CHAPTER VI

INTERACTION IN A REAL-WORLD VE APPLICATION

6.1 Integrating Techniques into an Application: Issues and Challenges

The formal categorization, design, and evaluation that has been discussed in the previous chapters cannot be an end unto itself. Rather, it must be done with a view towards applying those results to some practical, useful, real-world systems. The reason interaction techniques are so important is that they allow the user to act – carry out some task that is a part of the user's productivity, education, or enjoyment. Therefore, in this chapter we will consider a practical VE application with extensive interactivity requirements, and how that application can benefit from the formal evaluation of interaction techniques (hypothesis 3).

However, applying the results of our experiments to an application is not as straightforward as it might seem. Recall that in our methodology, application developers specify levels of performance that are required by the application (for many different performance metrics), and then implement techniques that have been shown to meet those requirements through testbed evaluation. There are a number of issues that we must deal with to accomplish this goal.

First, the specification of requirements is not a trivial matter. For quantitative metrics such as efficiency, the developer may have only a rough idea of the requirement. Qualitative measures will be even more difficult to specify. Also, since many VE applications are currently the first of their kind, one may not know the interaction requirements until testing has been done (and to do this testing, you need a working application). We can approach this issue by allowing developers to specify ranges of performance, and by standardizing more qualitative measures. The problem of unknown requirements is not likely to go away, and so iterative design will be imperative. We cannot hope to obtain the perfect set of interaction techniques on the first try for every system.

A second issue is that ITs cannot be considered in a vacuum. If we blindly choose those techniques that best fit our requirements, without regard for how well the techniques work together, we may create a more difficult-to-use application. The issue of *technique integration* is key. Developers of user interfaces have long held the principle that an appropriate overall interaction metaphor makes a system more usable. In the same way in a VE application, a set of three complementary interaction techniques may prove more usable than three unrelated techniques that meet every application requirement.

Finally, we must consider the specific tasks that are part of the application. VE systems for surgical training and interior design may both require accurate object manipulation techniques, but the same technique may not suffice for both applications. The surgery simulator likely needs a high level of realism, while the design application would only be concerned with final placement. Thus, the domain in which the tasks are performed is also important, and should be taken into account when ITs are chosen.

6.2 The Virtual Habitat

The immersive system to which we will apply our results has the goals of user learning and design, and has interesting requirements for interaction techniques for travel, selection, and manipulation. The domain of this system is environmental design, more specifically the design of animal habitats for zoos.

6.2.1 Original VR Gorilla Application

Figure 6.1 shows a wide view of the virtual gorilla exhibit, which is an accurate 3D model of the main gorilla habitat at Zoo Atlanta. The model includes terrain, rocks, trees, fallen logs, moats, an interpretive center, and four virtual gorillas. This model was originally used in an educational application aimed at middle school students (Allison et al, 1997). The students, by taking on the persona of an adolescent gorilla, could learn about gorilla behaviors, vocalizations, and social structure. The user could both explore the habitat freely and interact with the autonomous virtual gorillas.

Figure 6.1 The Virtual Reality Gorilla Exhibit

This original VR Gorilla system is quite interesting for the study of education in VEs and research into believable real-time virtual creatures, but is not as interesting from an interaction point of view, since the user only has to move through the space in some way. To accomplish this, the system uses a simple gaze-directed steering technique, with the user's vertical position constrained to a given height above the ground. However, the developers are considering a torso-directed technique, which is more like walking, and allows the user to look around while moving.

6.2.2 Application to Environmental Design Education

The system on which we will focus our attention is an extension of the original VR gorilla exhibit called the Virtual Habitat. This application is also educational, but is aimed at university-level students, and has the goal of teaching the principles of environmental design. The user is immersed within the same habitat model, with the only difference being that the virtual gorillas now remain stationary and do not react to the user.

The design of zoo exhibits is a topic on the boundaries between architecture, zoology, and psychology, and requires careful attention to a variety of sometimes conflicting requirements. The needs of the animals must be met, and so a naturalistic habitat is often a goal. The animals require some privacy, but visitors must also be allowed to see the animals. Plants need to match the region from which the animal has come, but must also be hardy enough so that they are not destroyed by the animals. In short, there are a number of interesting issues that can be explored by environmental designers (Coe, 1985).

Many of the details of this subject are difficult to learn without examples, and so we felt that the pre-existing virtual gorilla habitat would be an ideal way to provide these examples interactively. Therefore, in one component of the application, users can access embedded information about zoo exhibit design, which are positioned so that the abstract information and the real-world example are colocated. Thus, students have a more visual and interactive method of retrieving information. The embedded information can be in audio, text, or image formats (see figure 6.2). A small study (Bowman, Wineman, Hodges, & Allison, 1999) has shown that this approach, when paired with classroom teaching, may produce better learning and retention of information than a traditional lecture alone.

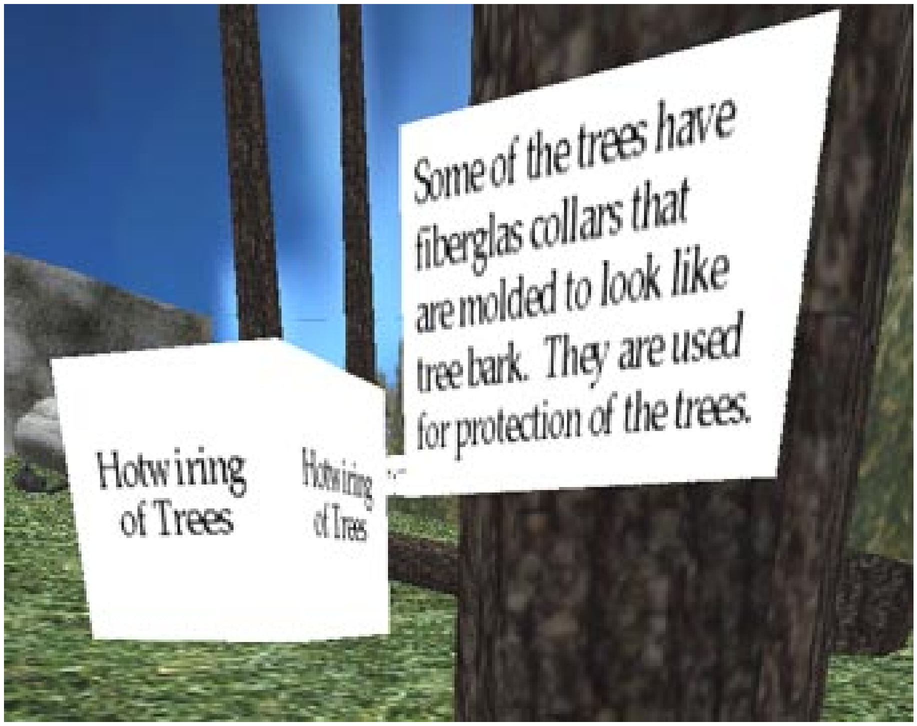

Figure 6.2 Embedded Audio and Text Information in the Virtual Habitat

We also want the user to be able to apply this newfound knowledge in a real-world setting. Therefore, the second component of the virtual habitat application allows *immersive design* – the modification of the existing habitat design while immersed within the habitat (Butterworth et al, 1992, Mine, 1997). Immersive design can tighten the design cycle and allow users to view the effects of changes immediately, but it is also quite different from the way architects are accustomed to designing. They must be able to reason and create in three dimensions, from within the design itself, rather than the normal 2D, outside-in view.

The immersive design component of the Virtual Habitat has three domain-specific tasks. First, users can modify the shape of the terrain, which is important for line-of-sight, privacy, and viewer subordination (Coe, 1985) issues. Second, the visual elements (trees, rocks, tufts of grass) in the habitat can be moved or deleted, or new ones can be created. These elements serve an important aesthetic purpose and influence the naturalism of the exhibit. Finally, the system allows modifications to the design of the visitor viewpoints into the habitat, including their position, viewing direction, and field of view. Issues here include viewing opportunities, privacy, and naturalism.

6.2.3 Interaction Requirements

As we have said, our methodology maps interaction techniques to applications through the use of requirements specification. That is, the application designer specifies levels of the various performance metrics that are required or desirable in the system, and then techniques which have been shown to exhibit those traits can be chosen. The Virtual Habitat application has a number of interesting requirements for interaction, and it includes all of the universal tasks: travel, selection, and manipulation.

There are essentially two different travel tasks that the user of the Virtual Habitat might wish to perform. First, general exploration of the environment needs to be supported. In this type of travel, the user is simply looking around, getting a feel for the layout, size, and features of the VE. For this purpose, a travel technique must be intuitive to the user, so that the focus can be on the environment and not on the technique. It must also allow continuous changes to the trajectory of motion, so that the user can instantaneously make course corrections. In terms of the performance metrics we have described for travel, a technique for exploration requires high levels of spatial awareness and information gathering. Ease of learning, ease of use, presence, and user comfort will also be important. Speed and accuracy are not requirements for such a technique.

Second, users may wish to travel to specific locations in the environment to obtain information. This type of travel has an explicit goal and direction, and is therefore unlike the exploration described above. It also has different requirements; in particular speed and accuracy will be quite important, since we do not wish to require the user to wait to get the desired information and we want the user to be able to move accurately to the location of the information. Since the user’s focus is on the destination, not the path, spatial awareness and information gathering ability *during* travel may not be as important. Such a technique will still require moderately high levels of ease of use and user comfort.

The application needs one or more techniques for selection, including a stand-alone technique to select audio annotations for playback, and a technique to select objects for manipulation in the immersive design component. These techniques may be the same, or they may be individually considered, as was the case with the travel techniques. It is more likely here that we can find a single selection technique to do the job, since the requirements for both tasks are similar. In general, we need a technique that can be used at a reasonable distance, and which is quite intuitive and easy so that users can focus on the task at hand. In terms of performance metrics, the application requires high levels of accuracy of selection, ease of use, and user comfort, with speed also being a main consideration.

Finally, we need one or more manipulation techniques with which to accomplish the immersive design tasks (moving visual elements, for example). We need expressive techniques which can be used to place objects at any location, but which are also well-constrained and easy to use. An additional consideration is that the manipulation technique integrates well with the selection and travel techniques that are chosen. Expressiveness (the range of positions and orientations in which an object can be placed), accuracy of placement, and ease of use will be the most important requirements for designers, and speed and user comfort will be secondary considerations.

6.3 Interaction Design

We will present three levels of interaction technique design for the Virtual Habitat application, which should provide us with some measuring sticks by which we can determine the effectiveness of our formal design and evaluation methodology. The first interaction design comes from a previous application and was based on a naïve understanding of the tasks and techniques involved. The second level of design was actually implemented and tested in the virtual habitat, and is based on an intuitive understanding of the tasks and techniques, and informal evaluations of published ITs. Finally, we present an interaction design based on the results of testbed evaluation.

By looking at the usability of these three designs, we should be able to ascertain whether the process of formal design and evaluation produced any performance advantages (hypothesis 3). We will show that our final design has significant advantages in performance and usability relative to the other two interaction designs.

6.3.1 Naïve Interaction Design

The first interaction design we will consider is taken from our initial attempt at an immersive modeling system: the Conceptual Design Space (CDS) (Bowman, 1996). This system was also aimed at architectural design, but differed in that it allowed the user to create objects from scratch or modify existing models. In terms of interaction, however, its requirements were very similar to the virtual habitat application. Users needed a travel technique to specify the viewpoint and the position from which they would design, a selection technique to specify objects for manipulation or for commands, and a manipulation technique with which objects could be moved or scaled.

The CDS system used a gaze-directed steering metaphor for viewpoint motion control. That is, the user looks in the direction he wants to move and presses a button. The main reason this technique was chosen was its availability: it was the default travel technique for the underlying VE software. We made one improvement to this basic technique by including a "walking" mode, in which the user was constrained to moving in the horizontal plane at the current eye height. This allowed users to obtain more human-scale views of the objects they were modeling.

This gaze-directed technique was frustrating to many users, because many of the movement tasks in a design environment are relative motion tasks, as described earlier. That is, the user is moving to a new location in the space from which a desired view of the object under consideration can be obtained. For example, the user may wish to view a building under construction in elevation from directly in front of the building. If the user happens to be closer to the building than desired, she must turn around and move away from the building, with no idea of when to stop. This leads to a long cycle of move-stop-evaluate-correct which can frustrate users quickly. The walking mode was somewhat useful, but the fact that it was an explicit mode that had to be turned on or off was problematic. Users typically did not wish to remain in one mode or the other for long periods of time, and did not wish to issue a command to change travel mode each time they wanted to switch. Thus, walking mode was underused.

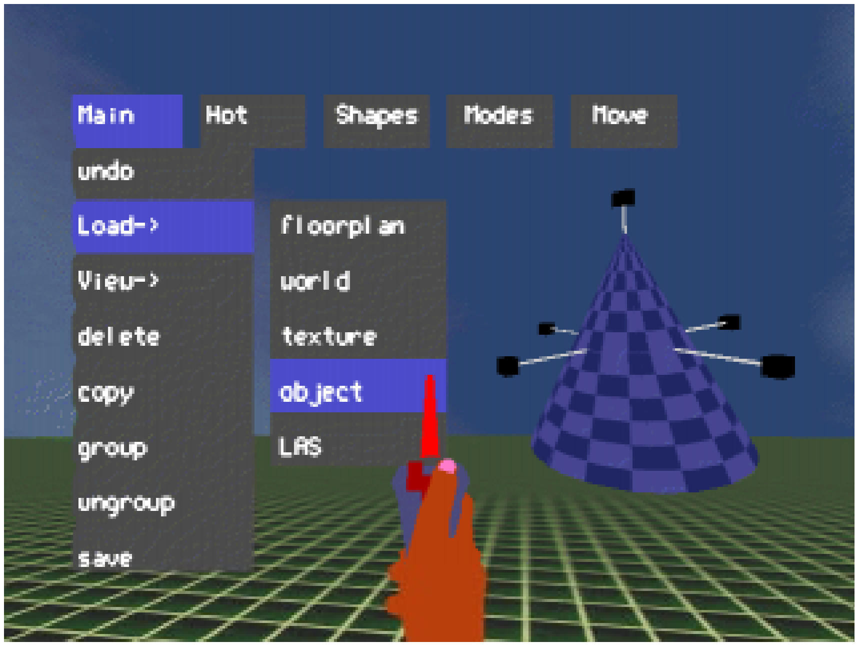

Figure 6.3 Virtual Menus in the CDS System

Selection and manipulation in CDS were based entirely on a ray-casting metaphor. A virtual light ray extended from the user's hand when a button was pressed. The light ray was used to select 3D objects, interface elements such as sliders and palettes, and object manipulation widgets. In addition, the ray was used to select items in the virtual menu system (see figure 6.3), which is similar to the one described in (Jacoby and Ellis, 1992). Menus contained commands for object creation, deletion, and copying, interface view commands, mode toggle switches, and so on. Objects could be manipulated directly with the light ray, or in a constrained manner using manipulation widgets attached to the object (figure 6.4). Depending on the mode, the user could translate, rotate, or scale the object in a constrained manner using these widgets.

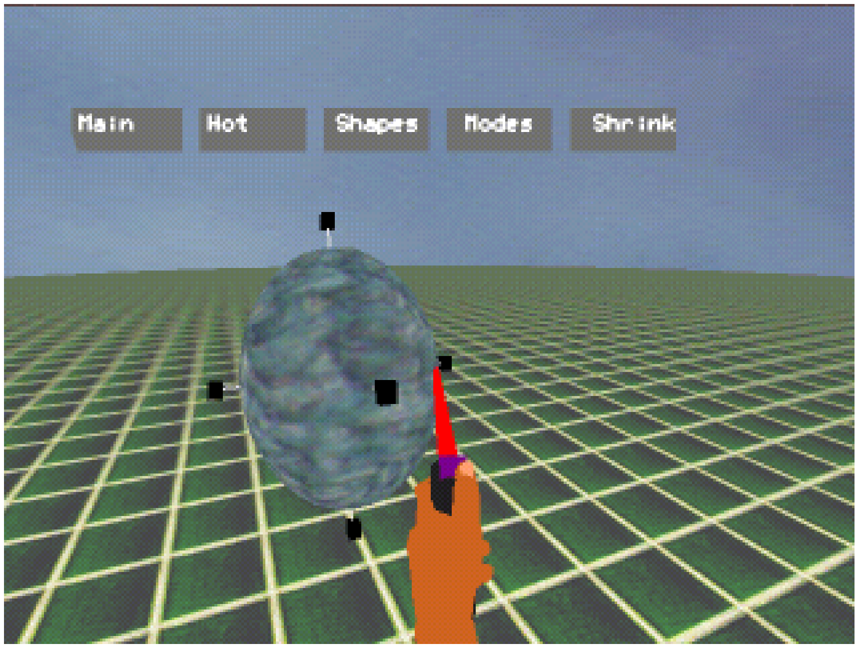

Figure 6.4 Constrained Object Manipulation in CDS with Ray-Casting

The ray-casting technique worked well in some areas, but fell short in others. Objects were easy to select, as were top-level menu items. However, items in submenus (which require precise pointing), and the small manipulation widgets were more difficult to hit with the ray. Manipulation of objects was quite imprecise when using the light ray directly, as we have already seen. Constrained manipulation was somewhat helpful, but getting an object into the desired position and orientation often took many attempts.

A usability evaluation with several graduate architecture students confirmed the advantages and disadvantages of this naïve interface. These users could see the promise of immersive design, with its immersive experience and immediate feedback, but were not very productive due to interaction issues.

6.3.2 Intermediate Design Iteration

Our second level of interaction design, based on experience, observation, informal evaluation, and the published literature, improved greatly on that of CDS. This was our initial design for interaction in the virtual habitat, which tried to provide many of the helpful constraints that were missing in CDS.

Just as virtual menus provided the system control infrastructure in CDS, we needed an overall system control scheme for the virtual habitat. We wished to avoid menus and explicit system modes based on previous experience and on general UI guidelines. Also, we wanted to avoid the imprecision of pointing in 3D space to select commands. To remedy this situation, we implemented a "pen & tablet" metaphor (Angus and Sowizral, 1995). This metaphor retains the advantages of using 2D interface elements in a 3D space (fewer DOFs to control, user familiarity, etc.), but also constrains the interaction so that it can be much more precise, efficient, and comfortable.

Figure 6.5 Physical Devices used in the Virtual Habitat Application

The physical input devices used in the pen & tablet interface are shown in figure 6.5. They consist of a physical tablet and a physical pen (or stylus), both of which are tracked in 3D space. The pen also has a single button. The tablet does not contain any electronic logic or have any display – it is simply a work surface. In the virtual environment, the user sees graphical representations of the pen and tablet, and a 2D interface is presented on the tablet surface (figure 6.6). The user interacts with this interface just as he would with a 2D interface controlled by a mouse, except that the pen is placed directly on the interface whereas a mouse indirectly controls a pointer on a screen. The interface can include menus, buttons, icons that can be dragged, and so on.

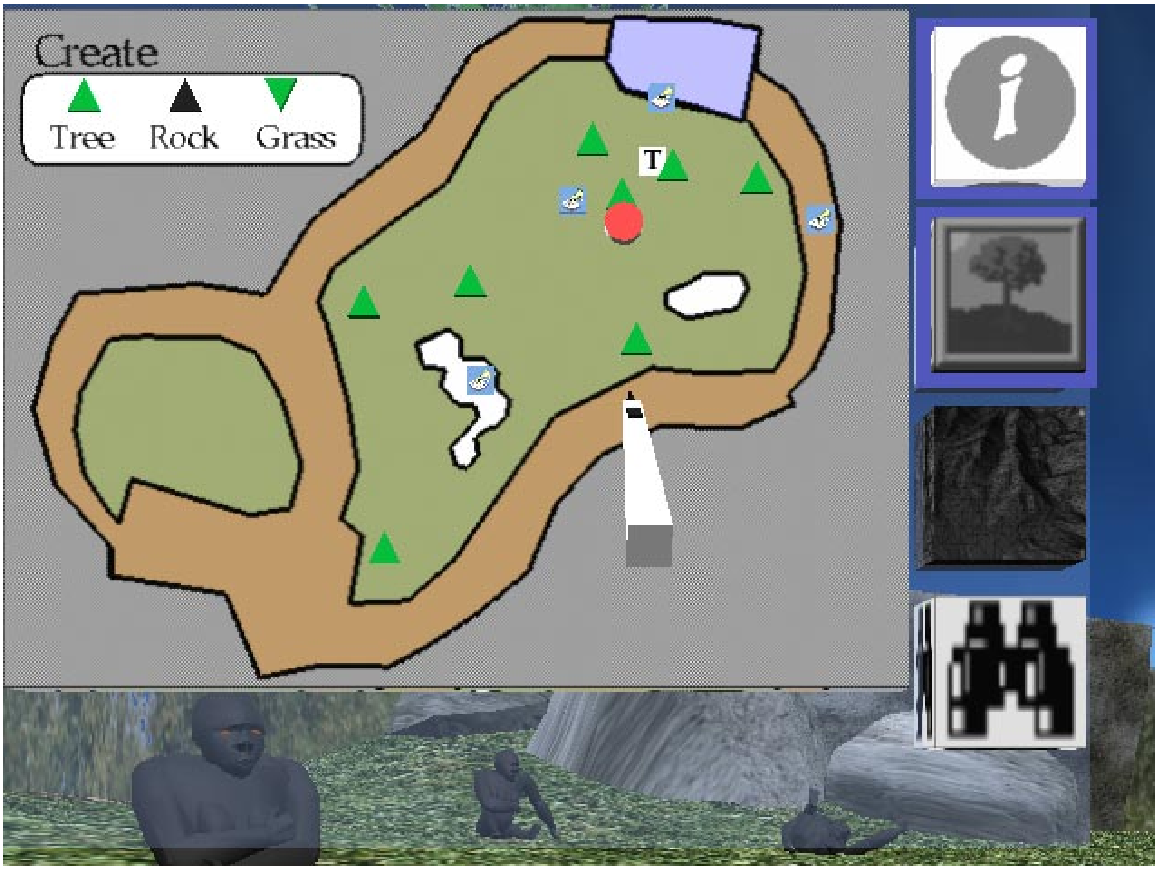

Figure 6.6 User's View of the Interface Tablet in the Virtual Habitat

The advantages of the pen & tablet metaphor are many. First, the interface is always available since the user carries it in her hand, but it can also be put away so that it does not obscure the environment (simply by placing the tablet out of the field of view). Second, the physical surface of the tablet provides an important constraint to input. Instead of pointing or gesturing in 3D space, with no guidance, the user can be assured of correct interaction as long as the tip of the pen is touching the surface of the tablet. This makes operations such as icon dragging much more precise and sure. Finally, this metaphor makes use of 2-handed interaction (Hinckley et al, 1997), where the non-dominant hand provides a frame of reference within which the dominant hand can work. This has been shown to be an efficient and effective method of 3D input.

With the pen & tablet metaphor as a basis, we began to design specific interaction techniques for the virtual habitat. Our design philosophy was to provide both tablet-based (indirect) and direct manipulation techniques for each of the major interaction tasks.

In the area of travel, we wished to support both exploration and goal-based motion, as discussed previously. For exploration, a directly controllable technique was needed. Instead of the gaze-directed technique used in CDS, we chose a pointing technique, in which the user points the stylus in the desired direction of travel. In this way, relative motion was supported, which is important for a design application. Goal-based travel was achieved on the tablet. A red dot represented the user's current position on the map of the environment which was the main feature of the tablet interface. To move quickly to a new location, the user could drag this icon to a new location on the map. The user was not moved as the drag takes place. Rather, to promote spatial awareness, the user only moved when the dragging had ended, at which time he was flown smoothly from the current location to the new one. Instead of an explicit walking mode, we chose to allow users complete 3D freedom of motion, except for a constraint on going below a certain height above the ground. Thus, users could simulate walking mode by simply pointing slightly downward, so that they traveled along at a constant height above the terrain.

Selection and manipulation of virtual objects could also be performed both directly and via the tablet. The direct technique chosen was the Go-Go technique described earlier, in which the user's virtual arm length grows at a non-linear rate as she stretches it away from her body. This allowed the user to select objects (such as trees or rocks) at a large distance, but with little cognitive load, as it is natural to stretch out one's arm to grab an object. Manipulation could then take place in the virtual hand. This type of manipulation supports more precise placement of objects. For coarse-grained placement, object icons on the tablet interface could be dragged to new locations. This was useful, for example, to create a grove of trees in one corner of the environment. We also greatly constrained manipulation to make it easier for the user. Objects always remained on the ground, and the user had no control over object rotation, since all of the objects we wished to manipulate have a natural orientation. Thus, the user was only manipulating two degrees of freedom, which matches nicely with the 2D tablet input.

There are also other selection tasks in the Virtual Habitat application. Only the Go-Go technique was used to select audio annotations for playback, as we did not wish users to be able to play annotations from anywhere (so they could experience the information in its proper context), but the tablet can be used to enable or disable specific annotations. The tablet was also used to toggle the display of various types of information on the 2D interface, to create new objects (by dragging icons onto the map), to position visitor viewpoints, and to select the terrain model. All of these are tasks which are more easily performed indirectly and/or in 2D.

A usability study was performed on this initial version of the virtual habitat application, and although it was rated quite highly, there is still room for improvement. Six teams of students used the application to modify the design of the virtual habitat for a class presentation. The usability study confirmed the usefulness and promise of immersive design, but more importantly for our research, provided us with a set of user ratings on various aspects of interaction. Users were asked to rate usability issues on a scale of one to five, with five being the most usable. A summary of the results is presented in table 6.1.

Table 6.1 Mean Usability Ratings for the Intermediate Virtual Habitat Interaction Design

Usability Categories	**Rating**
tablet: object creation	4.43
tablet: dragging user icon to move	4.21
changing terrain	4.21
moving viewpoints	4.20
moving viewpoint barriers	4.10
tablet: general interaction	3.86
tablet: object manipulation	3.86
user movement with stylus	3.71
go-go object manipulation	3.14

The entries in this table reveal some interesting trends. First, notice that when there was a choice of interaction technique (one using the tablet and one using direct, 3D manipulation), the tablet-based technique was preferred. For example, dragging the user icon on the tablet to travel to a new location in the environment was preferable to pointing in the direction of travel using the stylus. This stems from the advantages of the tablet mentioned earlier: it is always available, it has a physical work surface to constrain input, and it requires the user to control only two degrees of freedom. However, the use of the tablet also caused some problems for users, most notably due to orientation differences between the map and the environment. Some users found it difficult and disorienting to drag the user icon in one direction and then move in a different direction, or to drag an object on the tablet to the left and see it moving to the right in the virtual world. Most users were able to adapt to these difficulties by focusing on only one context at a time, and by noting relationships between object positions instead of absolute locations. For example, a user viewing the environment might decide to move a tree to the left. To make it a relative positioning task, he would translate the goal to something like "move the tree closer to the visitors building." Using this goal, either the tablet or direct manipulation methods would work well.

Feedback on the direct manipulation techniques was mixed. Some users found it natural and intuitive to point in the direction they wished to fly, and enjoyed the simplicity and flexibility of this technique. Others became disoriented when they moved in a direction other than the direction of their gaze, and could not point as accurately as they hoped.

The Go-Go technique for object manipulation fit the intuition of most users: to move an object one simply reaches towards it. However, there were difficulties due to the size of our environment. In order to allow users to reach most of the environment, the non-linear portion of the Go-Go stretching function (see Figure 5.1) needed to be quite steep. This meant that when the virtual arm was far from the user's body, a very small movement in or out would result in a large virtual hand movement. This made object selection difficult at large distances.

This usability evaluation was performed before the spatial orientation experiment and both of the testbed evaluations. Interestingly, however, these experiments would have predicted most of the major usability problems found here. Disorientation was quite harmful to users, because we used a steering technique without advising users of the proper strategies for maintaining spatial orientation. An arm-extension technique was used for object selection, which we showed empirically to be quite slow and tedious. This technique also exhibited the arm-strain characteristic that we found in our testbed evaluation.

6.3.3 Final Interaction Design

Our final design for the interaction techniques and metaphors used in the Virtual Habitat is based on the results of formal design and evaluation. Results from testbed evaluation have been applied to this system based on its requirements to show the usefulness of our methodology of formal evaluation and design. It is important to note that although we noted many other minor usability problems in our first evaluation of the Virtual Habitat, we left these things unchanged for the final iteration. The only differences in this version and the previous one are the changed techniques for travel, selection, and manipulation based on our formal evaluation. This is to ensure that any gains in usability are due to the application of our methodology, and not to other interface modifications.

The results of the travel testbed (section 4.7) showed that our intermediate design iteration actually met the application's performance requirements well. We found that speed and other metrics on both the exploratory and the directed travel tasks was best with continuous steering techniques, such as pointing. Although this was intended in the previous design iteration to be used for exploration, it appears to be well suited to the performance requirements of the goal-directed travel task. User comfort was not a major factor in the testbed experiment, but the pointing technique performed well in this category.

In our previous usability study, the map dragging technique was rated subjectively higher than the pointing technique. However, we noted some problems with it, and these problems were verified in the testbed evaluation. Most notably, users often did not know which direction to drag the user icon in order to move to a given location. In the usability study, we found that certain users were better with the map technique than others, and hypothesized that these people were able to do the mental rotations of the map necessary to determine direction. Therefore, we left the map dragging technique in place in the final design, but only encouraged users to utilize it after they are quite familiar with the spatial layout of the habitat.

A related usability problem that we found in the intermediate design iteration concerned the loss of spatial orientation on the part of users. Users often became lost or disoriented, especially after using the pointing technique to fly in a direction other than that of their gaze. Some users also had difficulty relating the static map information to the dynamic environment. These are exactly the concerns addressed by our spatial orientation experiment in section 4.6.3. In that evaluation, we found that subjects who used advanced strategies for maintaining orientation had the best performance. Therefore, in the final design iteration for this application, we modified our written and verbal instructions in order to train users in these strategies. Strategies relevant to the Virtual Habitat include 3D overview (fly up above the environment to get a survey view), backing in (moving backwards to a destination so that it is placed in the context of previously visited areas), proprioceptive pointing (reminding oneself of the location of known objects), stop & look (pausing to look around at the current location), and path retracing (moving again along previously traveled paths, often from a different direction). Users are not likely to use all of these strategies, but using one or more of them could increase spatial orientation.

The selection and manipulation testbed confirmed our informal observations of the Go-Go technique. It is not well-suited for selection of objects that are small and/or far away. Moreover, it was the lowest rated of the techniques in our usability study, due to the frustration people had with selecting distant objects. The testbed results showed that the HOMER technique was the best fit for the performance requirements specified above for selection and manipulation. It can select objects well at long distances, and ray-casting is quite easy to use and speedy. The manipulation component of HOMER is very expressive and also easy to use and moderately fast, according to the empirical results. HOMER was not near the top of the rankings for manipulation time in our study, but as stated above, speed of manipulation is not a key performance requirement of the Virtual Habitat.

Having chosen these techniques for our final implementation, we were faced with another problem: the stylus has only a single button, but both the pointing technique for travel and the HOMER technique for selection and manipulation would need that button. We implemented a solution that we felt would be easy for users to understand and use. Our implementation changes the use of the button depending on how long it is held down. The light ray is visible at all times, and objects are highlighted when intersected by the ray. If the user clicks the button (down and up) quickly (less than 0.7 seconds) and an object is highlighted, that object is selected. If the user holds down the button for more than 0.7 seconds, the ray disappears and the user begins to travel using the pointing technique. The single button also precluded us from using the indirect depth manipulation technique we studied in our experiment.

Several lessons can be gleaned from this design iteration. First, the technique that users prefer is not always the one with the best performance. Users preferred the map technique to pointing, but empirical evaluation showed pointing to be faster. Fortunately, we could include both techniques in our application. Also, when attempting to support better performance by using empirically proven interaction techniques, the tradeoffs and difficulties of integration must be taken into account. In our case, the usability problems with the intermediate iteration were severe enough that we were willing to work through the integration problems to solve them.

6.4 Final Usability Evaluation

When the interaction design was finalized, a new usability study was performed under similar circumstances and using the same evaluation metrics (interviews and usability ratings). In this way, we compared the usability of a system designed using intuition and observation to that of a system implemented based on formal evaluation and design methods (hypotheses 1 and 3). This study would validate the use of our formal design and evaluation methodology if increased performance were found.

Five user sessions were held, lasting for sixty to ninety minutes each. During the session, the users were instructed on the use of the techniques, allowed to explore the virtual habitat, shown how to access the information embedded in the environment, and presented with the design tools. Each user or group of users (users came singly or in a group of two) spent twenty to forty minutes using the design tools to modify the design of the gorilla habitat. Subjects were members of an undergraduate design class with experience in both traditional and computer-aided design. At the end of the session, each user or group was asked for their comments and observations on the system, as well as a set of usability ratings on the various features of the application. These ratings again were on a five-point scale, with five representing high usability. A summary of the results is presented in table 6.2, including average usability ratings and standard deviations for each of the system's features.

Table 6.2 Mean Usability Ratings (standard deviations in parentheses) for the Final Virtual Habitat Interaction Design

(features changed since the previous iteration)*

Usability Categories	Final iteration	Intermediate iteration
selecting annotations*	4.70 (0.45)	N/A
changing terrain	4.20 (0.76)	4.21 (1.15)
user movement with stylus*	4.10 (0.89)	3.71 (1.11)
tablet: dragging user icon to move*	4.10 (0.74)	4.21 (0.81)
direct object manipulation*	4.00 (0.35)	3.14 (1.18)
tablet: object creation	4.00 (0.71)	4.43 (0.53)
moving viewpoints	3.55 (0.94)	4.20 (0.84)
tablet: object manipulation	3.50 (1.00)	3.86 (0.94)
moving viewpoint barriers*	3.40 (1.39)	4.10 (1.02)
tablet: general interaction	2.90 (0.89)	3.86 (0.90)

The most important result from table 6.2 is that our application of the results of formal design and evaluation had positive results on reported usability. This is most easily seen for the direct object manipulation feature, which was changed from the Go-Go technique to the HOMER technique, and which received a much higher usability rating in the final iteration. This is despite the fact that this group of users seemed to have a lower baseline rating overall (for all unchanged components, the average usability rating was lower than the corresponding rating from the intermediate iteration. Also, ray-casting proved to be very easy to use as a selection mechanism for the audio annotations, receiving the highest rating of any feature. Although we did not measure the usability of the Go-Go technique for annotation selection in the previous iteration, it was the source of many verbal usability complaints by users.

Second, we note that the reported usability of the pointing technique was improved in the final iteration. Although the implementation of this technique did not change, the training given to users in the proper use of this technique was modified. Both written and verbal instructions were given to users telling them how to use this technique to maintain spatial orientation (e.g. flying upwards to get a survey view of the environment). This result validates our earlier finding that the training of specific strategies can have an effect on overall performance.

The map dragging technique for travel was rated highly, but slightly lower than the rating from the previous iteration. Again, this is consistent with other features that remained unchanged. Therefore, the additional training in strategies for spatial orientation did not increase the usability of this technique, again validating our earlier findings. Strategy sophistication can increase performance with steering techniques, but performance using target-specification techniques is relatively constant no matter what strategies are used. Also, fewer of the strategies are possible when using the map-dragging technique.

The comment of one subject is particularly enlightening with regard to the travel techniques used in this system. Although the map technique performed poorly in the testbed evaluation and is not useful on its own, it can be a good complement to a steering technique. The subject stated that he would not rate the map technique highly, except that it worked well in conjunction with the pointing technique. This leads to the general principle that multiple, redundant interaction techniques should sometimes be used to improve usability.

The only disappointment in this study was the use of HOMER to manipulate the foliage barriers at visitor viewpoints, which received a very low rating. From user comments, we feel this was due to the nature of the task. The barriers are very close to the user at the viewpoints. Since HOMER maps the body-hand distance to the body-object distance to determine the mapping between hand and object motion, near objects are difficult to move farther away. With the Go-Go technique, the same depth range can be accessed regardless of the original object distance. An indirect depth specification scheme using buttons would solve this problem, but is not possible with our single-button stylus.

The use of subjective ratings to measure usability is somewhat problematic, as we have no measure of the validity or reliability of this metric. The same is true for the comfort ratings used in the testbed experiments. We do have information on the variability of these ratings, which seems to be reasonable, but the results would be more powerful if usability or other types of performance had been measured with a proven metric, whether quantitative or qualitative. We leave the development of such a metric as future work.

On the whole, this usability study provided an unequivocal endorsement of our methodology. The use of the formal design and evaluation framework, testbed evaluation, and application of results based on performance specification caused a measurable increase in usability, supporting hypothesis 3.