A Study of Haptic Linear and Pie Menus in a 3D Fish Tank VR Environment

Rick Komerska and Colin Ware
Data Visualization Research Lab, Center for Coastal & Ocean Mapping (CCOM)
University of New Hampshire
{komerska | colinw}@ccom.unh.edu

Abstract

New pop-up menu styles such as pie menus and marking menus have proven to be measurably faster and more accurate in mouse and pen-based interfaces. These characteristics suggest that they may also be useful for 3D haptically enhanced environments. This paper reports on our development and evaluation of a pie menu style and two types of linear menu styles. All utilize force to aid the user in option selection and activation. Our evaluation results show that selection using the pie menu is considerably faster and more accurate than both types of linear menu. Selection using push-through or exceed border methods was found to be superior to conventional button selection. We conclude that pop-up pie menus, with the right combination of selection method and assistive forces, can provide an excellent solution to providing menu choices in 3D haptic environments and that considering speed accuracy tradeoffs is important in making design decisions.

1. Introduction

3D virtual computer graphics environments have long suffered from the drawback that a sense of touch is usually lacking. Although convincing visual imagery can be created, the users hand passes right through it violating any sense of natural interaction. The advent of devices such as the PHANToM (SensAble Technologies, Inc.) is changing this. It is now possible to construct small, high quality virtual environments allowing for objects that can be both seen and touched, albeit within a small working volume. This kind of environment can be used for applications such as molecular modeling [1], "sculpting" virtual objects [2] and exploration of geoscience data [3].

Despite the fact that VR can make some tasks simpler and more natural, it has still proven necessary to add conventional elements from tradition mouse-based WIMP (windows, icons, mouse and pull-down menus) to create usable systems. In particular, menu interactions have been found necessary to access a full range of functionality [4, 5]. However, touchable menus are a relatively new possibility and, although a few systems have added force feedback to enhance menu interactions, there has been little systematic study into the design of force assisted menus for interaction in 3D virtual workspaces. In the present paper we present such a study.

In approaching the problem of menu design for visual and haptic virtual environments, we decided early on to consider both pie menus and more conventional linear menus. Pie Menus [6] utilize a circular design where options are arrayed in segments around the perimeter. The menu appears centered at the cursor and to make a selection the user makes a movement into one of the radial segments, then selects by releasing the mouse button. Research has shown this menu style to be faster and less error prone than linear menus due the large selection regions and short, consistent movement needed to make a selection [7]. An additional advantage of pie menus is that users can learn the angle of movement required to make a selection and this can become encoded in so-called "muscle memory". Over time, the expert user can simply make a movement, with appropriate mouse button actions, without needing to look at the menu options. This kind of transition to rapid expert use is not available with linear menus. Marking menu techniques [8] embody this gestural behavior, and allow a series of nested selections to be accelerated into a zigzag gesture.

FlowMenus also employ a pop-up pie menu design with a pen interface and combine command and direct manipulation [9]. They are similar to marking menus but activate on return-to-center instead of pen-up. Each of these techniques has demonstrated improvement in selection time and accuracy over the traditional linear menu interface for many common menu tasks. However, a key disadvantage with pie menus is that only a limited number of menu options (typically eight or less) can be comfortably shown on a single menu.

A number of recent studies have shown that force feedback can provide improvement in the menu selection process, at least for linear menus. Raymaekers and Coninx [10] carried out a study where subjects made selections from a linear menu with seven options, using a PHANToM with a pen interface. They compared point and click with push-through interaction for option selection, both with and without a planar force co-registered with the visual menu panel. They found the point and click metaphor to be fastest and least error prone, and haptic feedback to be useful only in reducing the error rate.

Miller and Zeleznik [11] explored the utility of haptically enabling various parts of the X windows desktop, including menus, in an early qualitative study. They observed that force assistance appeared to be "promising". Using the haptic FEELit mouse, by Immersion Corp., Sjöström [12] implemented a haptic radial (pie) menu and received positive feedback from blind subjects who used it in a usability study.

Oakley et al carried out a series of more detailed quantitative studies exploring the value of haptically enabling various parts of the desktop [13-15]. In [15] they evaluated the utility of gravity (snapping to the center of targets), recess (border forces keeping the cursor within target bounds), friction and texture. The task was to select single isolated targets. Gravity provided the greatest benefit in reducing errors, while texture actually increased error rates. Oakley et al [14] examined the task of linear menu selection. For this task they concluded that applying simple assistive forces was generally not helpful. However they also developed an adaptive force algorithm that depends on the speed and direction of actions. Forces were reduced in the direction of travel and increased in the orthogonal direction. This reduced error rates for menu selections. Finally, Oakley et al [13] looked further at selection of a single target within a haptically enabled multiple target domain. They considered both structured (menu toolbar) and unstructured (computer desktop environment) target sets. Again they found the use of an adaptive force profile necessary to yield task completion times and error rates comparable to or better than the purely visual case. In addition, the adaptive force provided more benefit when selecting among closely spaced targets in the menu toolbar than among targets in the more sparsely populated desktop environment.

Pie menus and their variants have not been widely adopted in standard desktop WIMP interfaces. However, since haptic environments are relatively new, without established standards, this gives us the possibility of making use of the new designs should they be demonstrably superior. As part of a project to create a 3D haptically enhanced environment for GIS applications, our goal in the present study has been to investigate menu design and assess which combination of menu type, selection techniques and haptic assistive forces yield the greatest (and least) benefits, in terms of selection speed, error rate and user preference.

2. Menu Design

We developed haptically enhanced variants on linear menus as well as pie menus. We also created a new slanted linear menu design. Examples of each are shown in Figures 1 and 2. We also investigated a variety of force

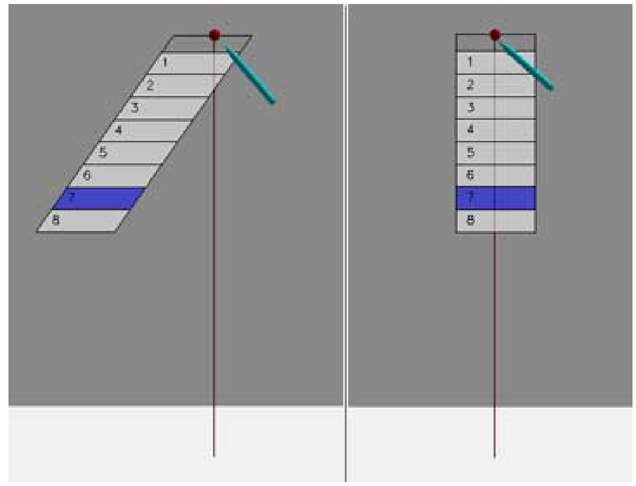

Figure 1. Slant and straight linear menus

enhancements. These include adding a force plane to keep the stylus on the menu, forces to keep the stylus within selection categories as well as two force-enhanced selection methods. One involves pushing the stylus down through the menu; the other involves moving the stylus laterally out of the menu (keeping in the menu plane). In both, selection is made when the user breaks through a force barrier. These are compared to conventional

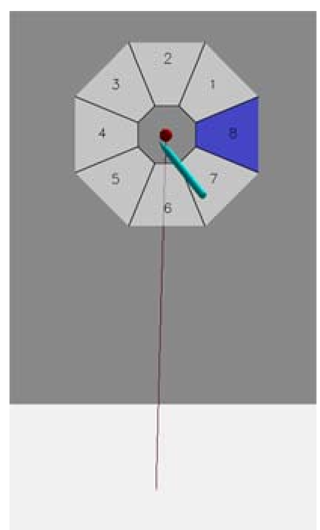

Figure 2. Pie menu

selection using the button on the PHANToM stylus.

All of the menu designs we considered were oriented to present a face-on view to the user in the 3D environment. They pop up with the pen tip initially at the top in the case of the linear menus and at the center in the case of the pie menu. They are sized to contain legible text and in fact are used as working menus in several other applications under our development.

The design for the straight linear menu is based on the standard pull-down menu. The overall menu dimensions are 14 mm x 35 mm, with the neutral region at the top having a height of 3 mm and each valid menu item having a height of 4 mm.

We developed the slanted linear menu based on our observation that users tended to rotate the pen about their wrist in making menu selections. We arrived at a slant angle of 35°, based on trial and error adjustments to make something that felt comfortable. The menu had a slant length of 61 mm. The height and average width of each menu item was the same as for the straight linear menu.

The pie menu is based on previous guidelines for designing pie menus [6]. We utilize a standard wedge size that subtends a 45° angle, with eight wedges aligned along the ordinal compass points. The menu layout has inner and outer radii of 5 mm and 16 mm, respectively.

2.1 Menu Selection Techniques

Three menu option selection techniques were investigated in our study:

- Release button
- Push-through
- Exceed border

In the release button method, the user moves the PHANToM pen from the initial menu starting location into the menu option of interest and releases the stylus button. This method is most similar to that used in most mouse-driven desktop systems.

Haptics allow for other selection techniques not involving the mouse button. Therefore, in addition to button selection, we decided to investigate push-through and exceed border techniques. The push-through method works in combination with the planar assistive force, described in more detail in the following section. In this technique, the subject moves the pen tip into the option and pushes into the plane of the menu. The selection occurs when the plane normal force on the pen exceeds 0.8 N. A variation on this technique, which we did not explore, is the virtual button with detent. We did not consider this method as we believed it would be considerably slower than the other two methods.

Finally, we examined an "exceed border" selection method, which relies on the subject to move the pen tip beyond the perimeter of the menu option of interest. Marking menus, as well as the results of Accot and Zhai [16] showing performance advantages for "goal-crossing" over traditional point and click task interaction, inspires our use of this technique.

2.2 Menu Option Assistive Forces

Three assistive force components were studied; 2D planar constraints, edge boundary constraints and option selection "snap" force. The force profiles used, especially that of the snap force, were selected by the authors based on trial and error to produce a comfortable interface.

2D planar constraints. A 2D planar force profile, having stiffness of 0.6 N/mm, constrains the pen tip to an infinite plane that is coincident with the menu plane. Most implementations of 3D menus have implemented a virtual plane to support movement in the planar surface of the menu, and we wished to determine if this were actually beneficial.

Edge boundary constraints. An edge boundary constraint force with a stiffness of 0.7 N/mm prevents the user from moving the PHANToM tip outside of the menu perimeter. It seems intuitive that constraining motion within the menu walls might improve performance. This region includes the neutral region where the pen starts as well as the eight menu option areas. Note that this menu border force extends infinitely in a perpendicular direction from both sides of the menu plane.

Option selection "snap" force. The option selection snap force acts to pull the pen tip to the center of the menu option entered. Although Oakley et al showed that this force did not speed menu selection in their task, we believe that such a force should provide perceptual benefits such as helping the user differentiate between menu options as well as decrease selection errors. We modeled this as a spring force having a constant of 0.5 N/mm, which is created between the current PHANToM location and the snap force center point, and is initiated when the PHANToM tip enters the menu option space. For the linear menus, the center point is the geometric center of the option entered. For the pie menu, this point lies at a radius of 8 mm along the radial ray bisecting the option entered. The snap forces are capped at 0.15 N for linear menus and when traversing among pie menu options. The force cap is increased to 0.5 N for the pie menu when entering from the neutral center area.

3. Task

For our evaluation, the task chosen for the subject was to move the PHANToM pen to select the target, then activate a pop-up menu. Once activated, the subject then had to choose the highlighted menu option as quickly and accurately as possible. This task was repeated across menu geometries, selection methods and assistive force combinations. Note that the focus of this task was on menu option selection performance *from the point at which the menu was activated*; the initial movement of the pen to the target was included only to provide a sense of realistic interaction within a hypothetical 3D virtual environment.

A proxy representing the PHANToM pen is shown in the center of the workspace. The target is shown as a red ball that floats over a white floor, connected by a line that provides depth cue information. To select the target, the user moves the PHANToM pen until the tip falls within 4 mm of the ball radius. At this point, the pen tip is subjected to a spring force with a constant of 0.3 N/mm that snaps the tip to the target center.

To activate the menu, the user depresses the PHANToM button and one of three menus appears as shown in Figures 1 and 2. The menu is created at the proper orientation to appear perpendicular to the subject's line of sight; for our fish tank VR setup (shown in Figure 3), this implies a plane oriented 45° to the desktop. Each menu is composed of eight options, one of which is highlighted blue. The user moves the pen in the menu plane into the blue menu option and makes the appropriate selection action. Visual feedback on the user's current option location is given by highlighting that option red. When an option is selected, the menu and target disappear. Audio feedback in the form of a chime or buzzer indicates whether the subject is successful in selecting the correct option. The red ball target then reappears at another random location in the workspace and the subject repeats the task. Targets were confined to a nominal volume of 108 mm wide by 63 mm high by 29 mm deep to allow all parts of all menus to be reached by the PHANToM.

Each menu type, selection method and set of assistive forces constitutes a condition. For each condition, all of the eight menu options are randomly tested, plus one extra that is thrown out at the start. After all conditions in a block have been tested, the entire block is repeated. Conditions are randomized across subjects and blocks.

3.1 Metrics

The principal metrics were menu option selection time and selection errors. Subjective measures were collected from a post-experiment questionnaire, asking the subject to rank the menu/selection options and comment on the assistive forces experienced during the experiment. Selection time was calculated from when the menu was activated until a selection was made. Supporting data collected included subject name, gender, menu type, selection option, assistive forces, target position, highlighted option and selected option.

3.2 Test Apparatus

The study was conducted using the Data Visualization Lab's Haptic-GeoZui3D VR environment [17]. In Haptic-GeoZui3D, the visualization and haptic components are unified using a fish tank VR arrangement shown schematically in Figure 3. A horizontal mirror is used to superimpose virtual computer graphics imagery onto the PHANToM 1.0 workspace. The placement of the mirror also means that the PHANToM and the user's hand are hidden from view. However, a proxy for the pen that the user holds is shown and, because the user's actual eye position is used to compute the CG imagery, visual and haptic imagery are co-registered at all times. To accomplish this, we use a 17-inch monitor set at a 45° angle above the mirror. Stereoscopic display is provided using NuVision Technologies stereo glasses with a monitor refresh rate of 100 Hz.

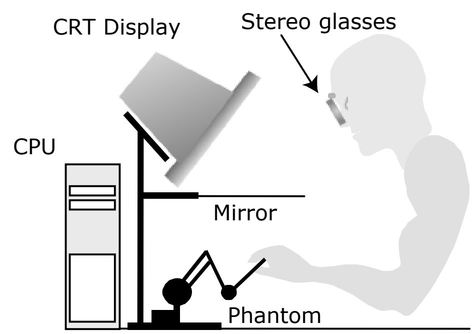

Figure 3. Fish Tank VR setup

4. Experiment 1: Linear and Pie Menus

The goal of the first experiment was to compare our three menu types using three different sets of assistive forces and two selection methods; release button and push-through.

Table 1. Experiment 1 test conditions

Menu Type / Selection Method			Assistive Forces			Number of Conditions
			Planar	Snap	Border	
	Straight	PT	on	on/off	on/off	4
		RB	on/off	on/off	on/off	8
	Slant	PT	on	on/off	on/off	4
		RB	on/off	on/off	on/off	8
	Pie	PT	on	on/off	on/off	4
		RB	on/off	on/off	on/off	8
					Total	36

Note: The "on/off" variable refers to assistive forces.

RB=Release button, PT=Push-through

4.1 Conditions and Trials

Table 1 summarizes the 36 conditions. Overall there were 8 trials per condition. The 36 conditions were presented in a different random order to each subject and the entire set was replicated twice (giving 16 observations per condition). This produced 576 trials per subject.

The subject was first briefed on the VR environment, menu types and selection techniques. Concerning the assistive forces, the subject was told only to expect some additional "guiding" forces but was not provided any further details. The subject then performed a set of 6 practice conditions, one for each menu type/selection method pair. This data was not recorded. During the practice trials, the suggestion was made to the subjects to hold their wrist (and the pen) turned slightly inwards, especially for selection with the slanted linear menu. The subject was able to take a short break between blocks, if desired. A questionnaire was administered at the completion of the experiment.

There were 10 male and 6 female subjects. Only right-handed subjects were used, as the slant angle direction for the linear menu was optimized for this hand. Most had no previous experience using the PHANToM device. A few had tried it for a brief demonstration.

4.2 Results

Because the design was incomplete (all the selection techniques except for push-through were run with and without planar forces), we performed two separate ANOVAs. The first ANOVA included all cases where the planar force was enabled. We found menu option selection time to be significant with regard to selection method (**F**(1,15) = 49.168, **p** < 0.0001) and menu type (**F**(2,30) = 113.128, **p** < 0.0001). The interaction between menu type and subject (**F**(30,30) = 7.491, **p** < 0.0001), as well as subject and selection method (**F**(15,30) = 7.172, **p** < 0.0001), was also found to be significant. No significant interaction was found between menu type and selection method. We also found no significant effect due to the snap or border assistive forces.

Table 2 shows the mean selection times for the menu types and selection methods. These results show the pie menu to be 25% faster than the next best method, the straight linear menu. The push-through technique also fared better than the release button method, with an approximately 12% speed increase across all menu types. The pie menu benefited the most from the push-through method, with a 15% increase in speed over the release button method.

Error rates were also lower for pie menus. Table 3 displays the mean error rate for each of the menu types and selection methods. The error rate for the slant menu is slightly less than the straight linear menu. For the selection method, the release button technique is 26-67% better than push-through across the menu types.

Table 2. Mean selection times (ms)

		Selection Method		
		Push-through	Release button	Mean *menu*
Menu Type	Straight	855	973	914
	Slant	900	987	943
	Pie	634	742	688
	Mean *selection*	796	900	

Note: Planar force enabled conditions.

Table 3. Mean error rate

		Selection Method		
		Push-through	Release button	Mean *menu*
Menu Type	Straight	6.3	4.0	5.2
	Slant	5.8	4.3	5.0
	Pie	3.3	1.1	2.2
	Mean *selection*	5.1	3.1	

Note: RB=Release button, PT=Push-through.
Planar force enabled conditions.

To examine the effect of having a haptic supportive plane underlying the menus, a second ANOVA was carried out for the release button method only but over all assistive force combinations and menu types. Here we found the 2D planar force to be a significant factor (**F**(1,15) = 10.912, **p** < 0.005). An examination of the mean times found that with the plane force off, selection time averaged 0.926 seconds. Turning this force on reduced the mean times to 0.900 seconds, yielding a minimal savings of 2.8%.

The subjective ranking data showed that 14 out of the 16 subjects rated the pie menus as best.

All subjects said they noticed the snap force, whereas approximately 38% and 69% noticed the border and planar forces, respectively. Subjects liked the snap force as it provided a better sense of positioning and control. The force level used was generally preferred also.

4.3 Discussion

It is evident that the haptic pie menu provides a much faster selection with lower error rates when compared to the haptic linear menus. On average they were 226 ms faster and had half the error rate.

Our finding that the release button technique yields fewer errors suggests a tradeoff between speed and error rate, regardless of the menu type.

In general, both menu type and selection technique were found to be much more important than the application of assistive forces in minimizing selection time. One exception was that the straight and slanted linear menus both benefited slightly from the addition of the planar force, when using the release button technique. For these menu types, the user is required to traverse a much greater distance to make selections. The planar force serves to minimize this travel distance by eliminating the off-plane movement component that adds to traversal time. We believe the pie menu did not benefit significantly from the planar force using the release button technique because of the short traversal distance required for option selection.

The role of the assistive forces, especially the option snap force, appears to provide more of a perceptual benefit. As mentioned previously, subjects liked the feeling of control they perceived when this snap force was applied even though the data does not show a significant effect on time or error rate.

5. Experiment 2: Pie Menus

Having established the superiority of pie menus, we made further refinements and tested them in an effort to realize further selection and accuracy gains. Two major changes were made in this experiment. First, we added the exceed border selection method to our set of selection techniques. This marking style interaction should be well suited for our pie menu selection task. Second, we refined the assistive snap force in an attempt to gain better performance from it. These refinements can be summarized as follows:

- Decreased the force capture radius to 4 mm
- Increased the maximum initial wedge snap force from 0.5 N to 0.6 N
- Deactivated the snap force when the pen tip crossed the 8 mm force center radius

Regarding the last bullet, the idea here was to support the exceed border selection method by leveraging the snap force into the wedge while not slowing down the final activation stroke through the wedge border. Note that for the exceed border selection and border "on" condition, selection occurs when the pen tip force against the border exceeds 0.5 N.

Table 4. Experiment 2 test conditions

		Assistive Forces		
		Snap	Border	Number of Conditions
Selection Method	PT	on/off	on/off	4
	RB	on/off	on/off	4
	EB	on/off	on/off	4
			Total	12

Note: The "on/off" variable refers to assistive forces. PT=Push-through, RB=Release button, EB=Exceed border

5.1 Conditions and Trials

In experiment 2, we evaluated three selection methods: push-through, release button, and exceed border. We also evaluated response times with snap and border forces turned on and off. The planar force was enabled for all conditions. Table 4 summarizes the conditions. A total of 12 conditions comprised each test block. This produced 192 trials per subject given 8 trials per condition and 2 test blocks.

The main experiment was presented in a similar fashion as before, including initial practice trials. A short questionnaire regarding user preferences was administered at the end. For this experiment, 7 female and 9 male subjects participated, including six subjects from the first experiment.

5.2 Results

By ANOVA, selection times were found to be significant with regard to selection method ($\mathbf{F}(2,30) = 8.114$, $\mathbf{p} < 0.002$) and in the interaction of subject and selection factors ($\mathbf{F}(30,19.24) = 5.667$, $\mathbf{p} < 0.0001$). The push-through force selection was 7% faster than button selection while exceed border selection was 17% faster. The snap force was significant ($\mathbf{F}(1,15) = 14.016$, $\mathbf{p} < 0.002$), while the border force approached significance ($\mathbf{F}(1,15) = 4.315$, $\mathbf{p} < 0.055$). There was a significant interaction though between the snap and border force factors ($\mathbf{F}(1,15) = 7.245$, $\mathbf{p} < 0.02$).

From Table 5, we observe that the border force helped in the push-through and release button methods, reducing selection times by 3.8% and 2.4%, respectively. Conversely, application of the snap force served to worsen selection times for all techniques. The effect on the push-through method was negligible, though this force increased release button and exceed border selection times by 4.4% and 5.5%, respectively. Applying both snap and border forces increased the selection times for the push-through, release button and exceed border methods by 6.2%, 8% and 22%, respectively, over the planar force only cases.

Table 6 shows the error rate percentage generated within each selection method, in combination with the snap and border assistive forces. Again, we see the trend where the application of the snap force hinders accuracy while the border force helps in this regard. With the snap and border forces combined, we see an increase in errors in the release button and, more pronounced, in the push-through methods. Conversely, we see a large drop in errors for the exceed border technique.

Table 5. Mean selection time (ms)

		Selection Method		
		Push-through	Release button	Exceed border
Forces	Planar only	661	700	560
	+ Snap	666	731	591
	+ Border	636	683	602
	+ Snap + Border	702	756	683
	Mean *selection*	666	718	598

Table 6. Mean error rate

		Selection Method		
		Push-through	Release button	Exceed border
Forces	Planar only	3.9	2.0	2.3
	+ Snap	3.9	3.5	2.7
	+ Border	2.3	2.3	2.0
	+ Snap + Border	7.0	3.5	0.4
	Mean *selection*	4.3	2.8	1.9

Subjects also ranked their overall preference for each selection technique in the following order: (1) release button, (2) exceed border, (3) push-through option. There was no overwhelming preference shown for any particular method though. Comments made afterwards revealed some common themes. For the push-through technique, subjects generally liked the haptic sensation of pushing into the menu to make a selection. There was a perception too that this was the slowest of the three techniques. Subjects perceived the release button method to provide the most control, and many thought it was the fastest and easiest technique to use. Finally, many subjects believed the exceed border method to be the most natural technique, although they felt themselves to be more prone to committing errors using this method over the other two methods.

5.3 Discussion

Both push-through and exceed border selection methods were faster than button selection. Push-through selection requires force assistance but the exceed border technique can work with or without border force constraints. The addition of a combined snap and border force in the exceed border selection method reduced the error rate dramatically.

Our attempts to improve performance with assistive forces yielded mixed results. The snap force by itself did not affect selection time or accuracy in the push-through method, but adversely affected both in the other two methods. Evidently, our attempt to improve performance in this manner failed. Application of the border force by itself did reduce selection time in the push-through and release button cases. This was likely due to this force keeping the pen in the selection space thereby saving the subject the extra return-to-option time that would have resulted from an overshoot of the pie perimeter. As expected, the exceed border method suffered using the border force, as the user had to overcome it to make the selection. We found the combination effect of snap and border forces difficult to understand, especially for the exceed border technique. Here we observed a large increase in selection time along with a large drop in error rate. Conversely, we observed a modest increase in errors for the push-through technique.

It is interesting to compare subject's perceptions against the observed data. Although users perceived the push-through method to be slowest, release button was in fact the slowest technique. Users were correct in their belief that release button offers good control in relation to reduced number of errors, but they were incorrect in believing that the exceed border technique increased errors.

6. Conclusion

Pie menus, used with the exceed border method and planar assistive force, appears to be the fastest selection technique of all the methods we tested. Adding a haptic border and snap constraint somewhat decreases the speed but increases the accuracy considerably. Therefore, the value of this assistive force in an implementation would depend on the relative value of speed and accuracy in selection.

Our results show that pie menus are substantially faster (by 25%) and more accurate than linear menus. This correlates with the 15% decrease in speed found in the mouse-based results of Callahan, et al [7]. Adding a simple push-through force interaction method reduced the selection time another 12-15% across both the pie and linear menus, although this was accompanied by an increase in error rate. Our results showing faster selection time using push-through rather than release button when interacting with the straight menu stand in contrast to those found by Raymaekers and Coninx [10]. We believe this is due to the fact that our menu option movements take place in the plane of the menu; the user does not first approach the menu orthogonally, with the risk (in the case of the push-through method) of colliding with the menu and accidentally making an incorrect option selection. We also employed stereovision for enhanced depth perception, which appears to be absent from their study.

In any practical implementation of pop-up pie menus, the eight item limit on selection choices is likely to be a handicap. When more than eight items are needed, the menus must be nested. Of course this problem is not exclusive to pie menus; pop-up linear menus also become unwieldy when the number of items becomes large. In any case, it is clear that some method must be found for effectively nesting pie menus in our environment and this may require us to re-visit some of the issues. For example, it is not clear if the exceed border or the push-through method will be best suited to menu nesting. Only implementation and additional testing will resolve this issue.

7. Acknowledgments

The authors gratefully acknowledge the support of NSF Grant 0081292 and NOAA. We would also like to thank Matthew Plumlee and Roland Arsenault of the Data Visualization Lab for their support, Hannah Sussman for helping oversee the experiments, and CCOM researchers and students who participated in the study.

8. References

[1] R. M. Taylor II, W. Robinett, V. L. Chi, J. Frederick P. Brooks, W. V. Wright, R. S. Williams, and E. J. Snyder, "The Nanomanipulator: A Virtual-reality Interface for a Scanning Tunneling Microscope," *Proceedings SIGGRAPH 93*, 1993, pp. 127-134.

[2] H. Chen and H. Sun, "Real-time Haptic Sculpting in Virtual Volume Space," *Proceedings ACM Symposium on Virtual Reality Software and Technology*, Hong Kong, China, 2002, pp. 81-88.

[3] N. M. Zamel, J. A. Pita, and A. H. Dogru, "Next-Generation Visualization Technologies for Exploration and Production," *Proceedings SPE Middle East Oil Show*, Bahrain, 2001.

[4] R. H. Jacoby and S. R. Ellis, "Using Virtual Menus in a Virtual Environment," *Proceedings SPIE*, 1992, pp. 39-48.

[5] D. A. Bowman and C. A. Wingrave, "Design and Evaluation of Menu Systems for Immersive Virtual Environments," *Proceedings Virtual Reality 2001 Conference (VR'01)*, Yokohama, Japan, 2001.

[6] D. Hopkins, "The Design and Implementation of Pie Menus," in *Dr. Dobb's Journal*, vol. 16, 1991, pp. 16-26.

[7] J. Callahan, D. Hopkins, M. Weiser, and B. Shneiderman, "An Empirical Comparison of Pie vs. Linear Menus," *Proceedings CHI `88 Conference on Human Factors in Computing Systems*, Washington, D.C., 1988, pp. 95-100.

[8] G. Kurtenbach and W. Buxton, "User Learning and Performance with Marking Menus," *Proceedings ACM CHI'94 Conference on Human Factors in Computing Systems*, Boston, MA, 1994, pp. 258-264.

[9] F. Guimbretière, A. Martin, and T. Winograd, "Measuring FlowMenu Performance," Stanford University Stanford CS Technical Report CS-TR-2001-02, September 2001.

[10] C. Raymaekers and K. Coninx, "Menu Interactions in a Desktop Haptic Environment," *Proceedings Eurohaptics 2001*, Birmingham, UK, 2001, pp. 49-53.

[11] T. Miller and R. Zeleznik, "An Insidious Haptic Invasion: Adding Force Feedback to the X Desktop," *Proceedings ACM Symposium on User Interface Software and Technology*, San Francisco, CA, 1998, pp. 59-64.

[12] C. Sjöström, "Designing Haptic Computer Interfaces for Blind People," *Proceedings Sixth IEEE International Symposium on Signal Processing and its Applications*, Kuala Lumpur, Malaysia, 2001.

[13] I. Oakley, A. Adams, S. Brewster, and P. Gray, "Guidelines for the Design of Haptic Widgets," *Proceedings 16th British HCI Group Annual Conference*, London, 2002.

[14] I. Oakley, S. Brewster, and P. Gray, "Solving Multi-Target Haptic Problems in Menu Interaction," *Proceedings CHI 2001 Conference on Human Factors in Computing Systems*, Seattle, Washington, 2001, pp. 357-358.

[15] I. Oakley, M. R. McGee, S. Brewster, and P. Gray, "Putting the Feel in 'Look and Feel'," *Proceedings CHI 2000 Conference on Human Factors and Computing Systems*, The Hague, The Netherlands, 2000, pp. 415-422.

[16] J. Accot and S. Zhai, "More than dotting the i's --- Foundations for crossing-based interfaces," *Proceedings ACM CHI'2002 Conference on Human Factors in Computing Systems*, Minneapolis, MN, 2002, pp. 73-80.

[17] R. Komerska and C. Ware, "Haptic Task Constraints for 3D Interaction," *Proceedings 2003 IEEE Virtual Reality - Haptics Symposium*, Los Angeles, CA, 2003, pp. 270-277.