HOME DESIGN IN AN IMMERSIVE VIRTUAL ENVIRONMENT

Jason Cowden[1], Doug A. Bowman[1], and Walid Y. Thabet[2]

[1]*Department of Computer Science*
[2]*Department of Building Construction*
Virginia Tech
Blacksburg, VA 24061 USA
540-231-2058
{jcowden, bowman, thabet}@vt.edu

ABSTRACT

Virtual environments (VEs) can contribute to the building construction design phase by providing a 3D immersive experience for design as opposed to a 2D representation of a 3D structure. Such walkthrough applications have been a common way to review designs before construction. The ability to visualize a model in a VE is helpful, but the ability to *modify* the design of a structure in an immersive VE can provide much more expressiveness and can shorten the design cycle. We present an immersive home design tool for the CAVE that allows a user to navigate through the model of a residential home and also redesign the elements of the home while immersed. Our initial prototype and our plans for the project are presented in this paper.

KEYWORDS

3D interaction, CAVE, Home design, virtual environments.

INTRODUCTION

Three-dimensional (3D) visualization is currently used widely in the building construction process. There is a point, however, at which traditional 3D visualization approaches (viewing 3D models using CAD software on desktop displays) become limited. While these visualizations are very helpful, they have no comparison to the experience of viewing a physical, full-scale mockup in real space. Thus, some construction companies create actual life-size mockups of model homes so that problems can be discovered before construction of these homes commences. The advantages of seeing a mockup are many. Experts and subcontractors such as architects, masons, carpenters, electricians, plumbers, and interior designers can easily discover many problems by walking through the mockup. Unfortunately, finding a problem is not the same thing as finding the solution. Therefore, several iterations may be needed, and there may be several mockups created and destroyed. The obvious costs of parts, labor, and travel costs for the domain experts to get to the mockup site are outweighed by the finished design; however, it is a cost that adds up for any company. Another purpose of physical mockups is to provide a better understanding of the 3D spatial structure and scale of the current design.

We claim that performing the process of design review and redesign in a virtual environment (VE) may mitigate many of the costs associated with physical mockups, while retaining the advantages of immersion within the design and a sense of the scale of the model. VE systems of this type are called *immersive design* systems.

The purpose of this research is to aid in the design phase of the construction process by using VEs for immersive design. In a VE, a final prototype can be designed more quickly without any overhead costs for parts, and a lesser amount for the labor. The costs of labor do not vanish since building the virtual prototype will require the collaboration of the initial designers and the domain experts. Also, if this collaboration is done remotely, the domain experts can interact with the same model without having to travel to a single site.

For the user to be able to make design decisions and perform design actions while immersed in a VE, the user interface (UI) needs to be intuitive and simple. Users are accustomed to 2D desktop interaction, but are usually unfamiliar with the 3D spatial interfaces used in immersive VEs. For certain interactions, a 2D interface might be more suitable and easy to learn. For example, the user might select from a menu to add a particular object to the model. Prior work suggests that we can embed a 2D interface within a 3D VE for an immersive design system. In this work, a pen and tablet interaction metaphor was used to manipulate a model. Objects were added to the virtual world by selecting the desired object from a menu on the virtual tablet and then selecting a location on the ground in the VE (Bowman, Wineman, Hodges, Allison 1998).

In this paper, we present an initial prototype for an immersive design system for residential home design review. We also describe our plans for extending this system to provide a full suite of functionality. Throughout, our focus is on usable and intuitive interfaces for the user.

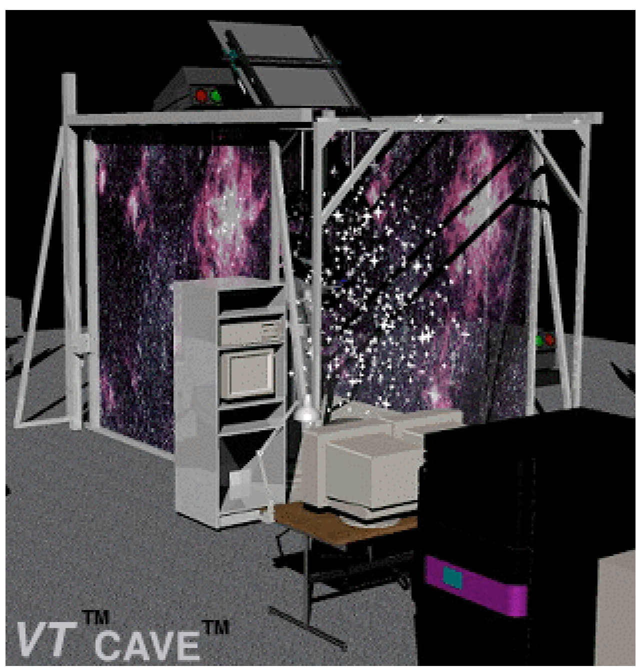

Figure 1: The VT CAVE.

CURRENT PROTOTYPE

This research project is being developed primarily for use in a CAVE environment. A CAVE is a spatially immersive display device (Cruz-Neira et al., 1993) with a tracking system as seen in Figure 1. There are four display surfaces (front wall, left wall, right wall, and floor), each being ten feet by ten feet in area. A stereo 3D image is projected on each surface. The user's head and one hand are tracked while he or she moves around inside the CAVE. The hand tracker is called a wand. The wand includes four buttons and a 2D joystick, and is used for user input, either by button presses or joystick movement. The button presses allow the user to manipulate the world, while the joystick allows the user to navigate throughout the world.

Because we developed our application using the application programming interface DIVERSE (Kelso et al., 2002) it can also run on other display devices, such as a head-mounted display (HMD) or even on a typical desktop display.

Current Functionality

Our initial prototype allows users to navigate through and interact with a predefined 3D home model. The model is built using a standard 3D modeling tool, such as Discreet's 3D Studio Max. Designers identify model elements by giving them standard names related to their function in the model (e.g. window, wall, door, etc.). Our system loads these elements as separate 3D objects and places the model in the scene.

The user can navigate around and through the model in three dimensions. Interaction with the model is currently limited to hiding/showing individual elements. Even this minimal interaction, however, can make a big difference in the user's perception of the model. Experienced users can see the home as it will look after the framing stage and after drywall and ceilings have been added, for example. This set of functionality has also allowed us to design and develop an interesting user interface, as we discuss in the following section.

Figure 2: Users can select a wall by pointing the wand at the wall and pressing the "select" button. The "virtual wand" showing the currently available commands is in the lower-left.

User Interface

Usable interfaces include both affordances and feedback (Norman, 1990). Affordances help the user understand what actions are possible and how to perform those actions, while feedback helps the user understand the results of prior actions. In the interface to our application, we provide affordances and feedback by displaying a graphical representation of the wand input device on one or more walls of the CAVE. This "virtual wand" has labels next to each of the buttons that show the user which commands are currently mapped to the buttons of the wand. When the user performs an action that changes the state of the system, these button mappings will also change to match the new situation. The user receives feedback for these changes because the labels on the virtual wand also change.

Figure 3: Once selected, a wall can be hidden to allow designers to see other parts of the structure.

When the user points the physical wand at an object that is selectable, such as a wall, window, or piece of furniture, the virtual wand display shows that a button press will result in selection of that object (Figure 2). Once an object is selected it can be hidden/deleted, moved, etc. depending on the type of object (Figure 3). Additional feedback is provided directly in the environment. A selected object is visually highlighted for the user, and if it is movable, a set of coordinate axes is displayed on the object.

To re-display hidden objects, there is a mode called "show" that displays all hidden objects as transparent. The user can then point to a hidden object and press the "show object" button, causing it to become opaque again. In addition, the user may choose to re-display all hidden objects of a certain type, such as showing all walls.

The user can navigate freely through the VE to get different perspectives on the model. Compared to a physical mockup, where access to certain parts of the model may be limited, a VE provides complete access, possibly allowing the detection of additional design problems. We use the pointing technique (Mine, 1995), in which users point the physical wand in the direction they want to travel, and then use the joystick to move in that direction (or the reverse direction). The joystick is also used to virtually turn about the vertical axis of the environment. The user can easily obtain even novel views, such as the bird's eye view shown in Figure 4.

Figure 4: Viewing a home model from a bird's eye point of view.

FUTURE ENHANCEMENTS

We intend to continue the development and evaluation of our immersive home design application to make it useful for real-world design reviews. This section details a few of the enhancements that are currently under development.

First, we will add object manipulation to the VE, so that users can actually modify the design of the home. For objects such as windows and furniture, we will include constraints, only allowing these objects to move in reasonable ways (e.g. windows must remain upright, on a wall). This will increase the accuracy of adding and editing these objects in the world. Also, objects that will change a fundamental part of the model, such as a window changing studs in the house frame, will only be able to move in ways that are allowable according to construction standards. Collision detection will keep objects from moving through walls or floors, while restrictions on the objects' degrees of freedom will prevent invalid movements, such as placing a chair on the ceiling.

Next, we will create an initialization tool that will ease the specification of models for the system. The user will be able to create a new model or load a previously created model to initialize the program. If a user decides to create a new model, he or she will have to enter enough information for the program to create a basic model that the user can add to or edit. The foundation and frame are objects that the user should not have to worry about constructing, especially since the program can create these depending on the dimensions the user inputs into the initial configuration. This initialization tool will be very specific to the task, as opposed to commercial systems, which have many uses. This will allow users to specify pertinent information directly. More specifically, dimensions will be the only data needed to build the house frame. Building construction standards will dictate how the program creates objects of this nature.

Once in the VE, the user will need to add objects to the model. This becomes nontrivial very quickly since there are many different types of objects that the user may want to add. For example, a user may want to add a window to a wall. There needs to be a way for the user to specify the type of window and what attributes the window has. A 3D menu system is a solution, but comes with its disadvantages as well. One disadvantage is that the menu may occlude part of the environment. If the user is trying to add an object to the world he or she may become disoriented when the action is done and the menu is removed. The physical input device (the wand) also makes menu design difficult. Since there are only four buttons, there can only be a limited number of options. This was an issue in a menu system developed for pinch gloves (Bowman & Wingrave, 2001). Part of the solution was to display the menu only when the context called for it. The same idea can be used in our system. Whenever it is necessary to have a menu, the other commands will be ignored and the buttons will be used to select from the menu, exit the menu system, or bring up different menus.

The user will be continuously updated on what options are available to him or her through the virtual wand display. Levels of abstraction will help keep the number of different actions reasonable. For example, electricians do not have to worry about seeing commands relating to changing the color of a wall, and an interior designer does not have to worry about hiding walls to see the house frame and other systems that are located there. The user will be asked to initially give an abstraction preference, which can be changed later within the VE. The VE will respond by showing only a subset of all the possible actions to each particular user. This makes interaction easier by providing a simpler, cleaner interface.

CONCLUSION

We have presented the initial prototype and future plans for a home design application implemented in an immersive VE. The ability to design while immersed in a VE is critical if a virtual mockup is used instead of physical mockups. Such systems represent a new and less expensive way of viewing, critiquing, and editing a home design. We have focused on the usability of the 3D interaction techniques and interface metaphors in our system, since usability plays a large role in end-user acceptance.

ACKNOWLEDGEMENTS

We would like to thank the 3D Interaction Group at Virginia Tech for many different ideas for the future of the project. Also we would also like to thank John Kelso and Ron Kriz for their help and assistance whenever it was needed.

REFERENCES

Bowman, D., Wineman, J., Hodges, L., &Allison, D. (1998). Designing Animal Habitats Within an Immersive VE. IEEE Computer Graphics & Applications, 18(5), 9-13.

Bowman, D. and Wingrave, C. (2001). Design and Evaluation of Menu Systems for Immersive Virtual Environments. Proceedings of IEEE Virtual Reality, 149-156.

Cruz-Neira, C., Sandin, D., and DeFanti, T. (1993). Surround-Screen Projection-Based Virtual Reality: The Design and Implementation of the CAVE. Proceedings of ACM SIGGRAPH.

Kelso, J., Arsenault L., Kriz, R. (2002). DIVERSE: A Framework for Building Extensible and Reconfigurable Device Independent Virtual Environments. Proceedings of IEEE Virtual Reality, 183-190.

Mine, M. (1995). Virtual Environment Interaction Techniques. Technical Report TR95-018, UNC Chapel Hill CS Dept.

Norman, D. (1990). The Design of Everyday Things, Doubleday, New York.