Virtual-SAP: An Immersive Tool for Visualizing the Response of Building Structures to Environmental Conditions

Doug A. Bowman[1], Mehdi Setareh[2], Marcio S. Pinho[3], Ndiwalana Ali[1], Alex Kalita[1], Yunha Lee[2], John Lucas[1], Matthew Gracey[1], Malini Kothapalli[1], Qinwei Zhu[1], Ameya Datey[1], and Pradeep Tumati[1]

[1]*Department of Computer Science, Virginia Tech, USA*
[2]*Department of Architecture, Virginia Tech, USA*
[3]*Instituto de Informática, UFRGS and Faculdade de Informática, PUCRS, Brazil*
{bowman, setareh, andiwala, akalita, ylee, jolucas, mgracey, mkothapa, qzhu, adatey, ptumati}@vt.edu
pinho@inf.pucrs.br

Abstract

Immersive virtual environments (VEs) offer interactive, real-time visualization capabilities for engineers, architects, and scientists. This paper presents Virtual-SAP, an immersive VE application allowing users to assess building structures and their response to various environmental conditions through an interactive design-build-test-redesign cycle. Virtual-SAP has three distinct user interfaces that support its use with a high-end, multi-tracker VE system, with a low-cost portable VE system, or on the desktop. Two of these interfaces have been proven highly usable in user testing, and the third will be tested soon. Virtual-SAP is being used for both research and education.

1. Introduction

Interactive 3D computer graphics have long been used for scientific and engineering visualization. Visualizations allow researchers and students to see phenomena and patterns that are difficult or impossible to understand from an inspection of numeric data or equations alone. More recently, immersive virtual environments (VEs) have been applied to data and information visualization [6, 10]. Immersive VEs can provide the experience of "being within" the data in order to view and interact with it in a more natural way. The use of natural head and body motions and the spatial placement of data around the user can lead to greater understanding of the three-dimensional (3D) layout and structure of the data. Furthermore, if the user can manipulate the data interactively using natural actions (e.g. hand motions), she could potentially steer the visualization easily and intuitively, leading to further understanding or insight.

We have used these principles in the design and implementation of Virtual-SAP, an immersive VE application focused on the visualization of building structures and their response to various environmental conditions. Current approaches to building structure design and analysis focus either on a high-fidelity graphical representation (e.g. CAD systems), or on numerical simulation of structures. Virtual-SAP combines these approaches by allowing the user to enter a 3D environment, build or import a 3D structural model of a building, run an environmental simulation, and view the results as a 3D animation while still immersed in the space. Currently, Virtual-SAP is being used to simulate earthquake conditions, although other effects such as heat or wind could also be simulated. The user may modify the structure any number of times and rerun the simulation to determine if the changes had the desired effect.

We begin by describing related work, and then provide an overview of the design and functionality of Virtual-SAP. Next, we discuss three unique user interfaces allowing the use of Virtual-SAP in many different contexts, and describe the results of a formal usability study. Finally, we present the current use of Virtual-SAP as an educational tool, and discuss future work for this ongoing project.

2. Related Work

Many VE applications have been developed that allow users to view the results of simulations. For example, Brogan et al [4] used dynamic simulation to generate the behavior of characters in a VE. Another application, the Virtual Wind Tunnel [7], allowed users to interact with simulated virtual flow fields.

Architecture has been one of the main application areas for VEs. Immersive walkthroughs have been used extensively for design verification [5, 14]. More recently, researchers have studied *immersive design*, the idea that VEs can be used not only for visualization, but also for design and modification of architecture. For example, the

ISAAC project [16] gave users tools for creating, transforming, and grouping 3D objects within a scene while immersed within the scene. The Virtual Habitat [3] used domain-specific interaction techniques to allow users to modify the design of an existing animal habitat. Virtual-SAP is an immersive design tool similar to these applications, but it also goes one step further by integrating environmental simulation into the tool. A project by Ishii and colleagues [12] has addressed similar real-time simulations in an augmented reality environment.

Several researchers have also looked at structural issues in VEs. Impelluso [11] linked both linear and non-linear 3D finite element programs with visualization applications in order to present the deformations of virtual objects subjected to various loading conditions. Yeh and Vance [21] used an immersive design system to perform an interactive sensitivity analysis and design optimization of a cantilever beam. Murakami et al. [17] developed a VE-based CAD system allowing the user to interact with virtual structures through joysticks that apply reaction forces computed by the finite element code. Canfield [8] developed a system for real-time simulation of mechanical systems in a VE. This system can project various scalar fields onto the exterior surface of 3D objects based on simulation results. Chou, et al. [9] developed a desktop VE-based learning approach for the structural curriculum at the National Chiao Tung University. Students could interact with structural elements through the use of a 3D mouse.
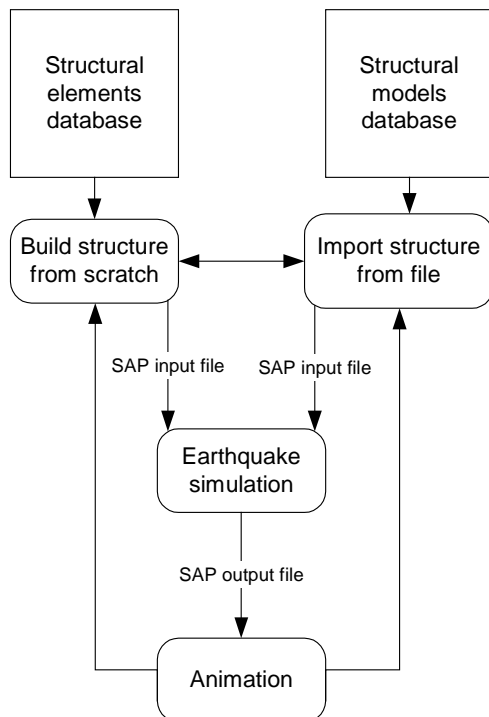
3. Virtual-SAP overview

Figure 1 shows the basic workflows for users of Virtual-SAP. First, users define a building structure. This can be accomplished by creating the structure from scratch using individual structural elements, by importing a model from a file, or by a combination of these two methods. Next, a Structural Analysis Program (SAP) simulation subjects the building structure to an earthquake. Finally, the results of the simulation are displayed via 3D animation. This build-simulate-animate cycle can be performed as many times as desired, in order to test the effects of various changes or structural options. These steps are described in detail in the sections below.

In addition to the basic build-simulate-animate operations, users have complete freedom to travel anywhere in the 3D environment at any time. This allows users to place the viewpoint at an appropriate location for modifying a particular section of the structure or for viewing the animation.

3.1. Definition of building structure

When Virtual-SAP starts, users are placed within an empty environment, with the exception of a 3D grid and a ground plane. Users can build a structure from scratch by selecting from various element types (beams, columns, beam/columns, walls, and slabs), various materials (steel, concrete, wood, etc.), and various element shapes and sizes. Once an element has been defined, it can be created and placed in the 3D environment. Elements snap to a grid in order to facilitate precise placement. Once elements have been placed, users can move, rotate, or delete them.
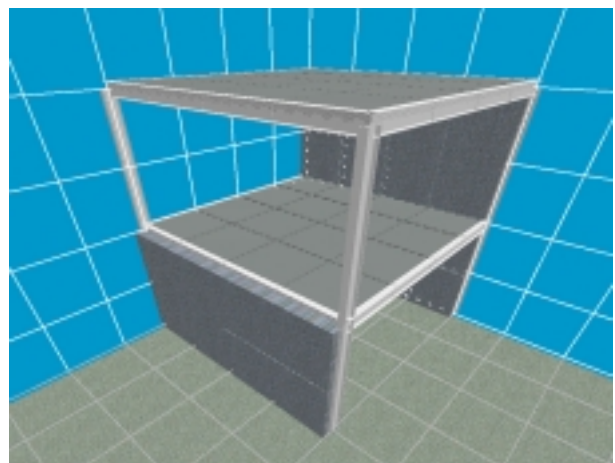

Figure 2. Complete structure in Virtual-SAP

Since building a complex structure from scratch can be tedious, users can also import structures from a text file. We use the SAP input file format (see section 3.2) for

Figure 1. Overview of Virtual-SAP

importing structures. At any time, users can also export the current structure to a file so that it can be reused later.

The most common method of building a structure is to import a basic framework (e.g. a two-story building shell) from a file and then extend the structure with additional elements. Figure 2 shows a complete structure built in Virtual-SAP.

3.2. Earthquake simulation

For the simulation of the structure's response to an earthquake, we use the Structural Analysis Program (PC-SAP IV). This finite element simulation package provides for the modeling of a wide range of structures and simulation in many different environmental conditions.

Users can select from a database of pre-defined earthquakes and also specify a scale factor so that the structure's motion can be amplified if desired. Virtual-SAP uses the building structure and earthquake information to produce a SAP input file (the same file format used for importing a structure). In order to accomplish this, the system must determine the location of each of the *nodes* within the structure and the element *connectivity* between nodes. For example, a beam is defined by two nodes (its endpoints) and one element connection (the line between the endpoints). SAP then runs (as a separate process) and produces an output file that specifies the displacement of each of the nodes in the structure at specified time steps.

3.3. 3D Animation

When the simulation is complete, the user can "play" the resulting animation (figure 3). The user can view the animation from any location within the VE, and can travel to new viewpoints even while the animation is playing.

By default, the animation runs in real time (1 second of simulation time = 1 second of animation time), although we plan to allow playback at different rates in the future. In order to animate the structure properly, the system must transform the nodal displacement values into transformations (translations, rotations, and scalings) to be applied to the geometric structural elements. This process must ensure that all elements connected to a particular node move together at that node. Elements are moved to their correct positions through a series of matrix transformations. For example, when updating a beam's position, the beam model is first translated so that one of its endpoints is at the origin. It is then scaled to the proper length. After that, the beam is rotated around the x, y, and z axes so that it is oriented properly. Finally, it is translated to its correct position in the structure. Floor slabs and wall segments are transformed in a similar way.
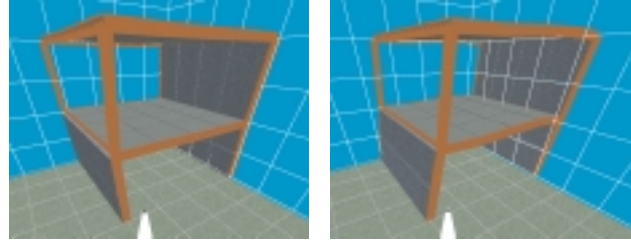

Figure 3. Two frames from the animation of the structure in figure 2

Virtual-SAP can also indicate to the user when a node has moved beyond a user-defined threshold displacement. The threshold is based on the height of the node above the ground (higher nodes are allowed to move farther before they pass the threshold). When a node's displacement moves beyond the threshold, all beam and column elements connected to that node are highlighted.

4. Virtual-SAP user interfaces

Virtual-SAP has three distinct user interfaces (UIs) to allow users to perform the tasks described in section 3. Each UI was designed for a specific context of use, but each one also provides access to the complete functionality of Virtual-SAP. Because of this, users have the freedom to choose the interface that best fits their work environment (available hardware) and set of tasks. We believe that this multiple interface approach makes Virtual-SAP somewhat unique among immersive VE applications.

The software systems on which Virtual-SAP is built are the same for all three UIs. The application uses the Simple Virtual Environment (SVE) library [13], and runs on Windows 2000-based PCs.

4.1. Immersive pen & tablet interface

We originally designed Virtual-SAP as a purely immersive, high-end VE application. Therefore, we designed a UI making use of a head-mounted display (HMD) and a multi-sensor tracking system. Our implementation (figure 4) uses a Virtual Research V8 HMD and an Intersense IS-900 VET tracking system. This system tracks the user's head and two input devices.

The dominant metaphor for this UI is the "pen & tablet" interface [1, 19]. The user holds a tracked tablet in one hand and a tracked pen (stylus) in the other. In the virtual world, the user sees virtual representations of both the pen and the tablet. The virtual tablet displays a 2D interface within the 3D environment (figure 5), and the pen can be used to select or manipulate items within this 2D UI.

The tablet interface is used for all system control interaction tasks within Virtual-SAP, as well as for some object manipulation tasks. It is organized into several "tabs," similar to many desktop interfaces, which contain sets of related controls. The *Elements* tab contains controls for defining element type, material, shape, and size, for creating a new element with the current attributes, and for rotating or deleting a selected element. The *Seismic* tab allows the user to select a pre-defined earthquake record or set the magnitude of the earthquake. The *Overview* tab displays a miniature 3D representation of the current model, similar to the World-in-Miniature technique [20]. The *Import/Export* tab allows the user to import a structure from a file or save the current structure to a file. Finally, the *Simulate* tab is used to simulate the response of the current structure to the current earthquake and to play back the simulation results. The *Loads* and *System* tabs are currently unused and are meant for future additions to the system.

Figure 4. Hardware devices used in the immersive pen & tablet UI

Figure 5. Tablet-based 2D interface

Users also interact directly with the 3D environment in this UI. We use a pointing technique [15] for navigation within the environment. Users point the stylus in the desired direction of motion and hold down a stylus button to travel in that direction. Users can also select and manipulate building elements directly with the stylus. In order to allow users to manipulate these objects at a distance, we use the Go-Go technique [18], a non-linear mapping between the physical and virtual hand positions that greatly increases the range of the virtual hand.

Figure 6. Desktop interface

4.2. Desktop interface

We recognized that not all of the tasks in Virtual-SAP require the user to be immersed. For example, building a complex structure from scratch might be tedious in the immersive UI, due to the weight of the HMD or input devices, the limited resolution of the HMD, etc. Therefore we developed a desktop UI for Virtual-SAP that can be used at a standard PC workstation with standard input devices (mouse and keyboard).

This interface uses a tabbed organization with controls identical to the tablet interface in the immersive pen & tablet UI. When using the desktop interface, two separate windows are used: one shows a view of the 3D environment (see figure 2) while the other is used for the tabbed UI (figure 6). The interface window is used for all of the interaction tasks handled by the tablet in the immersive UI, and it also includes controls for movement of the virtual stylus so that 3D positions can be specified for objects being manipulated. Viewpoint movement (travel) is done directly in the 3D view.

The desktop UI was implemented using the Fastlight Toolkit (FLTK, www.fltk.org) and a shared memory library based on the Memory Mapped Files available in Microsoft's Win32 API. When the user performs an action in the 2D interface, the relevant information is written to shared memory. The 3D Virtual-SAP application then reads the data and calls the appropriate function(s).

4.3. Simplified, low-cost immersive interface

After Virtual-SAP had been under development for almost two years, we became interested in using it as an educational tool in the university classroom (see section 6), using the application to teach students of building structures principles of sound structural design. For this purpose we needed an inexpensive, portable VE system that could be taken to classrooms and that would be easy to set up and maintain. Using off-the-shelf components, we constructed such a system with an A/V cart, a desktop PC, an LCD projector (so the entire class can view the VE), a consumer-level HMD (the Daeyang i-visor), a 3 degree-of-freedom head tracker (the Intersense Intertrax2), and a chord keyboard (the Handykey Twiddler2).

Since this system tracks only the orientation of the head, the pen & tablet interface was not feasible for this purpose. The design of the UI for this system was much more challenging, especially given the large amount of functionality in Virtual-SAP. We considered leaving the layout of the tablet the same and using one button on the input device to cycle through the various tablet options and another button to select the current option, similar to the way the TAB and ENTER keys can be used in many desktop interfaces. However, this would require a large number of button presses to access most options, which would not be appropriate for classroom use, where time is critical. Moreover, the virtual tablet would have to be attached to the user's view (since no tablet tracker is available), which would obscure most of the 3D environment, especially since the HMD has only a 30-degree horizontal field of view.

Therefore, we designed a new interface for this system based on our chord keyboard input device. The chord keyboard is designed for use as a one-handed text input device for mobile and wearable computer users, and we have also studied its utility for text entry in VEs [2]. The device has twelve keys on the front, arranged in three columns of four keys each. Normally the user holds the device upright so that the keys can be pressed by the four fingers on the dominant hand (figure 7, left), but we have users hold the device like a television remote control, and press the keys with the thumb (figure 7, right).

Since there are more than twelve functions in Virtual-SAP, we could not simply assign a key to a single function. Rather, we re-organized the tablet functions into three categories: elements, simulation, and general controls. The top three keys select the mode, and the rest of the keys change functionality depending on the current mode. Obviously users should not be forced to remember all of these changing key bindings, so we designed a visual aid that is attached to the user's view and displays the current function of each of the keys (figure 8). Contrasting colors are used to indicate the current mode.

Additional text at the bottom of the view indicates the currently selected element attributes.

The chord keyboard and head tracker must also be used for navigation and object selection and manipulation in this UI. For navigation, we use gaze-directed steering [14] and allow both forward and backward motion along the gaze vector. Since navigation is so common, we always assign navigation controls to the same two keys on the input device, regardless of the current mode.

Object selection and manipulation are also based on the gaze direction. A small crosshair in the center of the view helps the user to align her gaze with an object. When the selection key is pressed, the first object intersected by the ray along the gaze vector becomes selected, at which time the user can delete or rotate the object using additional key presses. The user can move the object by turning her head, or by using two keys to move the object forward or backward along the gaze vector.

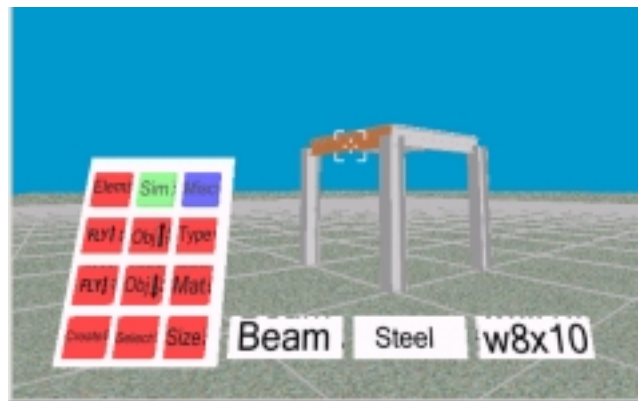**Figure 7. Standard chord keyboard grip (left) and modified grip for Virtual-SAP (right)**

Figure 8. Visual aids in the simple immersive UI

5. Usability study

We performed a task-based usability study of the pen & tablet and desktop UIs for Virtual-SAP. The goals of the study were to identify any critical usability problems with the interfaces and to examine differences in performance and preference between the two interfaces. The simplified immersive interface was still under

development at the time of the study, but we plan to use the same procedure to test its usability at a later date.

5.1. Tasks

We tested performance and usability on several common and critical tasks in Virtual-SAP. The first task involved building a moderately complex structure from scratch based on a written description and diagram. Second, users were to select an earthquake, run the simulation, and play the animation. The third task required users to navigate to another side of their structure and watch the animation again to see if any different behavior of the structure could be seen. Finally, users described to the evaluator how they would modify the structure to make it safer based on what they had seen.

5.2. Subjects

Six male undergraduate students participated in the study. All of the subjects had some limited prior experience with VEs, but none of them had extensive experience, and none had previously used Virtual-SAP. The average age of the subjects was 22 years. We did not choose to use architecture students for this study, because we were interested in the usability and performance of the Virtual-SAP interfaces, rather than the application's ability to teach building structures concepts. We will examine the use of the application for education in a separate study.

5.3. Procedure

Subjects completed a demographic questionnaire, signed an informed consent form, and read written instructions for the study. The evaluator then presented a ten-minute tutorial to the subject showing the most important features of Virtual-SAP; this is similar to the training that students or other users might receive before using the application for the first time. Next, subjects completed the four tasks in both the immersive and desktop interfaces. Half of the subjects began with the desktop UI while the other half started with the immersive UI. After each interface, subjects completed a post-questionnaire. At the end of the session, subjects were also given a questionnaire comparing the two interfaces.

Besides the questionnaires, metrics included the time to complete each task, the number of errors subjects made, and subjective comfort ratings in the areas of arm/hand strain, dizziness, and nausea, which were elicited before the tasks began and twice during the use of each interface. The evaluator also noted critical incidents and usability issues during task performance.

5.4. Results

Both interfaces performed well in the evaluation. All subjects completed all tasks in a reasonable amount of time. Subjects rated Virtual-SAP highly in terms of their satisfaction with the experience (average response 6.3 on a 7-point scale) and their understanding of the 3D earthquake visualization (6.5). However, subjects reported that more feedback was needed to confirm user actions.

Not surprisingly, the task of building the structure, which involved creating and placing fifteen separate elements, was the most time-consuming. Interestingly, performance of this task was faster on average with the immersive interface (4.76 minutes) than the desktop interface (7.30 minutes), a marginally significant difference based on a t-test ($t(10,2)=2.02$, $p=.071$). Based on observation, the main limiting factors in the desktop interface were the constant switching between two windows – one used to create an element and another used to place it – and the fact that stylus movement was indirect in the desktop UI. In the immersive UI, the context switch between the tablet interface and the 3D environment is much more fluid. Performance was similar in the two interfaces for the other tasks, although in each case the immersive UI was slightly faster.

Subjects also preferred the immersive UI by a five to one margin in terms of stylus control, ease of navigation, ease of switching between the 2D interface and the 3D environment, and overall ease of use. No significant discomfort was reported by any of the subjects during the study, but four out of six subjects felt that the desktop UI produced higher levels of physical comfort overall.

Based on observations of critical incidents and subject comments, several suggestions for improvements to the interface surfaced during the study. These include disabling certain controls when their use is not appropriate, improvements to the stylus movement interface in the desktop UI (or the use of an alternative method of object selection and placement), increased graphical and audio feedback, and changes to the color scheme. These improvements are being implemented in our current round of development.

6. Using Virtual-SAP in education

As we have noted, Virtual-SAP can be used as a research and design tool by structural engineers and architects, but it is also useful as a teaching tool for students of building structures. Consider the following classroom scenario:

A building structures class is studying earthquakes. They have attended lectures and studied written material explaining the theories involved, numeric models used, and properties of various materials and shapes. The VE

system has been used to show some predefined examples of earthquake effects to the entire class using a projector. Now the instructor wants the class to test its knowledge. He divides the class into small groups, and asks each group to select structural systems for a building, given certain architectural constraints. As each group finishes, a teaching assistant enters the design using a simple text file, and the structure appears in the VE. One of the group members is invited to wear the HMD and navigate to interesting viewpoints while the animation is displayed, and each group sees the results of the earthquake. The instructor allows each group to make one change to its structure that they think will make it more robust. When all groups have done this, all of the structures are loaded side-by-side into the environment, and the entire class watches as the earthquake is simulated.

As the scenario shows, there are many pedagogical benefits that could accompany the use of Virtual-SAP in the classroom. Students view structures in 3D rather than as 2D diagrams; they see the dynamic response of the structures to earthquakes, making clear the meaning of the theories they have studied; they see how architectural design decisions can change structural behavior; and they can interact with the system and test their own knowledge.

We are just beginning the actual use of Virtual-SAP in the classroom, using the portable, low-cost VE system described earlier. We are designing in-class activities and integrating the use of the application with traditional lectures. In particular, we are interested in whether immersion can be an important factor in learning these concepts, or if 3D animation is sufficient. We will also be studying learning and comprehension gains both quantitatively and qualitatively.

In addition, we are working with teachers at a local high school to introduce building structures concepts as a special topic in their curriculum using Virtual-SAP. Students in these classes will view earthquake animations from a set of typical structural models and will also be able to make modifications to the structures using the desktop UI.

7. Future directions

Virtual-SAP is an ongoing project. While the core functionality is in place, there are still many additional features and enhancements we plan to make to the system. This section briefly describes a few of these future plans.

One area under active research involves more efficient and intuitive selection and manipulation of groups of structural elements. For example, one might want to select an entire row of columns and move them together as a unit. To support such tasks we are prototyping various interaction techniques for multiple selection, based on the concept of a 3D selection box (figure 9).

Another limitation of the current system is that the simulation never results in complete failure of the structure – connected elements always remain connected regardless of the severity of the earthquake. We are designing a method for modeling building failure once the movement exceeds a certain threshold. Similarly, we would like to investigate methods for animation of torsion or twisting of structural elements.

Improving the interactivity of the design-simulate-animate cycle is a third area for improvement. Currently, the simulation process runs on the same PC as the VE, and the simulation of complex models can take 5 minutes or more (although simple models used for teaching purposes can be simulated very quickly). Obviously, this discourages the user from "tweaking" the structure based on the simulation results. We are investigating remote and parallel computation schemes to reduce the time needed for structural simulation.

We are also planning to use Virtual-SAP to simulate and animate other environmental conditions besides earthquakes. For example, we could simulate extreme temperature or wind conditions.

Finally, we are in the process of expanding the set of possible structures and conditions that can be modeled in Virtual-SAP. We are adding new element types, sizes, and materials, and we are also including new earthquake records in the system.
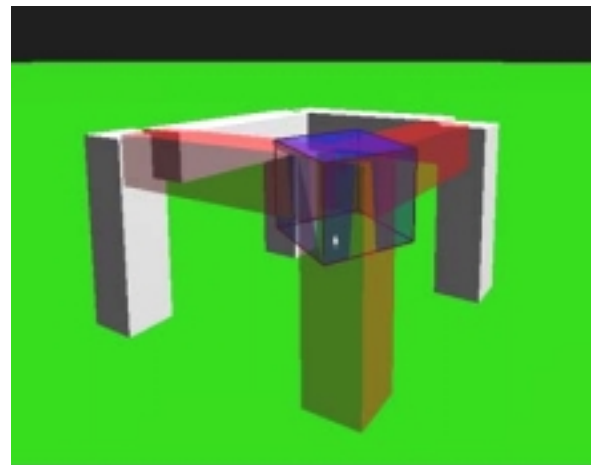

Figure 9. Prototype multiple selection and grouping technique

8. Conclusions

This paper has presented Virtual-SAP, an immersive VE application allowing the modeling of building structures and the simulation and animation of these structures' response to environmental conditions. Virtual-SAP can be used for design and analysis of real structures, research in structural design, and education. The

availability of three unique and usable interfaces to the application makes Virtual-SAP useful in many situations.

This project has also been beneficial to research in 3D user interfaces and interaction. We have learned valuable lessons about designing multiple interfaces for the same VE application. Many 3D interaction techniques have also been put to the test in the context of a real-world application, and new techniques are being developed to address the needs of tasks within this system.

Acknowledgements

References

[1] Angus, I. and Sowizral, H. "Embedding the 2D Interaction Metaphor in a Real 3D Virtual Environment." In *Proceedings of SPIE, Stereoscopic Displays and Virtual Reality Systems*, 1995, pp. 282-293.

[2] Bowman, D., Rhoton, C., and Pinho, M. "Text Input Techniques for Immersive Virtual Environments: an Empirical Comparison." In *Proceedings of Human Factors and Ergonomics Society Annual Meeting*, Baltimore, 2002, pp. 2154-2158.

[3] Bowman, D., Wineman, J., Hodges, L., and Allison, D., "Designing Animal Habitats Within an Immersive VE." *IEEE Computer Graphics & Applications*, 1998. **18**(5): pp. 9-13.

[4] Brogan, D., Metoyer, R., and Hodgins, J., "Dynamically Simulated Characters in Virtual Environments." *IEEE Computer Graphics & Applications*, 1998. **15**(5): pp. 58-69.

[5] Brooks, F. "Walkthrough - A Dynamic Graphics System for Simulating Virtual Buildings." In *Proceedings of Workshop on Interactive 3D Graphics*, Chapel Hill, North Carolina, 1986.

[6] Bryson, S., "Virtual Reality in Scientific Visualization." *Communications of the ACM*, 1996. **39**(5): pp. 62-71.

[7] Bryson, S. and Levit, C., "The Virtual Wind Tunnel." *IEEE Computer Graphics and Applications*, 1992. **12**(4): p. 25-34.

[8] Canfield, T. "Simulation and Visualization of Mechanical Systems in Immersive Virtual Environments." In *Proceedings of the 12th ASCE Engineering Mechanics Conference*, La Jolla, California, 1998, pp. 182-187.

[9] Chou, C., Hsu, H., and Yao, Y., "Construction of a Virtual Reality Learning Environment for Teaching Structural Analysis." *Computer Applications in Engineering Education*, 1997. **5,** pp. 223-230.

[10] Datey, A., "Experiments in the Use of Immersion for Information Visualization" Unpublished Masters Thesis, 2002, Virginia Tech: Blacksburg, VA.

[11] Impelluso, T. "Physically-Based Virtual Reality: Integrating FEM and Visualization." In *Proceedings of the 12th ASCE Engineering Mechanics Conference*, La Jolla, California, 1998, pp. 178-181.

[12] Ishii, H., Underkofler, J., Chak, D., Piper, B., Ben-Joseph, E., Yeung, L., and Kanji, Z., "Augmented Urban Planning Workbench: Overlaying Drawings, Physical Models, and Digital Simulation." In *Proceedings of IEEE/ACM International Symposium on Mixed and Augmented Reality*, 2002, pp. 203-211.

[13] Kessler, G., Bowman, D., and Hodges, L., "The Simple Virtual Environment Library: An Extensible Framework for Building VE Applications." *Presence: Teleoperators and Virtual Environments*, 2000. **9**(2): pp. 187-208.

[14] Lindsey, P. and McLain-Kark, J., "A Comparison of Real World and Virtual World Interior Environments." *Journal of Interior Design*, 1998. **24**(1): pp. 27-39.

[15] Mine, M., "Virtual Environment Interaction Techniques." Technical Report TR95-018, 1995, UNC Chapel Hill CS Dept.

[16] Mine, M., "ISAAC: A Meta-CAD System for Virtual Environments." *Computer-Aided Design*, 1997. **29**(8): pp. 547-553.

[17] Murakami, H., Nishimura, Y., Impelluso, T., and Skelton, R., "A Virtual Reality Based CAD System for Tensegrity Structures." In *Proceedings of the 12th ASCE Engineering Mechanics Conference*, La Jolla, California, 1998, pp. 197-200.

[18] Poupyrev, I., Billinghurst, M., Weghorst, S., and Ichikawa, T., "The Go-Go Interaction Technique: Non-linear Mapping for Direct Manipulation in VR." In *Proceedings of the ACM Symposium on User Interface Software and Technology*, 1996, pp. 79-80.

[19] Schmalsteig, D., Encarnacao, L., and Szalzvari, Z. "Using Transparent Props For Interaction with The Virtual Table." In *Proceedings of the ACM Symposium on Interactive 3D Graphics*, 1999, pp. 147-154.

[20] Stoakley, R., Conway, M., and Pausch, R. "Virtual Reality on a WIM: Interactive Worlds in Miniature." In *Proceedings of CHI: Human Factors in Computing Systems*, 1995, pp. 265-272.

[21] Yeh, T. and Vance, J. "Interactive Design of Structural Systems in a Virtual Environment." In *Proceedings of 24th Midwestern Mechanic Conference*, Ames, Iowa, 1995, pp. 185-187.