Pain and Fatigue in Desktop VR: Initial Results

Christopher D Shaw,
Department of Computer Science,
University of Regina, 3737 Wascana Parkway,
Regina, Saskatchewan, S4S 0A2, CANADA
Phone (306) 585-4071 cdshaw@cs.uregina.ca

Abstract

This paper describes a comprehensive experimental evaluation of a two-handed free-form surface editor called THRED, which uses a pair of Polhemus 3D trackers with added buttons in a complementary two-handed style. On top of the underlying free-form surface editor application was built two other user interfaces that provide reasonable competition for the two-handed style. The second interface uses one button-enhanced 3D tracker in the dominant hand, with the non-dominant hand selecting commands from the keyboard. The third style is a mouse-based interface that is a simplified clone of the Alias modeler. This user study evaluates these interfaces in terms of pain and fatigue. The results show that experienced minimal pain and fatigue with THRED, an a par with that experienced in the mouse-based interface, but there was statistically significant fatigue in the use of the One-Handed interface. The pain and fatigue surveys clearly indicate that THRED and the Mouse-Based interface yield low discomfort, which contradicts the established wisdom that bat-based interfaces are likely to be painful or fatiguing to use.

KEYWORDS:

User Interface Software, Virtual Reality, Experimental Evaluation, Pain and Fatigue, Interactive 3D Graphics, Two Handed Interfaces.

1 Introduction

A limited amount of research has been reported on the general acceptability of 3D tracker-based user interfaces. The usual goal of a 3D user study has been to demonstrate the improved effectiveness of a core interaction technique in performing a particular task. However, a faster task completion time is insufficient grounds for adopting a new technology if this faster time comes at some expense in training, user comfort, or other forms of user resistance. A more appropriate technique is to additionally examine the other factors that will affect the success of new technology, such as the amount of time it takes for users to become proficient with the system, whether the system causes pain or fatigue while in use, and whether users prefer it to the alternatives already in place. The work reported here evaluates the pain and fatigue potential for a new Two-Handed 3D tracker-based editor called THRED (for Two-Handed Refining EDitor [12]).

THRED is a "fish tank" system in which the user sits in front of a graphics console (figure 1) that has a screen, keyboard, mouse, and two Polhemus magnetic 3D position and orientation trackers with three added buttons per tracker. It is a simple quadrilateral-based free-form surface editor in which users can select and move one or more vertices to create the desired surface. The polygonal surface is composed entirely of a rectangular mesh of vertices, with each vertex tessellated with a pair of triangles. THRED continually animates the 3D object being created, and the user holds a magnetic sensor (Bat (figure 2)) in each hand, pressing the bat buttons to manipulate the surface.

Figure 1: A typical user of THRED.

THRED is essentially a vehicle for the exploration of interaction techniques and so has a limited geometry repertoire. However, the system is complex enough that all 6 available bat buttons have a unique function, including a bat-controlled menu that controls editor modes.

1.1 Experimental Approach

The experimental approach was to have users perform timed trials on THRED and on other user interfaces that represent competitive styles. To avoid the comparing THRED against a "straw man" interface, the other interfaces should represent the state of the art. Two competitive interfaces were chosen: a mouse-based interface that is a simplified clone of the Alias modeler [1], and an interface that uses a single button-enhanced 3D tracker in a similar style to JDCAD, developed by Liang [7].

Figure 2: The button-enhanced bat.

The Mouse-Based interface was selected from among the best professional CAD systems commercially available. The intent of this experimental comparison was to compare the new technology against the toughest level of competition. However, to avoid swamping the subject with the complexity of Alias, subjects in the experiment were presented with only a small fraction of the real Alias tool.

The One-Handed style was chosen because it implements the logical alternative to two bats – a single bat manipulated by the dominant hand. The non-dominant hand selects keyboard keys to invoke commands, allowing for for a wide range of operations. The One-Handed style also has the advantage of requiring less 3D tracking hardware than THRED.

THRED users interact almost exclusively with the two bats, using the mouse and keyboard only to use a file selection box to load and store files. Because users interact with THRED with their hands not resting on the table, and because some muscular effort is required to maintain the hand's presence above the table, there is concern that extended use of 3D trackers may result in arm pain or fatigue. The recognition of Repetitive Strain Disorders has heightened general concern that new user interfaces may be injurious to the user's health. The experiment reported here attempts to quantify what areas of the body might be most at risk of discomfort, and to compare interface styles with respect to their discomfort level.

2 Previous Work

This section briefly reviews previous user studies of 3D tracker-based user interfaces in the "desktop VR" style. Ware and Jessome [18] showed that 3D objects can be more easily manipulated by a bat than with a mouse, because the user does not have to mentally break down the 3D task into a sequence of 2D operations. Subsequent work by Ware *et al.* [17] has investigated the effectiveness of head-coupled perspective and stereopsis for the understanding of complex 3D scenes, and the effects of lag and frame rate [16] in selection.

Zhai, Buxton and Milgram have experimentally evaluated 3D volumetric cursors using various semitransparent rendering schemes [19]. They found that a semitransparent volume cursor is quite effective at selecting a moving object.

In an experiment comparing bat and mouse effectiveness for 2 nominally 3D tasks, Jacob *et al.* found that the bat worked best for a task that had 3 spatial dimensions, while a mouse with mode change worked best for a task with 2 spatial dimensions and a color dimension [5].

Pausch *et al.* found that a head-tracked display was more effective for visual search than controlling the view with the eye-in-hand metaphor [9]. They also found some negative transfer effects between these two visual control modes.

In the Ergonomics field, repetitive strain injury, back pain and so on are topics of major interest [10]. The central message of this work is that proper seating posture and workspace configuration, coupled with frequent short work breaks can help avoid injury.

For 3D user interfaces, the ocular fatigue effects of stereoscopic displays have been studied extensively [8, 15]. One of the early papers on 3D tracker use reported user fatigue when a Polhemus sensor acted purely as an absolute locator [2].

A physiological and subjective study of fatigue in HMD-based VR by Igarashi et al. indicated no significant increase in fatigue over a 30 minute period [4].

Perhaps the most studied health phenomenon in VR user is *simulator sickness*, a syndrome caused by visually perceived motion being at variance with vestibular perception of body motion. Volume 1 number 3 of *Presence* is devoted to this topic [14].

Perhaps because of simulator sickness, the physical comfort of users is an issue for a variety of VR and highly-interactive 3D interfaces. For interfaces that use the bat, there is a concern that arm fatigue may set in after prolonged periods of use, canceling out the increased effectiveness brought about by using the bat.

3 The Experiment

THRED is constructed so that the interface elements are separated from the surface data structure and operations. With THRED, interface styles can be compared alone, because they are based on the same underlying operations, and are performing the same fundamental task.

Our experimental approach was to build three different user interfaces, and have experimental subjects perform the same simple task on each. Pain and fatigue ratings were collected at the start, middle and end of the trials.

3.1 Two-Handed Interface

The first interface uses a pair of bats with added buttons in a complementary two-handed style. Each hand holds a bat, and presses the attached buttons to invoke commands. Each hand plays a distinct role, with the dominant hand being responsible for picking and manipulation, and the less-dominant hand being responsible for context setting of various kinds. For the sake of rhetorical convenience, we will refer to the dominant hand as the right hand and the less-dominant hand as the left, but the system is ambidextrous, because the bats are symmetric and can be handled with equal ease by both hands. In the interest of brevity, some interface details will be left out.

In this interface, the left hand is used to grab the surface and move it around, to select commands from a menu, and to control constrained reshaping. To enter grab mode, button 3 (nearest the wire) on the left bat is pressed and released. To release the surface, button 3 is again pressed and released.

Menu selection is accomplished using a hierarchical *sundial* menu, which is a circular popup menu controlled by button 2 (the middle button). The desired menu item is picked by orienting the *shadow stick* about its base so that the stick's endpoint lies visually in front of the sector, and releasing the button.

The right hand is used to pick vertices, and to move and reshape these vertices either freely in 3D or along an axis line. Rigidly attached to the right cursor is a selection probe, represented by a narrow cylindrical shaft. The user controls the probe's position and orientation, pointing it at the desired vertex like a laser beam. The underlying selection mathematics is actually more like a flashlight, so that the user does not have to hit a vertex precisely.

Reshaping is accomplished by pressing right button 2 and dragging the right bat, then releasing the button. In line-constrained reshaping, the left hand orients the left cursor to be aligned to the desired axis, and the right hand executes the reshape while the system constrains motion along that axis. This axis is only evaluated at right button press, so the user can relax the left hand if necessary once reshaping has started.

3.2 One-Handed Interface

The one-handed interface is so named because there is only one cursor, controlled by a single bat (figure 3). All of the 3D geometric functionality of the Two-Handed interface moves to the single 3D tracker in the right hand. The three buttons on the right tracker maintain the functionality of selection and 3D reshaping. The left tracker buttons no longer exist, so the left hand uses the 1, 2, and 3 keys on the keyboard to activate the commands that were previously assigned to left buttons 1, 2 and 3. The 1 key toggles constraint mode, the 2 key pops up the sundial menu on the right cursor, and the 3 key toggles reorientation of the 3D scene on the right tracker. All geometric operations that were previously performed by the left bat are now performed by the right bat.

Figure 3: The One-Handed interface. The right hand holds the Bat, and the left hand manipulates the 1, 2, and 3 keys.

Constrained reshaping operations are combined onto the right cursor. That is, the user must first align the constraint line to the desired axis with the right bat, then press the right button 2 to commence constrained reshaping. Once reshaping starts, the right hand need not maintain the orientation alignment constraint.

3.3 Mouse-Based Interface

The Mouse-Based interface interface presents the standard three orthographic views and one perspective view, in which the mouse can perform spatial operations (figure 4). Below the 4 views is a menu bar that contains 16 pull-up menus similar to the main task bar in Windows 95.

Figure 4: The Mouse-Based interface.

This user interface is specially optimized to minimize the number of menu selections that the user must perform. More frequently used commands are placed in pull-up menu slots nearest the menu bar, so that they are easiest to hit. The parent items in the menu bar are quite large, and thus also easy to hit. A few heavily-used commands like vertex selection and vertex motion are modes, so that the user does not have to continually select menu items to continue these operations.

Each pull-up submenu has “sticky” items. The most recent submenu item is remembered (and displayed in a box below the parent item), so the user can just press and release the mouse button on the parent item to again activate this sticky submenu item. Infrequently-used commands in a submenu are not sticky.

This clone interface uses the same underlying geometry and rendering operations as the One- and Two-Handed interfaces. However, the Mouse-Based syntax is the same as that of the Alias modeler, but simplified to remove all features not present in THRED. Thus, constrained reshaping in 1 or 2 dimensions can be done by dragging the mouse in the appropriate orthographic view, but full 3 DOF reshaping is impossible. Vertex selection is done by moving the mouse near the desired vertex and pressing the mouse button. Scene navigation is selected in each window by pressing a mouse button on an icon in the window title and dragging the mouse to move the scene.

4 Tasks

For this experiment, subjects were asked to perform multiple trials of the following two tasks. Both tasks required the subject to create simple geometric shapes which could be easily described, recognized, created, and measured. An easily-understood target shape minimizes the subject’s cognitive burden, allowing greatest possible concentration on performing the task. The subject can quickly verify if the created object is correct, because it is easily understood. A simple shape also allows the subject to create it quickly, allowing many practice iterations.

4.1 Task 1

When the user selects the *New* command, the editor creates a single square quad 2 by 2 meters (top left figure 5). The task is to create a flat, rectangular, connected, 3 by 3 array of 1 meter by 1 meter squares. This is the first task that any user would perform on the surface before creating a more complex shape.

The progress of steps is shown in figure 5: Subdivide the surface in X and Y twice – this is yields the top middle image of figure 5. The uneven size of rectangles arises because the subdivision operator divides in half the leftmost column or bottom-most row of rectangles. Next, the subject selects the two left columns and drags them left; then selects the left column and drags it left; selects the bottom 2 rows and drags them down; then selects the bottom row and drags it down.

This task is best accomplished using a series of line constrained movements, so that the grid does not become crooked. Rulers and grid lines are used to measure the length and width of the rectangles. A 10 centimeter snap-grid is active at all times.

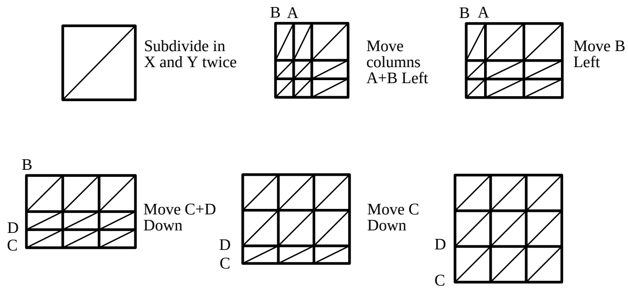

Figure 5: The 5 major steps of task 1.

4.2 Task 2

Task 2 continues where task 1 leaves off. The goal is to create a 1 by 1 by 1 meter open box using the 3 by 3 meter grid of squares.

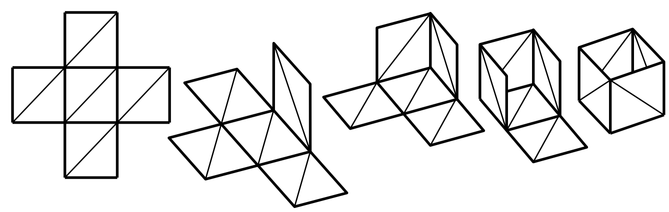

Figure 6: Task 2: Delete the 4 corner squares, then fold up each flap to be perpendicular to the center square.

Given the 3 by 3 grid, the box creation method is to cut off the corner squares and fold up the four flaps to be perpendicular to the central square. There is an alignment line that marks perpendicularity for the first flap.

Each folding step requires two axis-aligned moves. The first step moves the flap up 1 meter, and the next step moves the flap towards the center to make the flap perpendicular to the center square. Contour lines on the surface and ruler marks on the polygon edges are used by the subject to correctly adjust the height of the flaps. The 10 centimeter snap grid is active, so incorrect flap locations are visually obvious.

5 Experimental Structure

The experiment was a balanced, within-subjects design. Each subject performed the tasks using two of the three interfaces. There were three conditions, each consisting of one of the pairs of interfaces outlined above. Within each condition, subjects were randomly assigned to one of the two possible interface presentation orders. A total of four blocks of 5 trials were used: task 1 on interface

A, task 1 on interface B, task 2 on interface B, task 2 on interface A.

This design was chosen because a subject could complete the experiment more quickly, and because of possible differential transfer. The One-Handed and Two-Handed interfaces were rather similar, and an experimental design in which the subject used all 3 interfaces would make a training effect more difficult to analyze.

5.1 Subjects

There were a total of 18 subjects in this experiment, 16 males and 2 females, ranging in age from 19 to 28 years. Experimental volunteers were solicited from two second-year mechanical and civil engineering drafting/computer-aided design courses. These subjects were chosen because they had a reasonable knowledge of Computer-Aided Design, including a basic vocabulary of geometry, and therefore would readily understand the basic idea of how to perform the tasks. At the time of the experiment, all students had completed the term's course work in these classes. Volunteers were given $20 for their participation in the experiment.

5.2 Materials and Equipment

This experiment used a Silicon Graphics 150MHz Crimson RealityEngine with a 21 inch screen. The trackers were a pair of Polhemus Isotraks synchronized at an update frequency of 30Hz, and driven by other computers on the local ethernet. The subject sat in a comfortable, padded chair with padded armrests. The armrest height was approximately 1 inch below the height of the tabletop. The keyboard, mouse, and CRT were all mounted directly on this table without any height adjustments except for the mousepad. The subject sat directly in front of the console for all practices and trials. Subjects were encouraged to make themselves as comfortable as possible, and to adjust their workspace as much as they deemed necessary. All room lights were on for the experiment. There was no screen glare.

5.3 Experimental Steps

When the subject arrived in the experiment room, they provided demographic information, including a self-report of the amount of experience with using computers and designing geometric objects. Subjects also reported on the amount of pain and fatigue they were feeling prior to starting the training and trials (baseline measure). For each body part, the rating was made on a visual scale 10 centimeters wide, with a tick mark at each centimeter numbered from 0 to 10 (figure 7). This is a standard pain rating scale used widely by the medical community [3].

Subjects rated the 10 following body areas: 1. Eyes and face; 2. Neck; 3. Shoulders; 4. Back; 5. Left hand; 6. Left wrist; 7. Left arm; 8. Right hand; 9. Right wrist; 10. Right arm.

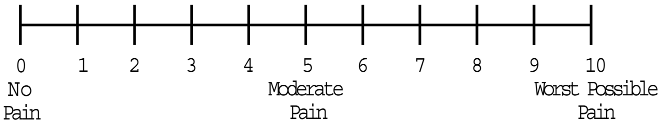

Figure 7: A pain rating scale. The body part was labeled to the left of the scale. This image is smaller than the actual scale used.

The trainer demonstrated the first of the two interfaces that they were to use. As a demonstrational vehicle, the trainer went through the steps required to perform task 1.

Subjects then sat at the console, made themselves comfortable, and performed a practice trial for the first task. During the practice trial, subjects received feedback and help to ensure the correct result. The practice trial was not timed.

Following the practice trial, the subjects were asked to complete 5 timed trials of the task with the following instruction in mind:

Go as fast as possible but make no errors.

The trainer elaborated on this by informing the subject that little blunders along the way were perfectly OK, but that the requirement was to produce a correct model at the end. Therefore, any errors induced in the model must be corrected before the trial is over.

Subjects were given a minimum of 30 seconds to rest between trials. Subjects received feedback on their performance after each trial. Next, subjects completed the same series of training, practice, and 5 timed trials using interface B. Upon completion of interface B trials, subjects were again asked to rate pain and fatigue. This is referred to as the *Midway Survey*.

Subjects next received training on task 2 using interface B, practiced task 2 once, and performed 5 trials. The same process of training, practice, and 5 trails then followed for interface A. Finally, pain and fatigue were assessed as before (the *End Survey*).

Condition Name →		MouseOne		MouseTwo		OneTwo	
# subjects →		3	3	3	3	3	3
Task 1	Int. A	Mouse	One	Mouse	Two	One	Two
	Int. B	One	Mouse	Two	Mouse	Two	One
Midway Survey		Srv	Srv	Srv	Srv	Srv	Srv
Task 2	Int. B	One	Mouse	Two	Mouse	Two	One
	Int. A	Mouse	One	Mouse	Two	One	Two
End Survey		Srv	Srv	Srv	Srv	Srv	Srv

Table 1: Ordering of trial blocks for the experiment. 18 subjects total.

5.3.1 Summary

Table 1 shows the ordering of trial blocks for the whole experiment. The right three pairs of columns show the three conditions of the experiment, with 6 subjects per condition. The conditions are labeled *MouseOne* or *M1* for the Mouse and One-Handed condition, *MouseTwo* or *M2* for Mouse and Two-Handed, and *OneTwo* or *12* for One- and Two-Handed. Each condition has one of two interface presentation orders, either ABBA or BAAB, with 3 subjects each. Each subject was randomly assigned to one of the columns of the table.

6 Task Times

Task completion times were recorded for each trial. At the end of the first block of 5 trials, the mean completion time for Task 1 was 85 seconds for the Mouse-Based interface; 67 seconds for the One-Handed interface; and 77 seconds for the Two-Handed interface. These results are presented simply to show that the task times are comparable.

7 Pain and Fatigue

A total of three pain and three fatigue ratings were collected per body part per subject over the course of the experiment. The first rating can used as a zero point for the two following ratings, thus controlling for pre-existing conditions. For example, one subject had dental braces installed the day before the experiment, and consequently rated eye/face pain at 7 throughout the experiment.

A total of 1080 raw pain or fatigue ratings were collected during this experiment. Dividing by the number of subjects per condition, there are 180 raw pain or fatigue means:

$$10\,Body\ Parts\ \times\ 3\,Conditions\ \times$$
$$2(Pain\ or\ Fatigue)\ \times\ 3\,Surveys$$

However, these raw numbers hide pre-existing conditions. Instead, the *normalized scores* will be reported, which are calculated by subtracting the initial rating from any subsequent rating of the same item by the same subject. This cuts the total number of means to report to 120.

The use of normalized scores means that negative scores are possible, due to per-subject variability in second and third raw ratings after a nonzero initial rating.

7.1 General Pain and Fatigue Results

There are a number of general results of the pain and fatigue survey.

1. There is no significant increase in *pain* in any body part at any time for any condition.
2. Midway pain and fatigue is usually higher than End pain and fatigue, indicating a habituation response.
3. The Left Hand shows a statistically significant increase in fatigue at the end of the trials for the OneTwo case.
4. When statistically significant differences occur *between* interfaces, the increasing order of interfaces pairs is MouseTwo, MouseOne, OneTwo. At the Midway point, these differences occur in Left Wrist, Shoulders, and Back fatigue. At the End, Left Hand, Left Arm, and Shoulders fatigue are statistically significant between conditions.

7.2 Pain and Fatigue Analysis

Most body areas under most conditions yielded mean increases in pain or fatigue less than 1. The general hypothesis is that average pain and fatigue level will not increase beyond 1. The deviations from this will be reported here.

In this section, statistical tests of normality often fail, indicating that the F-test and T-test are invalid. The alternative to the T-Test is the nonparametric Mann-Whitney U Test, which tests for differences between means by ranking all of the data, and using the distribution of the ranks into the two classes to determine a statistic. This test has a relative power of 95.5% compared to the T-Test, which means it will discriminate 95.5% of the means that the T-Test would find to be different [13].

The question of interest is whether subjects under a particular condition experience more pain and fatigue that other conditions. Thus, under the *Tests* columns of the following tables, probabilities of the appropriate between-subjects test are reported. This will indicate if there is a statistically significant increase in fatigue for one condition over another. In the following results, where the statistical tests of normality fail, the nonparametric Mann-Whitney U Test will be reported. Otherwise, test results ending in T indicate a T-Test.

Table 2 shows the scores for body parts which showed a statistically significant difference in fatigue between conditions. The top half are the Midway survey results, and the bottom half are from the End survey. The left 3 columns with the heading *Mean Fatigue* show the mean normalized fatigue scores for the 3 conditions *M1*, *M2* and *12*. Each mean contains 6 subjects. The right two columns headed *Between-Subjects Tests* shows the results of comparing the two means indicated by the subheading. For example, the *M2-12* column contains the probability that *M2* mean is different from the *12* mean.

For the shoulders, the MouseTwo score was significantly less than the MouseOne and OneTwo scores at the Midway and End.

For the Back, the Midway OneTwo score yielded a significant difference (p=0.045) from only the MouseTwo score. Other pairwise tests indicated no significant difference.

The comparison between the MouseOne and OneTwo conditions did not yield a significant difference.

Midway	Mean Fatigue			Between-Subjects Tests	
Region	M1	M2	12	M1-M2	M2-12
Shoulders	.42	-.04	.70	p=.032	p=.032
Back	.29	-.25	.37	p=.340	p=.045
Left Hand	.58	.00	1.65	p=.592	p=.070T
Left Wrist	.85	-.13	1.04	p=.093	p=.045
Left Arm	.10	-.13	1.57	p=.391	p=.092
End					
Shoulders	.57	-.06	1.11	p=.032	p=.021
Left Hand	.25	.00	2.34	p=.017	p=.005T
Left Wrist	.56	.00	1.07	p=.531	p=.128
Left Arm	-.16	-.32	1.44	p=.107	p=.031

Table 2: Fatigue results by condition.

For the left hand in the OneTwo case, the Midway and End scores are 1.65 and 2.34 respectively, with T-Tests for the null hypothesis yielding (p=0.069) and (p=0.010), respectively. The End score is a significant result, indicating that there is indeed a fatigue increase for the left hand in the OneTwo condition.

However, the other conditions for the left hand are not different from zero by a statistically significant margin. The MouseOne-OneTwo tests all yield ($p>0.184$), and are not shown.

From these summary tables, it is clear that the least left arm reported fatigue arises from the MouseTwo case, while the most reported fatigue arises from the OneTwo case, with the MouseOne case in the middle.

7.3 Alternate Analysis

The results reported above take a conservative approach to dealing with data that apparently deviates from the normal distribution. Some authors note that the F-test is somewhat robust in the face of non-normality [6]. Table 3 shows an analysis of variance on the pain and fatigue data indicated above.

Midway	Mean Fatigue			F-Test
Region	M1	M2	12	F(2,15)
Shoulders	.42	-.04	.70	p=.173
Back	.29	-.25	.37	p=.153
Left Hand	.58	.00	1.65	p=.138
Left Wrist	.85	-.13	1.04	p=.295
Left Arm	.10	-.13	1.57	p=.111
End				
Shoulders	.57	-.06	1.11	p=.090
Left Hand	.25	.00	2.34	p=.001
Left Wrist	.56	.00	1.07	p=.355
Left Arm	-.16	-.32	1.44	p=.038

Table 3: Fatigue results by condition.

These results indicate that the there is not a statistically significant difference between the reported fatigue means at the Midway survey. At the End survey, the Left Hand and the Left Arm once again have different means by a statistically significant margin. Bonferroni Least Significant Difference Post-Hoc tests on these two variables indicate that the OneTwo condition yields a fatigue increase for the Left Hand and the Left Arm at the 0.05 level.

7.4 Pain and Fatigue Discussion

The surprise of this survey is that there is a difference in overall left arm fatigue, especially considering that the MouseTwo case consistently has at or near the lowest scores. While using the Mouse-Based interface, the subject's left hand is idle, so the subject can rest it if necessary. Consequently, fatigue scores for conditions that include the Mouse-Based interface are the lowest, because the left arm is idle for half of the experiment.

In the Two-Handed interface, the subject manipulates the left bat. Fatigue scores in the MouseTwo condition are the lowest, while the highest scores occur in the OneTwo condition.

In the One-Handed interface, the subject selects keyboard commands with the left hand from the 1, 2 and 3 keys. Fatigue scores for conditions that include the One-Handed interface are the highest.

Taken together, this indicates that the higher fatigue scores for the left arm in the OneTwo case mostly arise from the use of the One-Handed interface. This in turn indicates that the use of the keyboard as a substitute for buttons on the bat is probably a bad idea. In a review of various types of computer-related workplace injury [11], Sellers notes that the leading cause of Cumulative Trauma Disorders (CTD) such as Carpal Tunnel Syndrome is static posture. He reports that the accepted recommendation for avoiding such problems is to frequently change postures. The One-Handed interface does not easily offer this opportunity, because the left hand is poised over the 1, 2 and 3 keys for the duration of each trial.

To add evidence to this view, the Shoulder fatigue scores are also highest when the One-Handed interface is used, and is lowest for the MouseTwo case. In fact, comparing between conditions indicates that the MouseTwo Shoulders scores are significantly lower from the other two, and that the MouseOne and OneTwo scores do not differ from each other. Sellers mentions that CTD symptoms are not confined to the hands, but can be displaced into the shoulders and neck area. This is consistent with idea that left arm fatigue arises from the static posture required by the One-Handed interface.

8 Conclusions

This is the first survey of pain or fatigue effects related to the use of 6 DOF tracking systems outside the study of simulation sickness. The importance of this is simply that one of the commonplace objections to the extended use of 6 DOF trackers is that arm pain or fatigue might set in. Because users interact with bat-based systems with their hands not resting on the table, and because some muscular effort is required to maintain the hand's presence above the table, this is a sensible concern.

The results of this survey indicate that THRED presented minimal difficulties in the realm of user pain and fatigue, while the One-Handed interface induced statistically significant fatigue. In order of increasing pain and fatigue, the subjects in the MouseTwo condition experienced the least, followed by MouseOne subjects. The OneTwo subjects had the most pain and fatigue, with statistically significant fatigue increases over the course of the experiment. This leads to the conclusion that the Two-Handed interface has as small an impact on user pain and fatigue as the Mouse-Based interface. The importance of this result is simply that commonplace fears about the use of 6 DOF trackers for a non-trivial period of time are unfounded if the Two-Handed interface style is used. In short term usage at least, the Two-Handed interface style is as painless as the Mouse-Based style.

Of course, subjects performed this experiment over the course of 2 hours (including training, task completion and form-filling), so the longer-term pain and fatigue effects of this technology are not addressed by this work. This is an area for future research.

Increasing spatial input bandwidth cannot be done without regard to the ergonomics involved. The pain and fatigue surveys clearly indicate that THRED and the Mouse-Based interface yield low discomfort, which contradicts the established wisdom that bat-based interfaces are likely to be painful or fatiguing to use. This evaluation indicates that THRED is as ergonomically sound as Mouse-Based interfaces in the short term.

References

1. Alias Research Inc. *Alias/3 User's Manual*. Toronto, 1993.
2. Norman I. Badler, Kamran H. Manoochehri, and David Baraff. Multi-dimensional input techniques and articulated figure positioning by multiple constraints. In Frank Crow and Stephen M Pizer, editors, *Proc. 1986 ACM Workshop on Interactive 3D Graphics*, pages 151–170, Chapel Hill, NC, October 1986.
3. RH Gracely and PJ Wolskee. Semantic Functional Measurement of Pain: Integrating Perception and Language. *Pain*, 15(4):389–398, 1993.
4. H Igarashi, J. Noritake, N. Furuta, K. Shindo, K. Yamazaki, K. Okamoto, A. Yoshida, and T. Yamaguchi. Is the Virtual Reality a Gentle Technology for Humans? An Experimental Study of the Safety Features of a Virtual Reality System. *IEICE Transactions on Information and Systems*, E77-D(12):1379–1384, December 1994.
5. Robert J. K. Jacob, Linda E. Sibert, Daniel C. McFarlane, and Jr. M. Preston Mullen. Integrality and Separability of Input Devices. *ACM Trans. Computer-Human Interaction*, 1(1):3–26, March 1994.
6. Roger E. Kirk. *Experimental Design: Procedures for the Behavioral Sciences*. Brooks/Cole., 3 edition, 1995.
7. Jiandong Liang and Mark Green. JDCAD: A Highly Interactive 3D Modeling System. *Computers and Graphics*, 18(4):499–506, 1994.
8. T Miyashita and T Uchida. Cause of Fatigue and its Improvement in Stereoscopic Displays. *Proceedings of the S.I.D.*, 31(3):249–254, 1990.
9. Randy Pausch, Dennis Proffitt, and George Williams. Quantifying Immersion in Virtual Reality. In *SIGGRAPH 97 Conference Proceedings*, pages 1–8, August 1997. Los Angeles, California.
10. G. A. Ryan. Effects of an Episode of Intensive Keying on Data Process Operators. *International Journal of Human-Computer Interaction*, 4(2):221–226, April-June 1992.
11. Don Sellers. *ZAP! How your computer can hurt you – and what you can do about it*. Peachpit Press, Berkeley, California, 1994.
12. Chris Shaw and Mark Green. THRED: A Two-Handed Design System. *Multimedia Systems*, 5(2):126–139, March 1997.
13. Sidney Siegel and John Castellan Jr. *Nonparametric statistics for the behavioral sciences*. McGraw-Hill, New York, 2nd edition edition, 1988.
14. Simulator Sickness Special Issue. *PRESENCE: Teleoperators and Virtual Environments*, 1(3):295–363, 1992.
15. D. A. Southard. Viewing Model for Virtual Environment Displays. *Journal of Electronic Imaging*, 4(4):413–420, October 1995.
16. Colin Ware and Ravin Balakrishnan. Reaching for Objects in VR Displays: Lag and Frame Rate. *ACM Transactions on Computer-Human Interaction*, 1(4):331–356, December 1994.
17. Colin Ware and Glenn Franck. Evaluating Stereo and Motion Cues for Visualizing Information Nets in Three Dimensions. *ACM Transactions on Graphics*, 15(2):121–140, April 1996.
18. Colin Ware and Danny R. Jessome. Using the Bat: A Six Dimensional Mouse for Object Placement. In *Proceedings of Graphics Interface '88*, pages 119–124, Edmonton, Alberta, June 6-10, 1988.
19. Shumin Zhai, William Buxton, and Paul Milgram. The "Silk Cursor": Investigating Transparency for 3D Target Acquisition. In *Proceedings of ACM CHI'94*, pages 459–464. ACM SIGCHI, 1994.