Speech-Based Cursor Control

Azfar S. Karimullah
Interactive Systems Research Center
Information Systems Department
UMBC
1000 Hilltop Circle
Baltimore, MD USA
(410)-455-3888
akarim2@umbc.edu

Andrew Sears
Interactive Systems Research Center
Information Systems Department
UMBC
1000 Hilltop Circle
Baltimore, MD USA
(410)-455-3883
asears@umbc.edu

ABSTRACT

Speech recognition can be a powerful tool for individuals with physical disabilities that hinder their ability to use traditional input devices. State-of-the-art speech recognition systems typically provide mechanisms for both data entry and cursor control, but the researchers continue to investigate methods of improving these interactions. Numerous researchers are investigating methods to improve the underlying technologies that make speech recognition possible and others focus on understanding the difficulties users experience using dictation-oriented applications, but few researchers have investigated the issues involved in speech-based cursor control. In this article, we describe a study that investigates the efficacy of two variations of a standard speech-based cursor control mechanism. One employs the standard mouse cursor while the second provides a predictive cursor designed to help users compensate for the delays often associated with speech recognition. As expected, larger targets and shorter distances resulted in shorter target selection times while larger targets also resulted in fewer errors. Although there were no differences between the standard and predictive cursors, a relationship between the delays associated with spoken input, the speed at which the cursor moves, and the minimum size for targets that can be reliably selected emerged that can guide the application of similar speech-based cursor control mechanisms as well as future research.

Keywords

Speech recognition, navigation, mouse cursor, predictive cursor

Assets 2002, July 8-10, 2002 Edinburgh, Scotland

INTRODUCTION

Speech recognition-based dictation systems can allow users to generate various textual documents including email, memos, and papers by speaking to the computer. As a result, such systems can be valuable tools for individuals with physical disabilities that hinder their ability to use traditional input devices such as the keyboard and mouse. Similarly, speech recognition can prove useful when either the environment or the tasks in which the user is engaged interfere with the use of more traditional input mechanisms.

To effectively control all of the functionality of computers that employ graphical user interfaces using speech recognition, users must be able to enter data and manipulate the cursor as if they were using a keyboard and mouse. While state-of-the-art speech recognition systems support both of these activities, the resulting interactions are often slower than those that are possible with a keyboard and mouse. An effective speech-based cursor control mechanism, that provides the ability to position the cursor anywhere on the screen and initiate clicking, double-clicking, or dragging actions, would be a valuable tool for any individual who needs to interact with a computer, but cannot use traditional pointing devices.

SPEECH-BASED CURSOR CONTROL

Two fundamental approaches have been employed for speech-based cursor control: target- and direction-based navigation. For target-based navigation, the user identifies the desired destination (i.e., the target) and issues an appropriate command. As long as the user knows the name of the desired target, they can navigate directly to a word, an icon, a menu, or a region of the screen. Unfortunately, target-based navigation becomes more error prone as the number of possible targets increases. Users may find it more difficult to remember the names of the targets, multiple targets may have the same name (e.g., three instances of the word "Friday" may be visible on the screen at one time), and recognition errors become more common as the vocabulary increases. Examples of target-based navigation include:

- "Select Friday" which highlights the word "Friday" in a document, and
- "Move mouse upper left" which places the cursor in the middle of the upper left region of the screen.

Direction-based navigation results in discrete or continuous movement. For discrete direction-based navigation, the user specifies the direction and distance (e.g., in inches, centimeters, letters, words): "Move left three words" would cause the cursor to jump three words to the left. For a continuous direction-based navigation, the user specifies the direction: "Move Left" causes the cursor to start moving slowly to the left until the "Stop" command is issued.

Our goal is to provide speech-based cursor control that will facilitate complete control over all applications. Target-based navigation can be effective when every possible target has a name, when users can remember all of these names, and the system can reliably distinguish the names. It is unlikely that standard target-based navigation will prove effective when providing users with complete control over a large range of applications. While screen regions could be used as targets, standard high-resolution displays combined with the 3x3 grid that is typically used to subdivide the screen, suggest that this approach is unlikely to be effective. Direction-based navigation using discrete movements could be used, but recent studies suggested that users experience a great deal of difficulty using discrete direction-based navigation in the context of a dictation-oriented application where movements were based upon characters, words, or lines of text [13]. While discrete direction-based navigation using standard units for distance (e.g., inches or centimeters) should be explored, this is not the focus of the current study. Instead, we focus on continuous direction-based navigation.

RELATED RESEARCH

The process of correcting recognition errors has received significant attention in the past decade. Several recent studies confirmed that correcting these errors accounts for a significant portion of the users' time [7, 9, 13]. More importantly, one recent study [13] concluded that users spent one-third of their time simply navigating from one location to another within the documents they were creating. While researchers have examined the optimal string length for digit input [1], alternative error-correction strategies for use in a phone dialing scenario [2], and various feedback mechanisms [3, 12], none of these studies focused on the issues involved in navigating through the documents that were created.

More recently, Suhm, Myers, and Waibel discussed multimodal techniques for correcting recognition errors that made use of the keyboard, mouse, a stylus, and speech [14]. They found that multimodal techniques resulted in correcting errors more quickly, but all of their techniques used touchscreen-based navigation which allowed users to simply touch the words they wanted to modify. Similarly, Danis et al. developed a speech-oriented editor where users could use a "Point and Speak" method to change the insertion point while dictating, but cursor movements were accomplished using a mouse [4].

McNair and Waibel also explored error correction, but focused more explicitly on speech-based selection of incorrect words [11]. Their early version of target-based navigation used the recognized word as well as the list of alternatives for each word when determining the target of the navigation command. They reported a failure rate of 15%. More importantly, as discussed earlier, target-based navigation is unlikely to prove effective for the tasks that are the focus of the current study.

Manaris and Harkreader explored the use of speech recognition as an alternative mechanism for generating keystrokes and mouse events with the goal of developing an alternative data entry technique for individuals with upper-body motor-control impairment [10]. Navigation was accomplished using direction-based navigation commands which generated continuous cursor movements (e.g., "Move left" followed by "Stop") as well as target-based commands which moved the cursor to one of five predefined regions of the screen (e.g., discrete movements). A pilot study was conducted using a Wizard-of-Oz simulation of their SUITEKeys software, but no results were reported for the navigation mechanisms included in this system.

Christian, Kules, Shneiderman, and Youssef explored speech-based navigation in the context of the web [4]. Navigation was accomplished by saying the words that served as a link to additional material or by speaking a number automatically associated with the link by the browser. This target-based navigation resulted in minimal errors, but significantly longer task completion times when compared to mouse-based navigation.

More recently, de Mauro, Gori, Maggini, and Martinelli discussed the design of a voice-controlled mouse [6]. Their system generates continuous movements in response to simple utterances. Unlike most implementations where users speak complete commands (i.e., "Move left"), individual vowels are mapped to commands. As a result, users must learn the appropriate mapping from a command (i.e., "Move left") to the utterance that causes that command to be executed (i.e., "A"). Their system also supports target-based navigation within regions of a window. As a result, once the cursor is located over the title bar of a window, several commands appropriate for that particular context become available including: minimize, close, and move. At present, navigation within textual documents is only supported using continuous mouse movements and no data is provided regarding the efficacy of their navigation mechanisms.

While speech-based navigation has been shown to be problematic, relatively few studies have explored the issues involved speech-based cursor control. Even fewer have compared alternatives, reported empirical results, or provided guidelines for the effective use of speech-based cursor control. The current article focuses on the issues that effect direction-based commands which generate continuous cursor movements.

A PREDICTIVE CURSOR

Three delays can be associated with spoken input. First, the individual must initiate a verbal response to a visual signal. Second, the individual must speak the required command. Third, the system must recognize the words that were spoken. When discussing the delays associated with spoken input, we are referring to the sum of these three delays.

Given the delay associated with spoken commands, a moving cursor will not stop immediately when the "Stop" command is issued. This is complicated by the fact that each component of this delay may vary between users. Various solutions could be implemented to address these delays:

1) The system could provide no assistance, requiring users to estimate where the cursor would stop if the "Stop" command were issued at any given time. This approach is currently used in some commercial products.
2) The system could compensate for the additional distance the cursor traveled while the "Stop" command was being processed. The speed at which the cursor moves is known. By determining the average delay, the average distance the cursor travels between when the user issues the "Stop" command and when it is recognized by the system can also be determined. When the "Stop" command is recognized, the cursor could jump backwards an appropriate distance to compensate for the delay. Since the cursor is always moved beyond the target (and then jumps backwards), this approach would increase the time required to select a target by an amount equal to the delay.
3) The system could help the user estimate where the cursor would be located if they issued the "Stop" command at any given time. By determining the average distance the cursor moves during the delay, a "predictive cursor" could be positioned an appropriate distance in front of the actual cursor to indicate where the cursor is likely to stop if the "Stop" command were issued. This approach does not increase the time required to select a target.

The first solution is already employed in some commercial products and the second solution artificially increases selection times. Therefore, we chose to compare the first and third solutions in the current study. We implemented the predictive cursor is as follows:

- Our system supports six basic navigation commands: Move left, Move right, Move up, Move down, Stop, and Click.
- The cursor moves at a rate of 20 pixels per second (i.e., approximately 53mm/second).
- Since the delay will vary between individuals, and each individual may take a different amount of time to issue the "Stop" (or "Click") command once the cursor reaches a target, the offset used for the predictive cursor is calibrated for each individual user. Calibration is accomplished by moving the regular cursor toward a small target and having the user issue the "Stop" command when the cursor reaches the target. This is repeated four times (once each moving up, down, left, and right) with the average distance between where the cursor actually stopped and the center of the cursor being used as the predictive cursor offset. The average offset for our participants was approximately 25 pixels (i.e., 66mm or a delay of 1.25seconds with the cursor moving at 20 pixels/sec).
- When the cursor is not moving, the predictive cursor is not visible (i.e., only the regular cursor appears).
- When the user issues any "Move" command, the predictive cursor appears next to the actual cursor, offset by the appropriate distance in the direction of the movement. For example, if the "Move right" command were issued, the predictive cursor would be offset an appropriate distance to the right of the actual cursor (see Figure 1).
- When either the "Stop" or "Click" command is recognized, the predictive cursor is hidden.

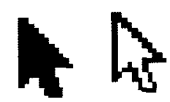

Figure 1: Illustration of the predictive cursor when the "Move Right" command is issued. The arrow on the right indicates the location where the cursor is expected to stop if the "Stop" command is issued at any given time.

THE EXPERIMENT

We report the results of a study that investigates several factors that may affect the efficacy of continuous direction-based navigation using a between-group experimental design. The purpose of this study was to investigate the benefits of providing a predictive cursor as well as the influence of target size, distance, and direction on the time required to position the cursor, positioning errors, user confidence as measured by the use of the "Stop" command, and user satisfaction ratings. The hypotheses for this experiment were:

- H1: The predictive cursor will have a significant effect on the time required to position the cursor when the target size is sufficiently small. For larger targets, the predictive cursor will not provide any benefit.

- H2: The target size will have a significant effect on the time required to position the cursor, the error rate, and user confidence.
- H3: The distance will have a significant effect the time required to position the cursor. The relationship between distance and time will be linear when targets are sufficiently large. When targets become smaller, distance will become less dominant and the linear relationship between distance and time will fade.
- H4: Diagonal movements must be accomplished using both horizontal and vertical movements. Since diagonal movements require two positioning activities (one horizontal and one vertical), diagonal movements will result in more errors than straight movements. When the target is sufficiently large, the time required for a diagonal movement will be dominated by the distance the cursor moves. When the target is sufficiently small, the time required for a diagonal movement will be dominated by the size of the target.
- H5: Satisfaction ratings (speed, accuracy, and comfort) will favor the predictive cursor.

METHOD

Participants

Twenty-eight individuals were recruited to participant in this study. Participants were required to speak English as a native language and could not have previous experience using speech recognition systems. Participants could not have any uncorrected visual impairments or documented hearing speech or cognitive impairments. Fourteen participants were male. The average age of the participants was 21.8 (stdev=5.5). Fourteen participants selected targets using the predictive cursor while the remaining fourteen used the standard cursor.

Equipment

A PC running Windows NT was used for this study. Spoken commands were processed using the IBM ViaVoice (Millennium Edition) speech recognition engine. Participants used a headset mounted microphone when interacting with the software. A custom application, developed using Visual Basic, presented targets and automatically recorded times, errors, and all other user activity. This application was configured to present either a standard cursor (i.e., a single arrow) or the predictive cursor (see Figure 1) depending on the experimental condition the participant was assigned to.

Procedure

A between-group design was employed such that each participant used either the standard or predictive cursor. Participants in the predictive cursor condition were guided through the calibration process to allow the system to customize the predictive cursor offset for each individual. As mentioned above, the average offset was approximately 25 pixels.

Participants selected a total of 72 targets (eight locations for each of three target sizes and three distances). Targets were grouped into blocks by size and each block was subdivided by distance. For each distance, targets were located in eight locations (see Figure 2). The order that the three blocks were completed was randomized. Within each block, the sub-blocks were also randomized. Finally, the order in which the eight locations were experienced was randomized within sub-blocks. Participants were allowed to take breaks as necessary.

When ready to begin a task, participants issued the "Ready" command. At this time, the cursor was positioned in the middle of the screen, the target appeared in the appropriate location, and an internal timer started. Participants issued appropriate commands to move the cursor over the target and then issued the "Click" command. The "Stop" command can, but does not have to, precede the "Click" command. If the cursor was not inside the target when the "Click" command is issued, an error was recorded and the participant had to reposition the cursor and issue another "Click" command.

Independent and Dependent Variables

The independent variables were: the type of cursor (i.e., standard or predictive cursor), target size, distance, and direction. The type of cursor was treated as a between-group variable with half of the participants completing the study with the standard cursor and the other half with the predictive cursor. Target size was treated as a within-subject variable with three levels. The targets were squares measuring 32mm, 64mm, and 128mm per side (referred to as S1, S2, S3). Movement distance was also treated as a within-subject variable with three levels. The center of the target was placed 1.9cm, 3.8cm, and 7.6cm from the starting cursor location to the center of the target (referred to as D1, D2, D3). Finally, movement direction was also treated as a within-subject variable with two levels. While eight locations were employed, we focus on the difference between those locations that can be reached using a single vertical or horizontal movement (i.e., straight movements, see Figure 2 positions 1, 3, 5, and 7) and those that require both horizontal and vertical movements (diagonal movements, see Figure 2 positions 2, 4, 6, and 8).

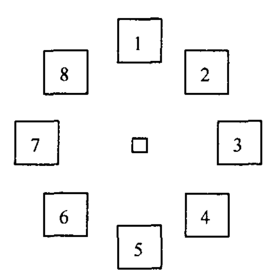

Figure 2: Targets were positioned in eight directions relative to the original cursor location.

Dependent variables included the time required to position the cursor over the target and issue the "Click" command, positioning accuracy, user confidence, and user satisfaction. The time between the "Ready" command and the "Click" command was automatically recorded for each target. Positioning accuracy was assessed by counting the number of "Click" commands issued when the cursor was not over the target. User confidence was assessed by counting the number of "Stop" commands issued. When sufficiently confident, users will issue the "Click" command without verifying that the cursor is in the correct location by issuing the "Stop" command. As users become less confident, they will use the "Stop" command to verify that the cursor is positioned correctly before issuing the "Click" command. Finally, numerous "Stop" commands indicates that the user experienced difficulty positioning the cursor. Satisfaction was assessed using three questions that asked about the participants perceptions regarding the speed and accuracy of the technique and how comfortable they were using the technique.

RESULTS

Positioning accuracy

Means and standard deviations for the number of errors made when selecting targets using the standard and predictive cursors are reported Tables 1 and 2 respectively. An ANOVA with repeated measures for direction, distance, and target size was utilized to assess the effect of cursor type. Cursor type did not have a significant effect on errors ($F(1,26)=0.41$, n.s.). Size and direction both had a significant effect on errors while distance did not ($F(2,52)=29.28$, $p<0.001$; $F(1,26)=12.99$, $p<0.002$; $F(2,52)=1.96$, n.s. respectively). Two significant interactions were identified: direction/cursor type and size/direction/cursor type. As expected, smaller targets and diagonal movements both resulted in more errors, supporting H2 and H4.

Selection time

Means and standard deviations for the time required to select the targets using the standard and predictive cursors are reported Tables 3 and 4 respectively. A logarithmic transformation was applied to these data prior to analysis to address the non-normal distribution of the data. An ANOVA with repeated measures for direction, distance, and target size was utilized to assess the effect of cursor type. Cursor type did not have a significant effect on the time required to select the target ($F(1,26)=0.11$, n.s.). Direction, distance, and size all had a significant effect on the time required to select the target ($F(1,26)=144.06$, $p<0.001$; $F(2,52)=648.57$, $p<0.001$; $F(2,52)=146.26$, $p<0.001$ respectively). Three significant interactions were identified: direction/size, distance/size, and direction/distance/size. As expected, smaller targets, longer distances, and diagonal movements resulted in longer selection times. These results support H2, H3, and H4. H1 is not supported.

	Straight			Diagonal		
	D1	D2	D3	D1	D2	D3
S1	0.55 (0.63)	0.80 (1.02)	0.86 (0.82)	0.70 (0.87)	0.55 (0.49)	0.80 (0.55)
S2	0.16 (0.19)	0.20 (0.20)	0.14 (0.16)	0.27 (0.29)	0.34 (0.46)	0.27 (0.27)
S3	0.02 (0.07)	0.04 (0.09)	0.09 (0.16)	0.07 (0.15)	0.04 (0.13)	0.09 (0.21)

Table 1: Mean error rates for tasks completed using the standard cursor (standard deviations in parentheses).

	Straight			Diagonal		
	D1	D2	D3	D1	D2	D3
S1	0.43 (0.85)	0.50 (0.45)	0.52 (0.45)	0.79 (0.88)	0.86 (0.63)	0.98 (0.72)
S2	0.29 (0.37)	0.30 (0.31)	0.21 (0.22)	0.43 (0.49)	0.48 (0.54)	0.50 (0.54)
S3	0.09 (0.21)	0.16 (0.23)	0.13 (0.24)	0.05 (0.11)	0.09 (0.16)	0.14 (0.23)

Table 2: Mean error rates for tasks completed using the predictive cursor (standard deviations in parentheses).

	Straight			Diagonal		
	D1	D2	D3	D1	D2	D3
S1	42.2 (36.0)	60.3 (54.8)	69.8 (37.1)	35.7 (24.4)	39.9 (26.4)	60.2 (29.1)
S2	14.1 (5.0)	20.5 (8.4)	26.9 (8.6)	15.9 (6.1)	25.2 (14.6)	43.1 (31.0)
S3	7.4 (2.4)	11.5 (3.3)	21.5 (6.0)	12.2 (7.1)	15.9 (4.6)	26.5 (4.3)

Table 3: Mean task completion times for tasks completed using the standard cursor (standard deviations in parentheses).

	Straight			Diagonal		
	D1	D2	D3	D1	D2	D3
S1	39.5 (26.3)	44.4 (22.4)	65.2 (33.7)	34.1 (32.3)	45.2 (21.9)	65.2 (28.0)
S2	15.4 (11.2)	19.3 (7.6)	30.5 (13.9)	23.1 (11.2)	29.2 (23.7)	36.7 (8.3)
S3	7.0 (1.4)	12.0 (3.1)	18.5 (1.8)	10.4 (2.4)	15.4 (3.2)	29.0 (5.4)

Table 4: Mean task completion times for tasks completed using the predictive cursor (standard deviations in parentheses).

User confidence (i.e., use of "Stop" command)

Means and standard deviations for the number of "Stop" commands issued when selecting targets using the standard and predictive cursors are reported Tables 5 and 6 respectively. An ANOVA with repeated measures for direction, distance, and target size was utilized to assess the effect of cursor type. Cursor type did not have a significant effect on the use of the "Stop" command ($F(1,26)=0.30$, n.s.). Size and distance both had a significant effect on the use of the "Stop" command, but direction did not

($F(2,52)=25.59$, $p<0.001$; $F(2,52)=6.68$, $p<0.002$; $F(1,26)=0.15$, n.s. respectively). One significant interaction was identified: target size/direction. As expected, smaller targets resulted in increased use of the "Stop" command. longer selection times, supporting H2. Unexpectedly, longer distances also resulted in increased use of the "Stop" command.

	Straight			Diagonal		
	D1	**D2**	**D3**	**D1**	**D2**	**D3**
S1	2.48 (1.97)	4.13 (3.85)	4.36 (4.02)	2.82 (2.78)	2.55 (2.37)	3.55 (3.02)
S2	0.91 (1.14)	0.96 (1.07)	1.16 (1.45)	1.11 (1.39)	1.05 (1.24)	2.21 (3.93)
S3	0.18 (0.41)	0.20 (0.37)	0.30 (0.51)	0.25 (0.44)	0.23 (0.39)	0.29 (0.65)

Table 5: Means for the number of times the "Stop" command was issued for tasks completed using the standard cursor (standard deviations in parentheses).

	Straight			Diagonal		
	D1	**D2**	**D3**	**D1**	**D2**	**D3**
S1	3.36 (4.72)	4.21 (3.54)	4.29 (3.94)	2.63 (3.27)	3.21 (2.84)	5.11 (7.53)
S2	1.11 (1.65)	0.96 (1.08)	1.59 (2.82)	1.66 (2.29)	2.32 (4.19)	1.57 (1.70)
S3	0.25 (0.42)	0.30 (0.41)	0.39 (0.54)	0.48 (0.74)	0.41 (0.63)	0.61 (1.01)

Table 6: Means for the number of times the "Stop" command was issued for tasks completed using the standard cursor (standard deviations in parentheses).

Satisfaction

Means and standard deviations for the questionnaire results are reported Table 7. Two sample t-tests confirmed that there were no significant differences due to the type of cursor participants used.

	Speed	Accuracy	Comfort
Standard	3.29 (0.83)	3.43 (0.94)	2.36 (1.01)
Predictive	3.50 (1.45)	3.50 (1.22)	2.50 (1.09)

Table 7: Subjective ratings of speed, accuracy, and comfort when using the standard and predictive cursors. Ratings were provided on a scale from 1 (positive) to 5 (negative). Standard deviations are in parentheses.

DISCUSSION

Cursor type did not have an effect on selection times, confidence, or satisfaction, but there were two significant interactions involving cursor type when analyzing errors. Unfortunately, a more detailed analysis confirms that the predictive cursor is associated with higher error rates for diagonal movements. Under the current conditions, the predictive cursor does not appear beneficial. Neither H1 nor H5 were supported.

Target size did effect selection times, errors, and confidence, providing strong support for H2. Distance had an effect on selection times, providing partial support for H3, but the interactions between distance and size must be explored in more detail to determine if H3 is fully supported by our data. This additional analysis is provided below. Unexpectedly, distance also had a significant effect on user confidence, with users issuing more "Stop" commands when moving the cursor larger distances. Direction had an effect on errors and times, providing partial support for H4, but interactions involving size and direction must be explored in more detail to determine if H4 is fully supported by our data. This analysis is also provided below.

Since the predictive cursor provides no benefits, the remainder of this discussion focuses on the results obtained when participants used the standard cursor. The goal is to provide insights which could be used to design more effective speech-based cursor control mechanisms as well as more effective interfaces given existing speech-based cursor control mechanisms.

Selecting a target involves moving the cursor a predefined distance, positioning the cursor over the target, and issuing the "Click" command. Since the cursor can only be moved horizontally or vertically, the analyses that follow use the actual distance the cursor would have to move to complete the selection rather than straight-line distance. As a result, distances for straight movements were 1.9cm, 3.8cm, and 7.6cm while distances for diagonal movements were 2.7cm, 5.4cm, and 10.7cm. Straight movements require a single positioning activity and can be completed with as few as two commands (i.e., "Move" in the correct direction and "Click"). Diagonal movements require two positioning activities (i.e., one horizontal and one vertical) and will require at minimum of three commands (e.g., "Move" horizontally, "Move" vertically, and "Click").

When targets are sufficiently large (large targets):

- The time required to select a target is a linear function of the distance the cursor must move ($r^2=0.98$, Figure 3). The regression equation results in a Y-intercept of only 3.87 and a slope of 5.56, indicating that it takes 5.56 seconds for every centimeter the cursor must be moved.
- It is easy to position the cursor over the target. This is illustrated by low error rates (0.06 errors/target, Figure 4) and limited use of the "Stop" command (0.24 commands/target, Figure 5). The small Y-intercept (3.87, Figure 3) also supports this observation.
- Differences between straight and diagonal movements are minimal. This is illustrated by the lack of significant differences involving the direction variable

as well as the strong linear relationship between distance and selection times as illustrated in Figure 3.

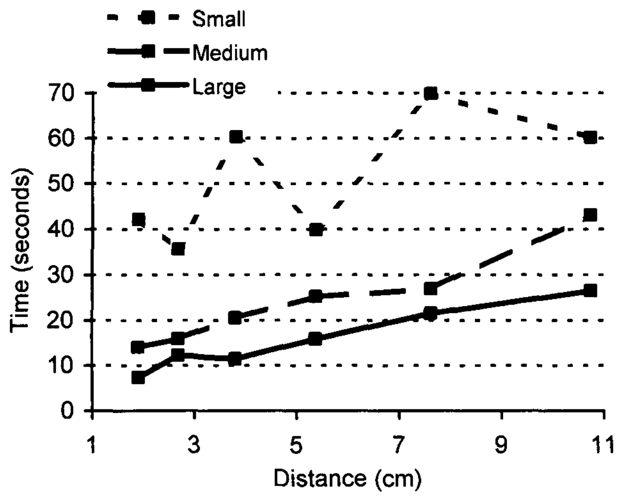

Figure 3: Time required to select targets of various sizes placed various distances from the original cursor location using the standard cursor (in seconds).

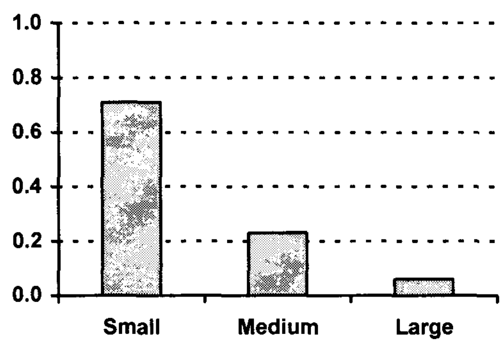

Figure 4: Average error rates when using the standard cursor to select targets of various sizes.

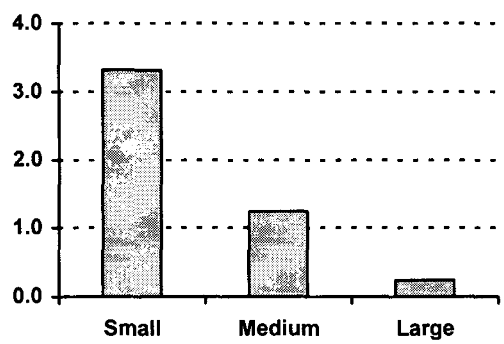

Figure 5: Number of "Stop" commands issued when selecting targets of various sizes using the standard cursor.

As targets become smaller (medium targets):

- The relationship between distance and time is still linear (r^2=0.95, Figure 3), but the Y-intercept increases to 10.88. The slope increases to 6.69.
- It is more difficult to accurately position the cursor over the target. Errors increase (0.23 errors/target, Figure 4), use of the "Stop" command increases (1.23 commands/target, Figure 5), and the Y-intercept is larger (10.88, Figure 3).
- Since positioning the cursor is more difficult, and diagonal movements require two positioning activities, diagonal movements begin to take longer than straight movements of the same distance. This is illustrated by the bumps at 5.4cm and 10.7cm on the medium target line in Figure 3 that correspond to diagonal movements.

When targets become so small they cannot be selected effectively using the current positioning commands (small targets):

- The linear relationship between time and distance is weaker (r^2=0.58) with a Y-intercept of 33.15 and a slope of 8.05. The Y-intercept, which represents the positioning times (as opposed to the movement time) increases substantially, is more variable, and begins to dominate the total time required to select a target.
- It is much more difficult to position the cursor over the target. Errors increase dramatically (0.71 errors/target, Figure 4), use of the "Stop" command increases dramatically (3.32 commands/target, Figure 5), and the Y-intercept is much larger (33.13, Figure 3).
- Since positioning the cursor is quite difficult, and diagonal movements require two positioning activities, one would expect diagonal movements to take longer than straight movements of the same distance. Interestingly, this does not happen. In fact, as illustrated by the three peaks along the small target line in Figure 3, straight movements actually take longer than diagonal movements. This suggests that, for targets of this size, selection time is no longer a function of the distance the user must move the cursor.

Overall, these results appear to prove strong support for H3. Results are mixed for H4. As predicted, selection time is dominated by the distance the cursor must move when targets are sufficiently large and positioning time becomes more critical when targets get smaller. However, diagonal movements did not result in increased error rates as compared to straight movements. It appears that users simply spent more time to ensure that errors did not increase for these tasks.

CONCLUSIONS

The predictive cursor did not provide the expected benefits. We believe this may be due to the relatively short and consistent delay which allows users in the standard cursor condition to predict where the cursor would stop at any given time. The difficulty users experience selecting targets appears to be a result of the speed at which the cursor was moving, the size of the target, and the delay itself.

In the current study, the cursor moved at a rate of 20 pixels/second. As a result, users had approximately 0.6sec in which to issue the "Stop" command when selecting small targets to ensure that the cursor would overlap the target (1.2sec for medium targets and 2.4sec for large targets). As expected, users had little difficulty accurately selecting the large targets while medium targets resulted in increased use of the "Stop" command, more errors, and longer selection times as compared to large targets. The strong linear relationship between selection times and distance for both the large and medium targets suggests that both large and medium targets are sufficiently large that they could be reliably selected by our participants.

Small targets result in even more "Stop" command usage, much higher error rates, and even longer selection times. More importantly, the linear relationship between distance and selection times is much weaker and the positioning time (Y-intercept) accounts for a substantial portion of the total selection time. While positioning the cursor is clearly difficult, and diagonal movements require two positioning activities, straight movements require more time than diagonal movements of the same distance. These results suggest that our participants could not reliably select the small targets.

The current speech-based cursor control mechanism allows users to select relatively large targets reliably, but not quickly. Future studies must investigate mechanisms that will allow smaller targets to be selected reliably while allowing all targets to be selected more quickly. Several follow up studies are underway that will provide additional insights into the effects of cursor speed, target size, and the delays associated with spoken input on selection times and error rates. Given the importance of the delays associated with spoken input, an additional studies underway that will provide insights into the magnitude of each of the three components of this delay.

ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation under Grants IIS-9910607 and IIS-0121570. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation (NSF). We would also like to acknowledge support provided by the UMBC Provost's Undergraduate Research Program.

REFERENCES

1. Ainsworth, W. A. (1988). Optimization of string length for spoken digit input with error correction. *International Journal of Man-Machine Studies*, 28, 573-581.
2. Ainsworth, W. A. and Pratt, S. R., (1992). Feedback Strategies for Error Correction in Speech Recognition Systems. *International Journal of Man-Machine Studies*, 36, 833-842.
3. Baber, C. and Hone, K. (1993). Modeling error recovery and repair in automatic speech recognition. *International Journal of Man-Machine Studies*, 39, 495-515.
4. Christian, K., Kules, B., Shneiderman, B. and Youssef, A. (2000). A comparison of voice controlled and mouse controlled web browsing. *Proceedings of Assets 2000*, pp. 72-79.
5. Danis, C., Comerford, L., Janke, E., Davies, K., Devries, J. and Bertrand, A. (1994). Storywriter: A Speech Oriented Editor, *CHI 94 Conference Companion*. pp. 277-278.
6. de Mauro, C., Gori, M., Maggini, M. and Martinelli E. (2001). Easy access to graphical interfaces by voice mouse. Available from the author at: demauro@dii.unisi.it
7. Halverson, C., Horn, D., Karat, C-M. and Karat, J. (1999). The beauty of errors: patterns of error correction in desktop speech systems. *Proceedings of INTERACT'99*. IOS Press, pp. 133-140.
8. Hauptmann, A. G. (1989). Speech and gestures for graphic image manipulation. *Proceedings of CHI'89*, pp. 241-245.
9. Karat, C-M., Halverson, C., Karat, J. and Horn, D. (1999). Patterns of Entry and Correction in Large Vocabulary Continuous Speech Recognition Systems. *Proceedings of CHI 99*, pp. 568-575 .
10. Manaris, B. and Harkreader, A. (1998). SUITEKeys: A speech understanding interface for the motor-control challenged. *Proceedings of the 3rd International ACM SIGCAPH Conference on Assistive Technologies (ASSETS'98)*, pp. 108-115.
11. McNair, A. and Waibel, A. (1994). Improving recognizer acceptance through robust, natural speech repair. *Proceedings of the International Conference on Spoken Language Processing*, pp. 1299-1302.
12. Noyes, J. M. and Frankish, C.R. (1994). Errors and error correction in automatic speech recognition systems. *Ergonomics*, 37, 1943-1957.
13. Sears, A., Karat, C-M., Oseitutu, K., Karimullah, A. and Feng, J. (2001). Productivity, satisfaction, and interaction strategies of individual with spinal cord injuries and traditional users interacting with speech recognition software. *Universal Access in the Information Society*, 1, 4-15.
14. Suhm, B., Myers, B. and Waibel, A. (2001). Multimodal error correction for speech user interfaces. *ACM Transactions on Computer-Human Interaction*, 8(1), 60-98.