or visualization file in a matter of a few hours. This model can be made of plastic, paper/wood, powder, or wax depending on the type of prototyping machine used in the process. These physical models not only create a hard surface, they also contain a certain familiar mass and texture quality to them. The model can then be re-introduced back into the virtual system

A tracker can be embedded in the model. Any motion of the real model can then be used to drive the motion of the virtual "twin". All the usual calculations such as collision detection are treated as virtual encounters in software. Feelings of inertia or moment are covered by the physical properties of the fabricated model. The task of finding and grasping the part is simplified to picking up the model and interacting. The simulation is not delayed by the task of connecting with the object or maintaining contact. The virtual twin reflects this motion in real time.

When a second virtual part is encountered the two are checked for intersections. If a collision has occurred then the information is relayed back to the user through a vibration to the hand. The vibrators we used are on the CyberTouch glove, but could easily be placed on the part, which would remove the need for the glove.

The following figures show a real part and its virtual twin being inserted into a virtual assembly. Besides the vibration to the hand, the user also receives visual interference feedback by highlighting the polygons that are interfering.

4.1 Setup

This system was written in C++ on an SGI Irix 6.5 platform using OpenGL. The 3D tracker was an Ascension Flock of Birds. The hand input was from a Virtual Technologies CyberTouch glove. The CyberTouch glove also has vibrators on the fingers, which under program control, can give the user interference feedback.

Collision detection was handled by the RAPID libraries from the University of North Carolina at Chapel Hill [RAPID00]. These libraries fit in well with the polygon-based STL file format used in all rapid prototyping machines. A library was developed to handle the communications with the Flock of Birds motion tracking system. Libraries were also developed to handle the parsing of the STL file format.

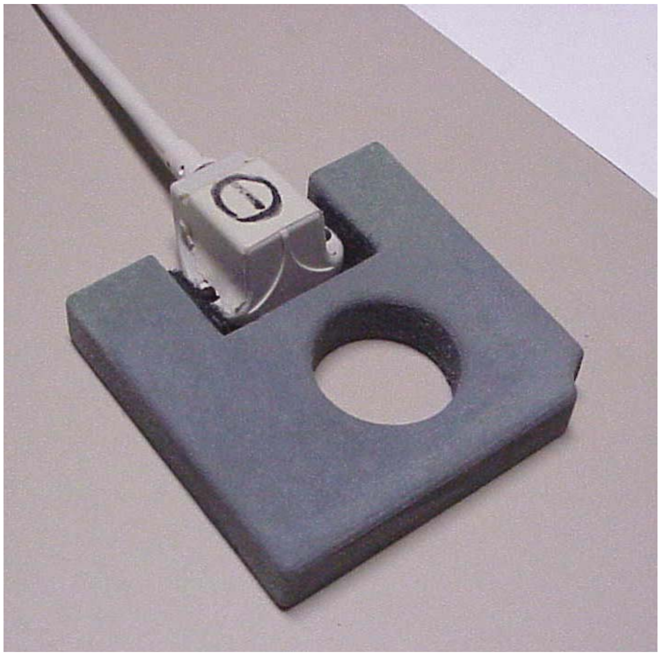

Figure 8: Tracker Attached to a Mechanical Part

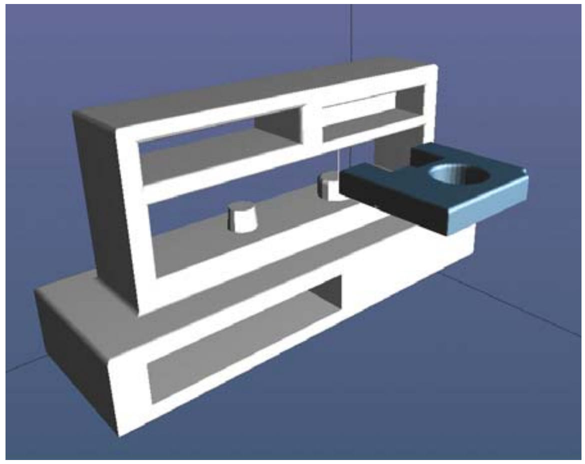

Figure 9: The Part Being Inserted into a Virtual Assembly

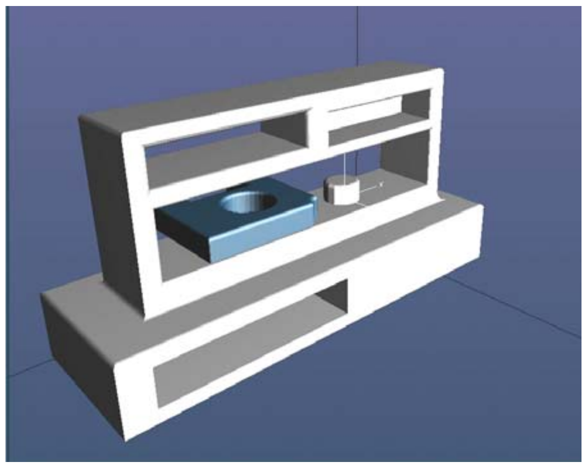

Figure 10: The Insertion Continues

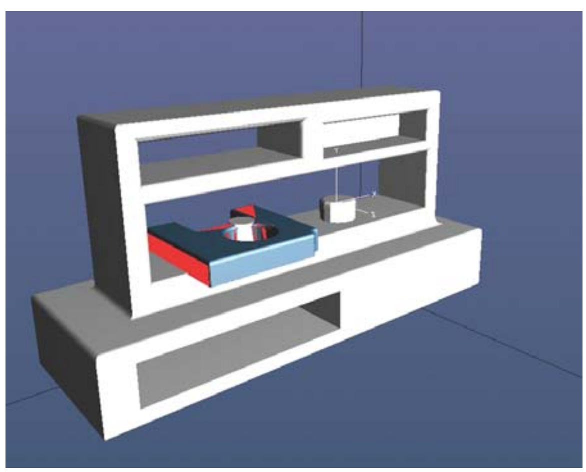

Figure 11: Interfering Polygons are Indicated by Color

Figure 12: The Real 3D Tracker

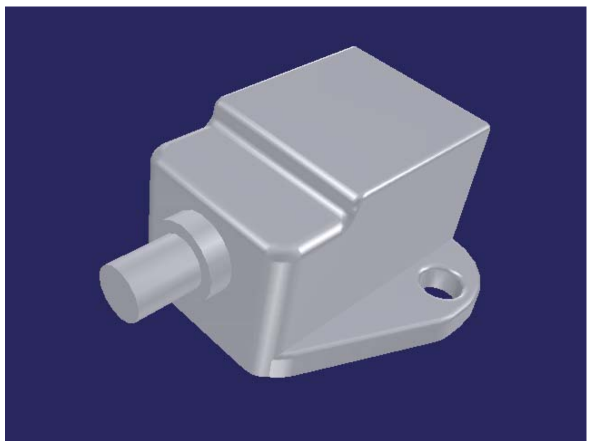

Figure 13: Virtual Model of the 3D Tracker

5. RESULTS

The real-virtual method overcame many of the problems we had seen with a pure virtual-virtual solution. Feeling lost in 3D happened much less often. The feeling of actually holding a part with its hardness, inertia, and surface texture feel was a much more comfortable and familiar environment than holding nothing and pretending it was something.

The 3D tracker, in effect, did all the work. Because its niche on the part was created digitally, there was no ambiguity what the 3D relationship was between the position of the tracker and the position of the part.

Because the location of the tracker is so important, we use solid modeling tools to create a niche for it in the part. We have found the CAD system IronCAD [IRONCAD00] to be an excellent tool for this. We also have used the TWIN solid modeling library [TWIN00] to write batch programs to create the niche automatically.

Figure 14: Niche Automatically Created to House the 3D Tracker

While the glove did not play the starring role we had originally thought, it did play an important part. Instead of a mechanism to track finger locations, it became a mechanism for supplying haptic feedback to the hand when an interference was detected. The feedback was a result of the position of the part, and thus, the position of the tracker. The user could grip the part any way he or she wanted to, and could even switch to a different gripping position in the middle of the operation without confusing the system. In fact, if the vibration feedback was attached to the part instead of the glove, the user could also switch hands.

6. CONCLUSIONS

Some day, force feedback hardware and software systems will be at such a state of realism that they will emulate all of the normal physical cues such as inertia, texture, and force feedback. Until that day, we must rely on tricks to come as close to that situation as we can. The real-virtual method for gaining insight into objects undergoing collision is one of those tricks. It is only one level of abstraction away from reality instead of two. It also provides inertia and surface texture cues that pure virtual systems miss. Perhaps most importantly, it does not depend on how the user holds the intersector part, and even allows the user to switch gripping positions in the middle of the operation with no side effects.

ACKNOWLEDGEMENTS

This work was funded by the National Science Foundation under grants MIP-9420099 and DBI-9602900. Thanks to Lynn Ten Eyck and Susan Taylor from UCSD and Art Olson from Scripps Research Institute for their insights into what makes a meaningful molecular visualization.

REFERENCES

[BAILEY95] Michael Bailey, "TeleManufacturing: Rapid Prototyping on the Internet," *IEEE Computer Graphics and Applications*, Volume 15, Number 6, November 1995, pp. 20-26.

[BAILEY96] Michael Bailey, "The Use of Solid Rapid Prototyping in Computer Graphics and Scientific Visualization," in SIGGRAPH '96 Course Note #37*: The Use of Touch as an I/O Device for Graphics and Visualization*, SIGGRAPH '96 Conference, New Orleans, LA, August 4-9, 1996.

[BAILEY98] Michael Bailey, Jack Johnson, and Klaus Schulten, "The Use of Solid Physical Models for the Study of Macromolecular Structure", *Current Opinion in Structural Biology*, Volume 8, Number 2, April 1998, pp. 202-208.

[BAILEY99] M. J. Bailey, "Manufacturing Isovolumes", *Proceedings of the International Workshop on Volume Graphics*, Swansea, UK, March 24-25, 1999, pp. 133-146.

[CLARK97] Dru Clark and Michael Bailey, "Visualization of Height Field Data with Physical Models and Photomapping", *Proceedings of IEEE Visualization '98*, October 1997, pp. 89-94.

[GUPTA97] Rakesh Gupta and Daniel Whitney and David Zeltzer. "Prototyping and Design for Assembly Analysis using Multimodal Virtual Environments", *Computer-Aided Des*ign, Volume 29, Number 8, 1997, pp. 585-597.

[IRONCAD00] hrrp://www.ironcad.com

[JAYARAM99] Sankar Jayaram, Yong Wang, Uma Jayaram, Kevin Lyons, and Peter Hart, "A Virtual Assembly Design Environment", *Proceedings of IEEE Virtual Reality '99*, March 13-17, 1999, Houston, TX.

[RAPID00] http://www.cs.unc.edu/~geom/OBB/OBBT.html

[SALISBURY95] K. Salisbury et al, "Haptic Rendering: Programming Touch Interaction With Virtual Objects", *1995 Symposium on Interactive 3D Graphics*, April 1995, pp. 123-130.

[SVILTIL98] Kathy A. Sviltil, "A Touch of Science", *Discover*, Volume 19, Number 6, June 1998, pp. 80-84.

[TWIN00] http://www.cadlab.ecn.purdue.edu/cadlab/twin