Evaluating the Usability of Crumbs: A Case Study of VE Usability Engineering

Kent Swartz[1], Umesh Thakkar[2], Deborah Hix[1], Rachael Brady[2]

[1]Department of Computer Science, Virginia Tech, Blacksburg, VA 24061
[2]NCSA, University of Illinois at Urbana-Champaign, Urbana, IL 61801

Abstract

The everyday use of virtual reality systems is creating a need for user interface design guidelines for application developers. Programmers are beginning to develop and distribute user interface toolkits. As a result, de facto standards are emerging. The technology has matured to a point that usability studies are necessary in order to understand the strengths and weaknesses of different interaction techniques. Unfortunately, formal methodologies for designing and evaluating usability of three-dimensional interactions inherent in virtual environments are only just emerging. In this paper, we evaluate a two-phase usability engineering methodology by applying it to the Crumbs CAVE™ application. We report on the effectiveness of the methodology as well as the usability issues discovered in the Crumbs program.

Keywords: Virutal Environments, Usability, Usability Engineering

1. Introduction

The proliferation of projection-based virtual reality systems is creating a need for standardized user interface paradigms. Researchers at Virginia Tech are examining traditional usability engineering methods to determine the modifications necessary to make them applicable in the design and evaluation of virtual environments (VEs). VEs lack a standard interaction style, such as the graphical user interface (GUI) window, icon, menu, pointer (WIMP) paradigm. Due to the innovative and complex nature of VEs, it is not presently clear that limiting VE development to standardized interaction styles is even desirable. In fact, most research on VE interaction methods focuses on categorizing specific styles for use in specific tasks [Bowman98][Gabbard97][Kaur97]. This lack of conformity in interaction style combined with the continual introduction of new I/O devices places more responsibility on VE application developers. We believe that early incorporation of usability assessments into the user interface design process will ensure that the substantial amount of resources required to create a VE are not wasted on producing an unusable system.

The next section introduces the main usability assessment methods proposed in the two-phase usability engineering methodology. Section 3 describes the Crumbs CAVE application along with our motivation in choosing this program for evaluation. The application of the methodology to the Crumbs program are detailed in section 4. The results of the user study are listed in section 5. We conclude with reflections on the effectiveness on the methodology and lessons learned during the case study in sections 6 and 7.

2. Usability Engineering Methodology

The method proposed for understanding the usability of a VE application involves two phases of evaluation. The first phase involves one or more usability specialists performing an overall inspection of the application [Nielsen94]. The inspection applies usability design guidelines currently under development at Virginia Tech. Using these guidelines the specialist locates obvious usability problems and recommends a redesign before introducing end users into the study. The second phase, or formative evaluation, involves placing representative users into task-based scenarios. These scenarios are comprised of specific benchmarks with predefined metrics.

2.1 Usability Inspection

Usability inspection techniques focus primarily on the interaction style of the application. They are designed to uncover usability violations in generic methods used for interaction between users and applications. They are usually not designed to address domain-specific areas of usability. One particular method of usability inspections is heuristic evaluation. Heuristic inspection is a usability assessment method in which one or a group of interaction specialists evaluate a particular user interface in depth to determine if it conforms to an established set of usability design guidelines. The specialists report any discovered

---

™ CAVE is a registered trademark of the Board of Trustees of the University of Illinois violations and use a user-centered perspective to rate the severity. The case study of Crumbs uses a heuristic evaluation based on the guidelines enumerated in the *Framework of Usability Characteristics* [Gabbard98].

The *Framework of Usability Characteristics* provides several VE related usability resources including a comprehensive set of usability guidelines, detailed discussion of the guidelines, and assistance in locating auxiliary references. The guidelines are an organized presentation of multiple VE research efforts gathered from various sources including but not limited to: World Wide Web searches for related work, VE-related journals and conferences, investigative research visits to some of the top VE research facilities, and human-computer interaction related literature. The framework is organized into four separate sections each associated with a specific component of VE interaction.

- VE Users – general user and task characteristics and types of tasks in VEs
- VE User Interface Input Mechanisms – usability characteristics of VE input devices
- The Virtual Model – usability characteristics of generic components typically found in VEs
- VE UI Presentation Components – usability characteristics of VE output devices

Each section contains guidelines and in depth discussion. There are 195 total guidelines between the sections.

2.3 Formative Evaluation

Empirical evaluations are one of the most popular methods of usability assessment. Empirical evaluations involve usability specialists observing representative participants as they complete a predetermined set of typical tasks while interacting with an application. The evaluation process is recorded either by videotape or written account. This historical account of the evaluation provides valuable information regarding the participant's confusions, errors, complaints, and other experiences. Quantitative measurements taken during empirical evaluations normally include task accuracy and time for task completion. User-based evaluations, in comparison to usability inspections, do address domain-specific aspects of an application. User-based evaluations focus on domain-specific tasks, and tasks should be chosen that are representative of tasks that users will be performing within the domain. In most cases, participants should possess domain knowledge and therefore are able to evaluate these specific aspects of the application.

Formative evaluation is one particular form of empirical evaluation where usability assessment involving user observation happens early and often in the usability engineering life cycle [Hix93]. The main purpose of formative evaluations is to continually improve application usability through iterative user observational studies. The iterative nature of formative evaluation assists the development team in uncovering usability problems early and determining a plan for redesign. Formative evaluations are often regarded with the misconception of resulting in minimal data taken on few participants. However, experienced usability specialists can collect enormous amounts of both quantitative and qualitative data using this technique. Although the data are usually not analyzed through a process resulting in statistically significant results, they do provide quantitative results necessary to compare interaction techniques and qualitative results to uncover usability problems.

2.4 Discussion of Methodology

Combining usability inspections and empirical usability evaluations is not a new idea. Usability inspection techniques should be supplemented with user-based evaluation because each technique discovers usability problems that are often overlooked by the other [Nielson94][Mack94][Nemire93]. A typical strategy is to apply a usability inspection method first to clean up the interface as much as possible, then to subject the revised design to the user [Mack94]. Because VEs lack standardized interaction styles, combining usability inspections and empirical evaluations is encouraged to provide an efficient and effective usability engineering strategy. VE usability specialists are familiar with interaction taxonomies and can, during inspection, potentially remove many obvious usability violations allowing empirical evaluations to focus more on domain tasks. This improves the chances of uncovering domain-specific usability issues because participants do not have to deal with awkward or ineffective interaction techniques. Inspectors performing heuristics evaluation can also highlight potential usability problems and recommend formative evaluation plans to incorporate tasks to assess the usability of these potential problems. A single iteration of an effective usability evaluation would include a heuristic or guideline-based technique by one or more VE usability specialists coupled with a formative usability evaluation with end-users. [Hix99] demonstrated success assessing the navigation metaphor used in a real-time battlefield visualization virtual environment using a similar approach of iterative applications of heuristic inspection followed by formative evaluation.

3. Crumbs

Researchers at Virginia Tech are working in collaboration with the NCSA Biological Imaging Group to apply this usability evaluation technique to Crumbs, an immersive visualization VE developed at NCSA. Crumbs runs in a CAVE and is used as a tracking tool for biological and medical imaging. Crumbs' developers want to leverage advantages of visualization and immersion to facilitate identification of complex biological structures [Brady95]. Crumbs provides users with a method of visualizing dense volumetric datasets in an immersive VE. Users are also given the opportunity to mark and measure individual structures in the dataset. Crumbs provides users with environmental objects, 3D widgets, and menus to interact with the system. The primary method of user interaction is by means of the CAVE default input device, the wand. The wand is a 6 degrees of freedom device that is used in Crumbs to provide similar functionality as the mouse in the standard GUI paradigm. The wand is used to manipulate the volumetric dataset, visualization tools, marked points as well as navigate pop-up menus.

Crumbs also supports the use of voice activation as a method to invoke menu commands and control the VE. Crumbs has several objects that assist the user in visualization, marking, and measurement tasks. Below is a list of the most important of these objects and a description of their job responsibilities.

- data volume object – A box containing the volumetric data set loaded into Crumbs.
- crumb object – An object used to mark specific structures in the data set. A curve drawn in between serially created crumb objects creates a line segment referred to as a fiber.
- clearbox object – A box of a specified width used to render specific regions of the data volume at a specified resolution.
- sliceplane object – Visualizes a single slice in the data volume at a specified resolution.
- trashcan object – Object used to delete other objects. Deletion is facilitated by dragging objects into the trashcan.
- colormap object – A 3D widget allowing the assignment of specific density values of the data set to specific colors. This widget relies on a color cube metaphor.
- contrast enhancement object – A 3D widget performing two functions. First it allows restriction of viewable density values in the data volume. Secondly it provides a visual representation of the color to density mapping currently supported by the colormap object.
- opacity object – A 3D widget allowing specification of opacity values for specific density values of the data set.
- sword – Metaphor used to describe the physical appearance of the wand in the virtual environment. The sword in used in a similar fashion to a GUI cursor.

4. User Study of Crumbs

Modification to an existing usability evaluation technique led us to a usability inspection of Crumbs in the spirit of Nielsen's heuristic evaluation method [Nielsen94]. However, due to the added complexity of VEs, the evaluation required more than Nielsen's ten heuristics. Currently, a standard set of heuristics for use in the inspection of VEs does not exist, but the set of VE usability design guidelines contained in the framework provides a reasonable starting point for usability evaluation [Hix99]. (Note: short lists that are used for heuristic evaluations are referred to as heuristics, while long lists are referred to as guidelines [Gray98]). These guidelines were systematically applied to Crumbs as a heuristic inspection in October 1998.

The developers of Crumbs were constrained to using a CAVE, a spatially immersive environment, and a wand, a hand held isotonic input device. Due to these constraints, many guidelines located in the VE UI Presentation Components and VE User Interface Input Mechanisms were not applied in this inspection. The inspection concentrated more on application interaction issues and exhibited more effort in evaluation of the Users and User Tasks section as well as the Virtual Model section.

4.1 Phase One – Heuristic Evaluation of Crumbs

The heuristic evaluation of Crumbs followed the theory of Bowman (1998) which states that at the lowest level VE tasks can be broken into three categories: locomotion, object selection, and object manipulation. Locomotion is the task of interactively moving the viewpoint within the environment. Object selection is the task of selecting one or more objects in the environment. Object manipulation is the ability to act upon currently selected objects. Two other tasks that sit on top of these core tasks are system commands and domain-specific tasks. Crumbs' usability inspection focused mostly on how Crumbs facilitated system commands and domain-specific tasks using object selection, and object manipulation. Locomotion was not addressed since Crumbs uses object manipulation as its navigation paradigm.

Table 1 summarizes some of the usability problems and possible redesign recommendations as given to developers of Crumbs. Fifteen problems were identified and following are sample issues. The evaluator worked in cooperation with the developer to incorporate redesign recommendations into the Crumbs.

Table 1: Guidelines-based Usability Issues and Redesign Suggestions

Framework Guideline	Usability Issue	Redesign Recommendation
Strive to maintain interface consistency across the application.	Crumbs is not consistent with its use of the term "crumbs" within the menu system. Identical items are also referred to as "Points".	Make the interface consistent using either "crumbs" or "points".
Language and labeling for commands should clearly and concisely reflect	Crumbs' menu system uses language such as "Toggle Spline" with no symbolic representation whether the	Simply using an icon to represent when a certain utility is in use would potentially

Framework Guideline	Usability Issue	Redesign Recommendation
meaning.	spline is currently in use or not. This mandates the user to know the state of Crumbs prior to entering the menu.	eliminate this added task.

Crumbs developers decided not to act upon some reported usability violations. This decision was based mostly on size of the redesign effort. Table 2 lists a couple of the possible usability issues developers chose not to redesign.

Table 2: Guideline-based Usability Issues Not Redesigned

Framework Guideline	Usability Issue	Redesign Recommendation
Pay close attention to the visual, aural and haptic organization of presentation (e.g. eliminate unnecessary information)	Crumbs permits the attempt at execution of the "Toggle Spline" menu item when there are less than two crumbs placed. This should be "grayed out".	Only allow the user to execute the commands currently available.
Strive to maintain interface consistency across the application.	A direct manipulation technique, dragging an object to the trash can object, is used to delete objects. However, an indirect manipulation technique, selecting a menu item, is used to exit or close objects.	Offer the user a direct manipulation method of closing objects.

The guideline-based inspection also identified several possible usability issues that the inspectors and developers decided to postpone for redesign consideration until the issue was validated as a problem in formative evaluations

Table 3: Guidelines-based Usability Issues Requiring Further Evaluation

Framework Guideline	Usability Issue	Recommendation for Further Evaluation
Take into account user experience (i.e., support both expert and novice users).	Use of the opacity object is not initially intuitive and requires training for proper use.	Design a task in the formative evaluation to assess usability of the opacity object.
Provide stepwise, subtask refinement including the ability to undo.	Placing crumb objects is a stackable task and Crumbs should consider a quick method of unstacking (i.e. undo).	Design a task in the formative evaluation to determine if the user desires an undo capability.

4.2 Phase Two – Formative Evaluation of Crumbs

After modifications suggested by the usability inspection were completed, we performed a formative evaluation of Crumbs as described in detail in the following sections. A set of tasks were chosen which were representative of real world Crumbs use by scientists. Furthermore, tasks were added to assess usability issues either reported as violations or possible problems in the guideline-based inspection but not corrected prior to the evaluation (e.g. Tables 2 and 3). Formative evaluation of Crumbs took place at NCSA and included collaboration of VE usability specialists from both NCSA and Virginia Tech. This initial evaluation served a dual purpose, as a pilot test to iron out potential difficulties in evaluation procedures, and also as an evaluation of Crumbs using domain expert participants.

4.2.1 User Tasks

Three sets of tasks were used for the evaluation. Two sets concentrated on tasks used to visualize and measure structures located in a single Crumbs dataset, while the third set tested the sonification implementation of Crumbs. The first set of tasks used a spiral CAT scan of a human spine. The spine data volume contains seven vertebrae of the lower back of a human male. The task was to position seven markers at the three dimensional centroid of each vertebra. This 'negative' task of placing the marker in a hole requires the user to select the size and resolution of the imaging tools to maintain sufficient detail while still viewing the global context of the hole boundaries. The second set of tasks analyzed a confocal microscope image of the tail of a fruit fly sperm. The sperm tail data volume consists of a one-dimensional strand that is curving and wrapping around itself in three-dimensional space. The user was presented with a partially traced tail, and their main task was to complete the tracing. The user needed to position him/herself and viewing tools in such a manner that the location of the tail was easily visible. Then the user placed markers along the tail in the highest density regions of the strand. This task assessed whether the user could place a marker in a 'positive' location; that is, if the user could place a marker within a dense region where the feature being traced was specifically visible. The third set of tasks addressed the validity of Crumbs sonification in representing various values of volumetric density. The user was given aural stimuli that Crumbs uses to represent various density values and then prompted to determine the stimuli representing either the highest or lowest density.

4.2.2 Participants

This evaluation targeted expert participants with either a working knowledge of Crumbs or domain expertise on the datasets used in the evaluation tasks. From prior use of Crumbs, the participants had experience interacting within an immersive three-dimensional VE and required little or no training on the application. Although each participant varied in his/her individual application experience, VE experience, and domain experience, their familiarity with either the application or tasks provided enough consistency to validate categorizing them as experts. Five people chose to participate in the formative evaluation. The first participant was used as a pilot test.

4.2.3 Equipment

All evaluation sessions occurred in the NCSA CAVE. A stopwatch was used to record the elapsed time for participants to complete certain tasks. During the pilot test, the CAVE had two concurrent streams of video recording the evaluation. One stream used a scan converter to record the front wall of the CAVE while the second stream used a video camera to record the participant's actions. In combination with the video camera, a wireless microphone was used to record the participant's comments. Because the front wall of the CAVE is displayed in stereo, recording of the front wall proved to be less useful than hoped for, and therefore was dropped from the evaluation following the pilot study.

4.2.4 Evaluation

A pilot study preceded the formative evaluation to uncover possible flaws in tasks, methods, and equipment proposed for evaluation usage. One key flaw was uncovered concerning physical setup of the pilot study that could negatively affect formative evaluations. Positioning evaluators within the CAVE introduced negative influences that were not expected. Some of the problems encountered included 1) the evaluator interrupting a task by standing in a physical location that is required for participant interaction 2) the evaluator's feet got tangled in the cords needed for the tracking system and 3) mentally interfering with the participant's feeling of presence by their close physical proximity. Most of the formative evaluation sessions were conducted with three evaluators. One evaluator, who also functioned as the mediator, collected data on errors. Another evaluator controlled video recording and collected data on participant comments. The final evaluator recorded elapsed completion time for each task. All three evaluators were located outside the CAVE.

5. Results of Crumbs Formative Evaluation

Quantitative data consisted of task completion times, task error counts, and questionnaire scores. Data analysis was further broken down into awareness, manipulation, presence, and selection. Some domain tasks are a combination of selection and manipulation and this fact is taken into account during data analysis. This study chose to focus mostly on qualitative data recorded in the form of critical incidents to discover usability issues. Quantitative data were collected, but used mostly to support qualitative findings.

In this section we present four kinds of results: the followup questionnaire, the effectiveness of the sonificaiton implementation, the usability issues discovered during the data analysis tasks, and participant suggestions during the course of the evaluation.

5.1 Follow-up Questionnaire

Participants were asked to complete a questionnaire following evaluation completion. The questionnaire allowed participants to rate their judgements on a scale of one to four. Table 4 reports the results of this questionnaire.

Table 4: Follow-up Questionnaire

Question	Average Score
Rate your overall satisfaction with the application.	2.875
Rate the usability of the menu system.	3.25
Rate the effectiveness of the sonification in assisting crumb placement.	2

How easy was the opacity object to use?	2.25
How easy was the colormap object to use?	1.5
Rate the use of the sword for selection.	3.25
Were the objects as a group easy to manipulate?	2.375
Did you feel as if you were a part of the environment (as if you were a physical entity in the application)?	2.75
Did you get disoriented in the data volume (were you ever unaware of the data volumes x, y, z axis)?	3.125
Rate the effectiveness of the audio annotation.	2

As seen from the results of the first question, participants rated their overall satisfaction of Crumbs above average. The other questions show some possible usability problems, but these questions were included because of known weaknesses of the Crumbs application prior to testing.

5.2 Effectiveness of Sonification

Assessing the benefits of the sonification in Crumbs was a result requested by the designer. Tasks were developed to execute identical structure marking. The first iteration through the task set included the sonification. The subsequent execution of the task set was completed with sonification removed. It was hypothesized that if traditional task measurements of error count and task completion time was better for the first execution of the task set, then sonification was beneficial. However, the benefit of sonification was not evident in the sperm tail marking tasks. The two participants that completed the task did not appear to rely on sonification at all. Sonification was designed in Crumbs as a three part activity corresponding to the creation and positioning of a crumb. The interlude is designed to signify creation of the crumb. Following the interlude, sonification changes according to the region density currently pointed at by the sword. Finally, once the crumb had been "dropped" at a location, the sonification concludes with a postlude [Brady96]. For a majority of the crumbs created, both participants finished positioning an individual crumb prior to interlude completion. Therefore, they were unable to use sonification for any benefits. One participant stated his opinion following the evaluation saying, "I do not like the sonification". This participant proved to be one of the more skillful users of crumb object manipulation. A different participant that did not finish the task attempted to use sonification to assist in crumb placement, but expressed that they were unaware of the target sound and that the "sound is the same everywhere". This response leads the authors to believe that more training was needed prior to the sonification task.

The formative evaluation also contained five individual tasks that concentrated on how well sonification mapped to changing density. Participants were asked to differentiate densities from two or more audio outputs created by the sonification. Participants used phrases such as "I guess" and "not obvious" in voicing their answers to the tasks, which demonstrates that they were not sure about all their answers. On multiple occasions participants claimed that it "didn't sound like you did anything" and requested that the task be executed again to assist in their decision making.

Finally the participants gave the sonification a rating of two on a scale of four on the questionnaire responding to a question directly asking them to rate the benefits of sonification in Crumbs. The sonification as implemented in Crumbs does not seem to be benificial.

5.3 Usability Issues

Assessment of quantitative and qualitative data resulted in the discovery of a number of usability issues concerning Crumbs. These issues were then investigated using Gabbard's *Framework of Usability Characteristics* to discover issues that violated one or more guidelines. Some issues also violated general HCI guidelines. Each issue is assigned a unique usability issue number that is used as a reference. For each we discuss specific VE and general usability guidelines it violates, and also discuss possible strategies for redesigning Crumbs for better usability.

Issue #1: Awareness of middle wand button for scale/mark mode

The middle wand button is a facilitator of multiple actions within Crumbs. The middle button's primary two functions are to scale the data volume and to manipulate crumb objects. To switch between button functionality modes, the user uses a menu item to issue the appropriate command. Although the menu system provides a cognitive affordance for the current mode, participants must constantly bring up the menu system and check the middle button sub-menu to ensure the current mode is the desired one. Although this cognitive affordance eliminated errors for participants, it greatly increased task completion times. One participant specifically addressed multi-functionality for the middle button saying, "the middle button has too many functions" and "Sometimes I inadvertently drop a crumb when I think I am going to scale". In reference to using modes in an application, Hix and Hartson state, "When it is used, the designer should be careful to distinguish different interaction modes for the user, so the user clearly knows at all times which mode is active". The issue also violates a guideline of the Gabbard framework. Which states that emphasis should be placed on information relating to user tasks.

Redesign Suggestion

This problem is similar to mode awareness issues in graphical editors. In this situation, the shape of the cursor indicated the mode of the editor [Hix93]. In a similar manner modifying the sword appearance to indicate current mode would supply the needed cognitive affordance.

Issue #2: Inconsistent use of direct manipulation

Crumbs provides eight separate types of objects: colormap, contrast enhancement object, clearbox object, sliceplane object, crumb object, trashcan object, opacity object, and the data volume box object. Individual instances can be created of clearbox objects, sliceplane objects, and crumb objects. For these three types of objects, Crumbs provides the trashcan object as a direct manipulation approach to deleting an individual instance of an object. However, for the remaining objects, removing them from the environment is either not supported or requires the non-direct method of using system commands. Although there is a distinct difference with deleting object instances and closing system objects, the strategy of providing a direct manipulation method for one and not the other caused errors, as noted in both quantitative and qualitative data. Providing contradictory methods of accomplishing similar tasks conflicts with a framework VE guideline which specifies, "The look and feel of command presentation, be it visual, aural, or haptic, should be consistent within a single interface". This issue is also addressed by Hix and Hartson, "Similar things are expected to be done in similar ways". This issue was categorized previously by the guideline-based evaluation.

Redesign Suggestion

It is a common opinion that VE applications should use direct manipulation techniques whenever feasible. Therefore, to remove the direct manipulation inconsistency, Crumbs needs to provide a direct manipulation method of exiting or closing objects. Simply using the same metaphor of dragging the object to the trashcan object is not sufficient since this action symbolizes *deleting* the object. This is different from *closing* the object. One possibility is to provide a separate object, maybe a toolbox object, that a user drags objects into if they want to close them.

Issue #3: Inconsistent method of selection

Crumbs uses a consistent selection technique for the eight objects introduced in issue #2. These objects are selected by placing the tip of the sword within the object's boundaries and clicking the left wand button. However, several of these objects contain sub-objects or widgets allowing a user to alter object settings by interacting with sub-objects. Selection of sub-objects is facilitated by a middle button click. On several occasions, participants attempted to use the left wand button to select sub-objects. Requiring separate methods to perform similar tasks also conflicts with the framework VE guideline that specifies, "The look and feel of command presentation, be it visual, aural, or haptic, should be consistent within a single interface". Hix and Hartson also guide designers that, "Similar things are expected to be done in similar ways." Although there is a difference between selection of an object in its entirety and selecting a sub-object, the difference is not distinct enough to warrant use of two separate selection buttons. This is also counter-intuitive due to its deviation from popular GUI interaction styles. GUIs facilitate selection as a "point and click" activity, using a mouse to position a cursor over the desired object and a mouse button to perform the selection. This same activity is used for selection of entire objects (i.e., windows, dialogs, etc.) or sub-objects (i.e., buttons, scroll bars, elevators, etc.). Crumbs attempts to leverage this experience substituting a wand for the mouse, a sword for the cursor, and a wand button for a mouse button. However, GUIs use the same mouse button to perform the selection of both objects and controls, whereas Crumbs uses two separate wand buttons. The first guideline in the framework states, "Take into account user experience".

Redesign Suggestion

"If something is done a certain way in an interface task, users expect the same thing to be done the same way throughout the rest of the interface" [Hix93]. Therefore, Crumbs needs to be redesigned. The selection method used to select sub-objects should mirror the technique used to select objects in their entirety.

Issue #4: Awareness of current position in crumb placement task

Participants experienced a common awareness deficiency when completing a task. They would lose their place when dropping crumbs in the sperm tail data set. The sperm tail marking task required continual manipulation of multiple objects to improve data visualization in order to facilitate crumb placement. While attempting to manipulate these objects to locate more of the sperm tail, all three participants that attempted this task lost the location of the last crumb they had placed. Participants made statements such as, "I lost my place" and "I seem to have lost my way". Because their attention was directed elsewhere, participants became unaware of the location of the current end of the sperm tail and therefore spent a substantial amount of time trying to relocate it. Continually keeping track of current location of the previously placed crumb object in a structure that requires a multitude of crumb objects in close proximity places a large memory load on users. This stresses users' mental aptitudes. A framework guideline warns, "Take into account users' technical aptitudes". Requiring users to keep track of crumb object positioning mentally also contrasts one of Nielsen's ten heuristics that stresses "recognition rather than recall".

Redesign Suggestion

Changing physical properties of the two crumb objects located at both ends of the fiber would allow users to rely on recognition rather than recall. Crumb objects could be enlarged or a different color could be used to act as a cognitive affordance.

Issue #5: Occlusion of 3D widgets

Crumbs provides a set of interaction objects, 3D widgets, to support altering data volume opacity and color settings. Because these interaction objects are 3D artifacts located in three-space, they can be hidden from view by the data volume object or other interaction objects. This inability to locate an interaction object was unsettling to participants and led to multiple errors and prolonged task completion times, especially in instances where the user specifically requests to activate the object with the desire to interact with it. If an interaction object did not appear the instance an *edit* command was initiated, most participants concluded that the *edit* command had failed and instinctively attempted to re-execute the command. A guideline in the framework's VE System and Application Information section instructs VE application designers to focus on "attention to visual organization of the display". It further specifies that "guidelines include minimizing overall and local density, and emphasizing information related to user tasks".

Redesign Suggestion

Interaction objects must remain visible to facilitate use. Requiring users to locate an interaction object they had just requested in three-space prior to interacting with it proved to be counter-intuitive and confusing. Although this usability issue only occurred when participants scaled the data volume large enough to occlude the interaction objects, the lack of awareness in this situation consistently led to errors and frustration. Addition of an audio cue as feedback to the *edit opacity* or *edit colormap* commands would allow users to be aware of command execution. Also, objects used to alter data volume attributes should be drawn in front of other environmental objects providing, access at all times.

Issue #6: Inappropriate Audio Annotations

Crumbs effectively uses audio annotations in some situations as task feedback. However, execution of two tasks resulted in audio feedback that did not facilitate usability. Selection of a crumb object responds with an *ouch* audio cue. Manipulation of the opacity object results in a *good luck* audio cue. Both of these messages conflict with the framework guideline that states, "System messages should be worded in a clear, constructive manner so as to encourage user engagement (as opposed to user alienation)". One participant, when commenting about the *good luck* annotation, said, "This is really annoying". Two of the five participants reported after the evaluation that audio cues were either unprofessional or that they "are not related to the actions that they are associated with". This is consistent with guidelines-based inspection results as well.

Redesign Suggestion

Provide both of these situations with audio feedback that more effectively reflects the action.

Issue #7: Arm movement facilitating cascading menus

Transitioning from a cascading menu item to its sub-menu requires movement of the sword tip. If, during the course of moving from parent menu item to sub-menu, the sword tip happens to locate outside the desired parent menu, the sub-menu disappears and the menu item corresponding to current sword tip location is highlighted. This behavior is consistent with GUI cascading menu interactions. Intuitively, users attempt to execute this motion using their arm instead of their wrist. The arm motion required for sword movement is far greater in the CAVE than on a desktop. Therefore natural arm movement is not a straight line, but rather an arcing motion. Due to this arcing motion, one participant was unable to execute a menu item located on a sub-menu on the first try. This issue is facilitated by the length of the menu item label. Width of the menu is adjusted dynamically to allow encapsulation of the menu command with the longest length. Menus with longer lengths require more arm movement. This is especially problematic for users with shorter than average arm lengths. This conflicts with the framework guideline that instructs, "accommodate natural, unforced interaction for users of varied age, gender, stature, and size" and "input devices should make use of user physical constraints and affordances".

Redesign Suggestion

One course of action is to design menu item labels are limited in size yet still clearly and concisely reflect meaning. Another redesign possibility is to alter the physical interaction facilitating cascading menus. Cascading menus interaction is different for some GUI interaction styles. Although most styles are similar to Crumbs design and interaction, some style require the user to select a parent menu item which results in the sub-menu remaining visible until the next button click. Although this does solve the arm movement issue, it could conflict too much with user's prior experience interacting with cascading menus.

Due to lack of space, we are unable to give full explanations of the other usability issues discovered during the formative evaluation. However, we do list the remaining issues below.

- Menu truncation due to sword location
- Different crumb modifications use similar interactions
- Awareness of colormap object box value
- Use of color box metaphor in colormap object
- Two stepped load colormap interaction
- Colormap object and Opacity object intuitiveness

5.4 Participant Suggestions

Following their evaluation seminar participants were given the chance to suggest improvements for Crumbs. Due to the open forum in which participant suggestions were recorded, individual suggestions varied between participants. Below is a list of the primary suggestions.

- Participants requested a method to allow Crumbs to automatically locate dense structures.
- Participant requested modified approach to object manipulation. Another participant addressed what they believed to be an inconsistency between the manipulation method used to control orientation of the data volume box and other objects in the system. They wanted the application to "allow the user to rotate an individual object like a clear box, because they are hard to orient properly". When the evaluator asked the participant to expand on the suggestion, they revealed that they had a hard time using the *object in hand* metaphor provided by the application to orient objects. The data volume box, on the other hand, separates the orientation control from the translation control. The participant felt this method allowed for more precise control.
- Participants wanted ability to control size of selection mechanism, i.e. alter the length of the sword.
- Participant felt that textual representations of interface objects were necessary. For example, if the user did not want to use the color cube to specify colors for certain densities, they should have the choice of using a more conventional textual based interface.
- Participant suggested providing an audio help feature.
- Participant suggested the inclusion of more tools to assist users with data visualization. One participant wanted multiple views of the data volume, "so one can see what is going on simultaneously from several angles". Other requests by the participant included 1)objects in the environment that are near the sword or at the end of the sword "to turn a different color so one can see when the end of the sword is approaching them", 2) add shadows into the environment "to help one see projections of the crumbs line" and 3) a dithering mode that "turns the volumetric parameters back and forth slowly automatically (with all tools) so that one can see when something is in the volume".

6. Discussion

This initial application of a formative evaluation driven by a set of VE specific guidelines and a previously executed guideline-based inspection was executed to assess both the usability of Crumbs and the strengths and weaknesses of our formative evaluation technique in VE assessment. The evaluation did successfully uncover usability issues, but did not fully result in measures allowing assessment of all aspects of the environment covered in the framework guidelines.

A key finding of our research is the framework of VE-specific usability guidelines successfully uncovered usability problems when applied in a guideline-based inspection. This guidelines-based inspection also successfully provided guidance in creating tasks to further assess usability issues categorized as possible usability problems. Using guidelines-based inspections to help drive formative evaluation task creation in VEs is a novel idea and more research is needed to support these findings.

Another key finding of conducting this evaluation was the difference between collecting pertinent data on specific physical tasks and participant's cognitive state. When evaluating the usability of an immersive VE, at least two separate issues are addressed: usability of the interaction style in supporting *VE Users* and *User Tasks* and the level of immersion the VE's *Virtual Model* provides. Conventional measures of task performance and subjective satisfaction can be effectively applied to assess the usability of interaction styles to evaluate the environment's locomotion, object selection, and object manipulation techniques. In addition, participant-supplied ratings, recorded in a questionnaire following the evaluation, of the separate categories of interaction styles are also effective on collecting quantitative data on interaction style usability.

Measuring immersion supported in the *Virtual Model* is a much more abstract concept than measuring the interaction styles. This evaluation attempted to collect some quantitative data on the *Virtual Model* by way of participant rating using a questionnaire. As reported in table 4, users ranked their sense of presence at 2.75 on a scale of one to four. The Virtual Model in the Crumbs Application depends on the specific dataset being analyzed. It is difficult for participants to believe they exist inside a fertilized fruit fly nucleus.

7. Conclusion

We have presented a two-phase usability engineering methodology and applied it to the Crumbs CAVE application. This exercise has produced two key findings with respect to the methodology framework, and uncovered many usability issues with the Crumbs application. We find it interesting that some of the usability issues in Crumbs is a direct result of the application being originally designed as a demonstration program for the trade show SigGraph 94. The authors believe that many VE applications start as demonstration programs which, like Crumbs, become scientific analysis tools. Perhaps some of the issues discovered during the evaluation of Crumbs can be applied to other such programs.

Acknowledgements

We would like to thank our colleagues at Virginia Tech and NCSA for their continued support of ongoing HCI and VE research. In particular members of the VT-CAVE office: Dr. Ronald Kriz, John Kelso, Valdis Kletnieks, Kevin Curry, and Joseph Vandyke for software and hardware support, NCSA CAVE office: Stuart Levy, Cami Tiska, George Estes, and Jeff Carpenter. I would also like to thank Joseph Gabbard for his continual support and insight regarding this research.

References

[Bowman98] Bowman, D. Interaction Techniques for Immersive Virtual Environments: Design, Evaluation, and Application. "Boaster" paper presented at the Human-Computer Interaction Consortium (HCIC) Conference, 1998.

[Brady95] Brady, R., Pixton, J., Baxter, G., Moran, P., Potter, C. S., Carragher, B., and Belmont, A. (1995). Crumbs: a virtual environment tracking tool for biological imaging. In IEEE Symposium on Frontiers in Biomedical Visualization Proceedings, pages 18-25.

[Brady96] Brady, R., Bargar, R., Choi, I., Reitzer, J., "Auditory Bread Crumbs for Navigating Volumetric Data", Proceedings of IEEE Visualization '96: Hot Topics. IEEE Computer Society Press, Los Alamitos, Calif., pp 25-27, 1996.

[Gabbard97] Gabbard, J. L. A taxonomy of usability characteristics in virtual environments. M.S. Thesis, Virginia Polytechnic Institute and State University. 1997. Document available online at http://www.vpst.org/jgabbard/ve/framework/.

[Gray98] Gray, W. D., & Salzman, M. C. (1998). Damaged merchandise? A review of experiments that compare usability evaluation methods. Human-Computer Interaction, 13, 203-261.

[Hix93] Hix, D. and Hartson H. R. (1993). Developing User Interfaces. John Wiley & Sons, Inc.

[Hix99] Hix, D., Swan, J. E., Gabbard, J. L., McGee, M., Dubrin, J., & King, T. User-centered design and evaluation of a real-time battlefield visualization virtual environment. In Proceedings of the Virtual Reality '99 Conference, Houston, Texas, March 13-17, p. 96-103. 1999.

[Karr97] Kaur, K. Designing virtual environments for usability. In Proceedings of the INTERACT '97 Conference on Human-Computer Interaction, Sydney, Australia, p. 636-639. Document available online at http://web.soi.city.ac.uk/homes/dj524/kully.html 1997.

[Mack94] Mack, R. L., Nielsen, J. Executive Summary. In Usability Inspection Methods, chapter 1, p. 1-23. John Wiley & Sons, Inc. 1994.

[Nemire94] Nemire, K. Building usable virtual environment products. CyberEdge Journal, 1994, Volume 4, Number 5, pp. 8-14.

[Nielsen94] Nielsen, J. Heuristic evaluation. In Usability Inspection Methods, chapter 2, pages 25-62. John Wiley & Sons, Inc. 1994.