Speech-Based Navigation: Improving Grid-Based Solutions

Shaojian Zhu[1], Yao Ma[2], Jinjuan Feng[2], and Andrew Sears[1]

[1] UMBC, 1000 Hilltop Circle, Baltimore, MD, USA, 21250
{szhu1,asears}@umbc.edu
[2] Towson University, 7800 York Road, Towson, MD, USA, 21252
{yma1,jfeng}@towson.edu

Abstract. Speech-based technology is a useful alternative to traditional input techniques such as the keyboard and mouse. For people with disabilities that hinder use of traditional input devices, a hands-free speech-based interaction solution is highly desirable. Various speech-based navigation techniques have been discussed in the literature and employed in commercial software applications. Among them, grid-based navigation has shown both potential and limitations. Grid-based solutions allow users to position the cursor using recursive grids to 'drill down' until the cursor is in the desired location. We report the results of an empirical study that assessed the efficacy of two enhancements to the grid-based navigation technique: magnification and fine-tuning. Both mechanisms were designed to facilitate the process of selecting small targets. The results suggest that both the magnification and the fine-tuning capabilities significantly improved the participants' performance when selecting small targets and that fine-tuning also has benefits when selecting larger targets. Participants preferred the solution that provided both enhancements.

Keywords: MouseGrid, Speech-based Cursor Control, Accessibility.

1 Introduction

Speech recognition technology enables people to use speech as an alternative input method when interacting with computers. Some commercial operating systems like Windows Vista™ provide speech functions for a variety of graphical interface-based tasks such as interacting with the desktop applications and browsing the web [1]. When using computers, two types of tasks account for the vast majority of user activity: text generation and navigation [2]. Previous research has confirmed that modern speech recognition technology can be quite effective for text generation tasks [3], but it remains difficult for users to complete spatial navigation tasks using speech [4]. A number of researchers suggested that the use of pointing devices should not be completely eliminated when people use speech recognition to interact with computers [5]. Multimodal solutions that use speech for text generation and pointing devices for

T. Gross et al. (Eds.): INTERACT 2009, Part I, LNCS 5726, pp. 50–62, 2009.

spatial navigation are often recommended for people who can use their hands effectively [6].

However, people with disabilities experience lots of difficulties when using traditional input devices like the mouse. Therefore, a hands-free speech-based solution is likely to be more effective for disabled people. For example, there are a number of diseases and conditions that can affect the hands and arms, such as high level Spinal Cord Injuries (SCI), Amyotrophic Lateral Sclerosis (ALS), or stroke, making a mouse difficult to use [7]. Compared with other alternatives such as head-controlled devices, eye-controlled interactions, or electrophysiological solutions, speech-based solutions are less expensive, provide a more natural interaction style, and can be easier to learn. In order for speech technologies to be widely adopted by users with disabilities that affect their hands or arms, more effective speech-based cursor control solutions are highly desired.

Resources have been invested in this area, seeking to identify better ways for people to use speech recognition to easily complete the cursor control tasks. Grid-based solutions, developed by Kamel for blind users [8], and subsequently evaluated by sighted users [9], provide a flexible, reliable, alternative but opportunities for improvement remain. The grid-based method allows the user to select targets on screen by 'drilling-down' to a smaller grid until the cursor is positioned in the desired location. However, in a recent field study that focused on how people interact with speech technologies in realistic environments, we found the widely adopted version of grid-based cursor control is inadequate, resulting in poor performance and satisfaction ratings [10].

Based on our previous research [10], we proposed two enhancements to the grid-based solution: magnification and fine-tuning. A software prototype was developed integrating the two enhancements, and an empirical study was conducted to evaluate the efficacy of the enhancements. The results confirm that both fine-tuning and magnification resulted in significant improvements. In this paper, we discuss the motivation behind the two enhancements, how the empirical study was conducted, and the major results. Implications and directions for future research are discussed at the end of the paper.

2 Related Research

There are a number of research projects that used speech as one of the interface modalities. Substantial advancements have been made in speech recognition systems regarding both recognition speed and accuracy, with dictation accuracy reaching as high as 98% in controlled environments [11, 12]. However, a hands-free speech-enabled system must support both dictation and cursor control [2] and difficulties dealing with spatial tasks using speech have been discussed for years [6].

There are two main categories of cursor control for speech-enabled systems: direction-based and target-based solutions. Alternative speech-based cursor control solutions can also be discussed based on the type of movement that results: discrete or continuous movements [9]. For discrete direction-based navigation, users specify the movement direction and distance, such as ‘Move left three words’. The distance can be specified in inches, centimeters, words or other units that are appropriate for the situation. In some situations, a default distance such as one word or line is assumed. For continuous direction-based solutions, users must begin by issuing a command that initiates the movement (e.g., move left). Responding to the command, the cursor moves smoothly in the specified direction at a fixed speed until being stopped by another command (e.g., ‘stop’). The direction-based approach can allow users to control the movement direction and distance, but it also has substantial disadvantages. First, performance is influenced by the relative position of the cursor and target location. Second, the interaction is not natural for many users. Third, it does not provide enough flexibility for different contexts. Fourth, selection of small targets can be both slow and error prone.

Compared to direction-based navigation, target-based cursor control takes advantage of contextual information, allowing users to select targets by specifying the name of the desired object. For example, ‘select Friday’ could move the cursor to the word ‘Friday’ in a text document. For many users, this technique is more natural and direct, but it only works when all possible target have a name that is known by the user [13]. As graphical interfaces become more complex, many potential targets may not have clearly visible label accompanying them and the use of existing target-based techniques can become difficult. When the label is not visible, and multiple targets share the same name, target-based solutions become more difficult and less efficient.

A number of studies have focused on making speech-based navigation techniques more effective. Some focused on speech-based navigation techniques in the context of text documents [3, 14] while others focused on general target-selection tasks [13]. In general, target-based navigation has proven effective in the context of text documents, but less effective in the context of desktop interaction when the names of icons or targets are not clearly labeled. Direction-based techniques that result in continuous cursor movement have been reported to be both slow and error prone [13], and direction-based techniques that use contextual information (e.g., number of lines and words) may prove challenging since the only units available to specify distances are pixels or physical distances (e.g., inches or centimeters) both of which users are likely to have difficulty estimating accurately.

The grid-based technique was proposed as a way to select target without contextual information. Using this technique, the user recursively drills down through each grid until the cursor is placed on the desired object. Kamel and Landay [8, 15, 16] developed a speech-based grid drawing tool for blind people, employing a 3x3 grid, demonstrating the potential of grid-based speech cursor control. Dai [2] built on these results and evaluated two alternatives: the traditional solution with a single cursor in the middle of the grid and an alternative which placed a cursor in the center of each of the nine cells of the grid. The results indicated that the nine cursor solution was faster but resulted in more errors. To date, the grid-based approach has shown potential for desktop interactions when the target is fairly large (e.g., desktop icons), but it becomes increasingly cumbersome as targets get smaller (e.g., word, menu icons, letters). When targets are sufficiently small, grid-based navigation becomes slow and error prone. Some commercial speech-based business systems like Windows Vista™ provide variants of grid based cursor control. Within Windows Vista™, grid-based navigation is available through the MouseGrid command.

Compared to the target-based technique, grid-based solutions should result in less cognitive load because the grid numbers can be both visible and intuitive and the target selection procedure is straight forward. Another advantage is that the grid-based technique can position the cursor anywhere on the screen (e.g., a blank space on the screen), something that is not possible with target-based solutions. As discussed by Dai et al. [9], grid-based cursor control can place the cursor at any point on screen in N steps which can be calculated based on Formula 1:

$$N = \mathrm{Log}_n(D/A). \tag{1}$$

In the formula, n is the number of grid rows/columns (normally three), D is the resolution of screen and A is the size of the target. The effectiveness of the grid-based approach is affected by the relative position of the cursor and target. If a target is located near the center of a grid, the user may be able to select it easily. More importantly, selecting small targets using the grid-based approach tends to be problematic. When the targets are small, the user may need to issue five or more commands to focus on smaller portions of the screen, with each command making it somewhat harder to determine the relative position of the target and the grid [13]. At the same time, grid lines and the numbers in the cells of the grid can become distracting. To address these challenges, we proposed the following two enhancements, magnification and fine-tuning to the grid-based approach.

3 Software Prototypes

To make it easier for users to select small targets, we proposed two enhancements to the grid-based technique: magnification and fine-tuning. Once the grid becomes sufficiently small, the magnification function shows the user a magnified version of the grid making it easier to see small targets. The fine-tuning function allows the user to fine-tune the cursor location using four simple commands: ‘up, down, right, and left’. Each command moves the cursor to the specific direction by a specific number of pre-defined units, making it easy to shift the cursor small distances when that is all that is required to select a target.

The first version implements the basic grid-based navigation technique. The second implements the magnification capability. When user zooms a third time, the magnification capability is automatically activated simplifying the selection of small targets (See fig. 1a and 1b). The user can continue zooming if desired, with the selected cell being magnified each time. The third version implements the fine-tuning capability. After zooming a third time, the application stops zooming. Instead, a new cursor is presented indicating the current location (See fig. 1c and 1d). This cursor can be repositioned using the four fine-tuning commands. The fourth condition implements both magnification and fine-tuning (see fig. 1e and 1f).

Fig 1a and fig 1b show the transition after the user issued the command ‘seven’ with the magnification capability. The standard technique would shrink the focus into the seventh cell of the grid. With the magnification capability, the system will not only focus on cell seven, but also magnifies that region to make it more visible.

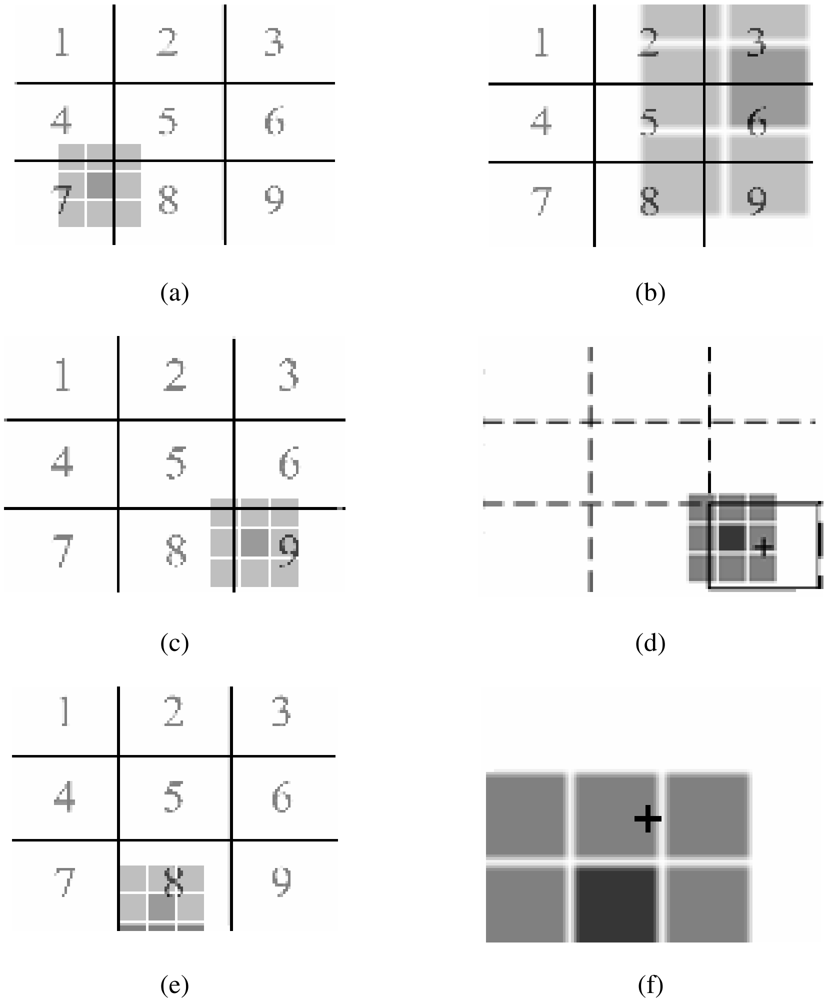

Fig. 1. (a) Before magnification. (b) After magnification. (c) Before fine-tuning. (d) After fine-tuning. (e) Before magnification & fine-tuning. (f) After magnification & fine-tuning.

Fig 1c and fig 1d show the transition after the user issued the command ‘nine’ with the fine-tuning capability. The basic grid-based technique just focuses on the ninth cell. With the fine-tuning function, the system focuses on cell three, then enables the direction-based fine-tuning commands (up, down, left, and right). The actual cursor is visualized using a black cross.

Fig 1e and fig 1f show the transition after the user issued the command ‘eight’ with both magnification and fine-tuning capabilities.

4 Methods

4.1 Participants

Twenty native English speakers (12 Males, 8 Females) volunteered to participate in our experiment. None of them had any cognitive or motor impairments. The average age of our participants was 21.3 (stdev = 3.36), and the average computer experience was 13.3 years (stdev = 4.09). Most of the participants had information technology related backgrounds. Each participant completed all four conditions with each condition requiring the selection of 40 targets.

4.2 Equipments

A PC running Windows XP was used for this study. A 17-inch LCD non-wide screen monitor was used for the visual output with the resolution set to 1024x768. Participants sat at a comfortable distance from the display. Voice commands were processed using the Microsoft SAPI5.1 speech recognition engine. Each participant used the same Andrea NC61 headset microphone to complete the study. A C# application was used to implement the prototypes with logging functions that recorded interaction activities including task completion times and target/cursor positions.

4.3 Experiment Design

A within-group experiment design was adopted. Each participant finished all four conditions listed in table 1. Condition 1 was the baseline condition in which neither of the enhancements was offered. The order in which participants completed the conditions was randomized.

Table 1. Four Experimental Conditions with different settings

	Without magnification	**With magnification**
Without fine-tuning	C1	C2
With fine-tuning	C3	C4

At the beginning of the study, the participant was given a training session with four target selection tasks to allow them to become familiar with the speech-based cursor control solution. Multiple training sessions were given upon request. After the training session, the participant completed 40 target selection tasks as part of each condition. For each task, the location of the target was randomly defined. The target size was randomly selected among four choices: 10x10, 20x20, 40x40 and 80x80 pixels (i.e., square targets 3.37, 6.74, 13.49 and 26.98 millimeters respectively). The four possible sizes were selected to represent the sizes of common graphical user interface components: letters (10), words/menus/small icons (20), buttons (40), and desktop icons (80). Under each condition, the participant selected a total of ten targets of each size.

4.4 Independent and Dependent Variables

The independent variables of interest include the type of grid-based navigation supported and target size. The dependent variables examined include target selection time, error rates, and subjective satisfaction ratings. An error was documented when a participant issued the ‘ok’ command to select a target and the cursor was located outside of the target area.

A questionnaire using a 5-point Likert-scale (1 as most positive and 5 as most negative) assessed the participants’ subjective perceptions of speed, accuracy, and comfort level after each condition. A general questionnaire was completed after all conditions asking the participants to rank order the four alternative solutions (1 as most favorable and 4 as least favorable). Demographical information was also collected via the general questionnaire.

4.5 Hypothesis

We investigated the impact of magnification and fine-tuning capabilities on user performance and user satisfaction. User performance is measured by task completion time and error rates. User satisfaction is measured via subjective ratings collected through questionnaire. We tested the following hypotheses:

H1a: Task completion time will be shorter in the magnification condition than the baseline condition.

H1b: Task completion time will be shorter in the fine-tuning condition than the baseline condition.

H1c: Task completion time will be the shortest for the condition providing both magnification and fine tuning.

H2a: Error rate will be lower in the magnification condition than the baseline condition.

H2b: Error rate will be lower in the fine-tuning condition than the baseline condition.

H2c: Error rate will be the lowest for the conditions providing both magnification and fine tuning.

We expect that the size of targets play an important role for user performance, with smaller targets being harder to select. The following hypotheses are proposed with regard to target size and user performance.

H3a: Participants will spend longer selecting smaller targets.

H3b: Participants’ error rates will be higher when selecting smaller targets.

Finally we examine whether there is significant difference in user satisfaction among the four conditions. The following hypotheses are proposed related to user satisfaction:

H4a: The magnification and fine-tuning capabilities will lead to more positive subjective assessments.

H4b: Users will prefer conditions with magnification and fine-tuning capabilities as compared to the control condition.

5 Results

5.1 Selection Time

Mean target selection times for each target size under each condition are reported in Table 2 and illustrated in Fig 2. A repeated measures ANOVA with target selection time as the dependent variable and target size and condition as independent variables confirmed a significant effect for both condition (F (3, 57) = 4.85, p < 0.005) and target size (F (3, 57) = 183.4, p < 0.001). The interaction between size and condition is not significant (F (9, 171) = 1.72, n.s.).

Table 2. Average task completion time for each condition/size (standard deviations in parentheses)

	size 10	**size 20**	**size 40**	**size 80**
Basic (N/A)	143.1 (37.9)	109.2 (22.7)	92.8 (11.5)	89.0 (15.9)
Magnification Only	125.4 (22.6)	100.9 (12.8)	92.7 (13.7)	87.3 (16.2)
Fine-tuning Only	126.1 (11.0)	103.9 (8.4)	82.7 (6.2)	79.9 (6.2)
Both	125.7 (15.4)	101.7 (10.6)	83.9 (8.9)	81.3 (6.6)

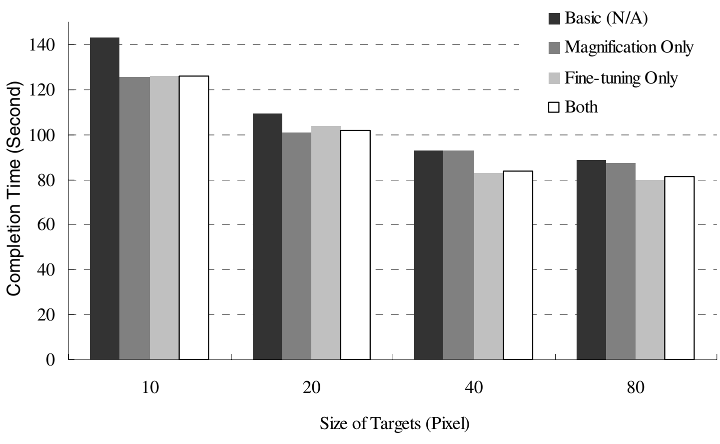

Fig. 2. Average task completion time for each condition/size

The different grid-based solutions resulted in significantly different target selection times for targets of size 10, 40 and 80 (F (3, 57) = 2.91, p < 0.05; F (3, 57) = 6.39, p < 0.001; F (3, 57) = 3.90, p < 0.05). However, difference for targets of size 20 was not significant (F (3, 57) = 1.51, n.s.). Table 3 summarizes the results of the one tailed Post Hoc tests comparing specific condition/target size combinations. When the target size was 10, participants spent significantly longer with the baseline solution as compared to any of the other solutions, suggesting both magnification and fine-tuning improved performance. When the targets were size 40 or 80, participants spent significantly less time in conditions that offered the fine-tuning enhancement as compared to those that did not. Overall, the results suggest that the magnification function did not improve performance for relatively large targets, but when targets are sufficiently small magnification is beneficial. In contrast, fine-tuning was useful for both large and small targets. Therefore, H1a is supported for small targets only. H1b is supported for both large and small targets. H1c was not supported (i.e., the benefits of magnification and fine-tuning were not additive).

Table 3. Summary of Post Hoc tests for Repeated Measure ANOVA

Conditions under comparison	Size 10	Size 40	Size 80
Baseline vs. Magnification	$p < 0.05$	n.s.	n.s.
Baseline vs. Fine-tuning	$p < 0.05$	$p < 0.001$	$p < 0.05$
Baseline vs. M & F-T	$p < 0.05$	$p < 0.05$	$p < 0.05$
Magnification vs. Fine-tuning	n.s.	$p < 0.05$	$p < 0.05$
Magnification vs. M& F-T	n.s.	$p < 0.05$	$p < 0.05$
Fine-tuning vs. M & F-T	n.s.	n.s.	n.s.

A regression analysis with time as the dependent variable and condition and target size as independent variables show a significant impact of both condition and size. Size explained substantially more variance in task time than condition (48.6% vs. 2.5%). So H3a is supported.

5.2 Accuracy

The mean error rate for each target size and condition combination are reported in Table 4 and illustrated in Fig 3. A repeated measures ANOVA with error rates as the dependent variable and size and condition as independent variables confirms that condition did not have a significant effect on error rates ($F(3, 57)=0.57$, n.s.). H2a, H2b, and H2c are not supported.

Target size did have a significant effect on error rates ($F(3, 57)=6.84$, $p < 0.01$) with participants making more errors when selecting smaller targets. The interaction between size and condition was not significant ($F(9, 171) = 0.87$, n.s.). H3b is supported.

Table 4. Average error rates (%) for each condition/size (standard deviations in parentheses)

	size 10	size 20	size 40	size 80
Basic (N/A)	4.5 (8.8)	1.0 (3.0)	0.0 (0.0)	1.5 (3.6)
Magnification Only	5.0 (0.0)	1.0 (4.4)	0.5 (2.2)	1.5 (4.8)
Fine-tuning Only	2.5 (5.5)	1.5 (4.8)	0.0 (0.0)	0.0 (0.0)
Both	2.0 (4.1)	0.5 (2.2)	1.5 (3.6)	1.0 (3.0)

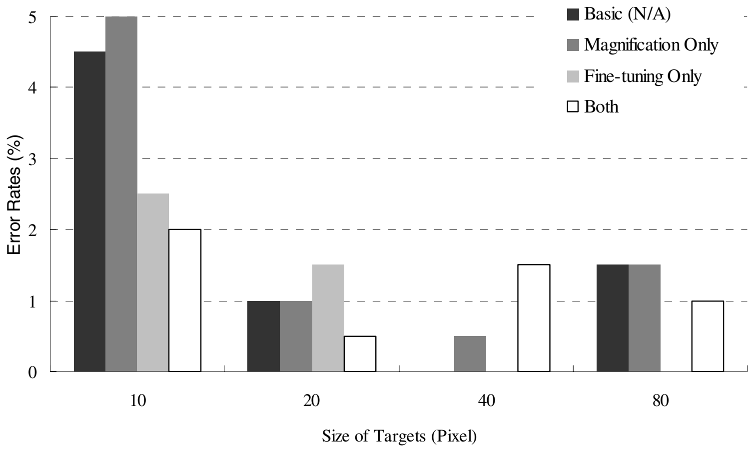

Fig. 3. Average error rates (%) for each condition/size

5.3 User Satisfaction Rating

Mean user ratings concerning speed, accuracy and comfort are reported in Table 5. A repeated measures ANOVA confirmed that condition did not have a significant effect on user satisfaction with speed (F(3, 57)=1.5, n.s.), accuracy (F(3, 57)=3.0, n.s.), and or comfort (F(3, 57)=1.2, n.s.). H4a is not supported.

A CHI squared test did identify a significant difference among the rankings provided for the four solutions (X^2 (9) = 35.6, $p < 0.001$). Table 6 indicates the

Table 5. Subjective rating or speed, accuracy, and ease of use Ratings were provided from 1 (most positive) to 5 (most negative) (Standard deviations in parentheses)

	Speed	Accuracy	Comfort
Basic (N/A)	2.05 (0.75)	1.55 (0.68)	1.55 (0.60)
Magnification Only	1.80 (1.00)	1.50 (0.82)	1.40 (0.59)
Fine-tuning Only	1.80 (0.69)	1.35 (0.58)	1.35 (0.58)
Both	1.75 (0.78)	1.40 (0.50)	1.40 (0.50)

Table 6. The number of participants who rated each solution as their first, second, third, or fourth preference

	First	Second	Third	Fourth
Basic (N/A)	3	1	5	11
Magnification Only	3	8	5	4
Fine-tuning Only	3	4	10	3
Both	11	7	0	2

number of participants who rated each condition as their first, second, third, or fourth preference. Eleven participants rated the solution that offered both magnification and fine-tuning as the best. Only three participants rated each of the other solutions as the best. H4b is supported.

6 Discussion

The data confirmed that the new capabilities (magnification and fine-tuning) did result in improved performance. Both magnification and fine-tuning significantly decreased task completion times compared to the baseline condition. However, neither magnification nor fine-tuning decrease the error rate significantly. We believe one possible reason is that the participants completed the tasks without time pressure. Therefore, they could spend as much time as necessary to achieve a desired level of accuracy. As we can see from Table 4, the highest error rate occurred under the magnification condition for targets of size 10, but even under this condition the participants' average error rate was just 5%. In general, users rarely made mistakes even for the smallest targets. It will be interesting to examine accuracy and efficiency in more detail. For example, it may prove useful to analyze the number of commands participants issued and the minimum number of commands required to select each target as well as the specific commands used.

Participants preferred the solution that provided both magnification and fine-tuning. While performance measures reveal no differences between this condition and those that provide a single enhancement, these results confirm that participants preferred to have both options available to be used if and when they desired. By providing both capabilities, users could fine-tune the cursor location as necessary including some situations when they were selecting large targets and the cursor was very close to being on the target. Similarly, users could take advantage of the magnification function when selecting small targets.

As expected, target size plays an important role with regard to performance, having a significant effect on both task completion times and error rates, which is consistent with the existing literature [9]. In addition, size explained a large portion of the variance in task completion time. While more effective grid-based navigation solutions can make selecting small target significantly faster, small targets will still require more time than larger targets. Therefore, designers of graphical user interfaces should still pay attention to the size of interface component to facilitate smooth interactions.

The pattern of results for task completion times is interesting. Clearly, magnification provided no benefit when selecting the two largest targets. For targets of size 20, a difference begins to emerge but it is not significant. Finally, when selecting the smallest targets, magnification allowed for significantly faster task completion times as compared to the baseline condition. It is possible that the lack of benefit for larger targets may be due directly to the size of the target or because users did not zoom in enough to activate magnification. A more detailed analysis of the specific commands issued may provide useful insights. For example, such an analysis may help provide a more definitive answer as to how small a target must be before magnification is useful.

The pattern of results is quite different for the solutions that provided the fine-tuning capability. In these conditions, users were able to select both large and small targets faster than they could with the baseline solution. While magnification was not useful when selecting large targets, it appears that there are some situations where the cursor may be close enough to a target that fine-tuning is still a useful alternative.

The results for the targets of size 20 may suggest that these targets are near the threshold where magnification first becomes useful. If this is the case, it could be that users end up spending more time deciding which enhancement to use, slowing the overall process of selecting the target. In future studies, we plan to examine the effect of target size in more detail by using a larger variety of target sizes.

While the solution that provided both magnification and fine-tuning was no more efficient than the solution that provided just fine-tuning, more than half of the participants rated the solution with both enhancements as the best. We suggest that this may be because this solution provides more flexibility than any of the other alternatives. Magnification is likely to be useful only for small targets, but fine-tuning could prove useful regardless of target size. Most importantly, users are free to use both magnification and fine-tuning if and when they believe it would be useful.

7 Conclusions

To address the limitations of existing grid-based navigation solutions, especially inefficiencies when selecting small targets, we proposed, implemented, and evaluated two enhancements: magnification and fine-tuning. Our empirical evaluation with 20 participants suggests that the fine-tuning function significantly reduced target selection time for both large and small targets while magnification was only useful when selecting small targets. Neither enhancement had a significant effect on error rates. Importantly, there was a clear preference for the solution that provided both magnification and fine-tuning. Besides, the results from this study imply the possible benefits for people with disabilities who are not able to use traditional input devices. We are currently conducting a study that involves individuals with physical disabilities to evaluate whether the results observed with able-bodied users can be generalized to users with disabilities.

Acknowledgments. This material is based upon work supported by the National Science Foundation (NSF) under Grant No. CNS-0619379 and National Institute on Disabilities and Rehabilitation Research (NIDRR) under grant number H133G050354. 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 NSF and the NIDRR.

References

1. Vista™ Speech Recognition, `http://en.wikipedia.org/wiki/Windows_Speech_Recognition`
2. Oviatt, S.L.: Multimodal interactive maps: Designing for human performance. In: Human-Computer Interaction, vol. 12, pp. 93–129. Springer, Heidelberg (1997)

3. Feng, J., Sears, A.: Using confidence scores to improve hands-free speech based navigation in continuous dictation systems. ACM Trans. Comput.-Hum. Interact. 11(4), 329–356 (2004)
4. Lai, J., Yankelovich, N.: Conversational speech interfaces. In: Jacko, J.A., Sears, A. (eds.) The Human-Computer interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications. Human Factors and Ergonomics, pp. 698–713. L. Erlbaum Associates, Hillsdale (2007)
5. Cohen, M.H., Giangola, J.P., Balogh, J.: Voice User Interface Design. Addison Wesley Longman Publishing Co., Inc., Redwood City (2004)
6. Oviatt, S.: Multimodal Interfaces. In: Jacko, J.A., Sears, A. (eds.) The Human-Computer interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications, pp. 413–432. L. Erlbaum Associates, Hillsdale (2007)
7. Sears, A., Young, M., Feng, J.: Physical Disabilities and Computing Technologies: An Analysis of Impairments. In: Jacko, J.A., Sears, A. (eds.) The Human-Computer interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications. Human Factors and Ergonomics, pp. 829–852. L. Erlbaum Associates, Hillsdale (2007)
8. Kamel, H., Landay, J.: The Integrated Communication 2 Draw (IC2D): A drawing program for the visually impaired. In: CHI 1999 (1999)
9. Dai, L., Goldman, R., Sears, A., Lozier, J.: Speech-based cursor control: a study of grid-based solutions. In: Assets 2004: Proceedings of the 6th international ACM SIGACCESS conference on Computers and accessibility, pp. 94–101. ACM, New York (2004)
10. Feng, J., Zhu, S., Hu, R., Sears, A.: Speech technology in real world environment: early results from a long term study. In: Assets 2008: Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility, pp. 233–234. ACM, New York (2008)
11. Halverson, C., Horn, D., Karat, C., Karat, J.: The Beauty of Errors: Patterns of Error Correction in Desktop Speech systems. In: Proc. INTERACT 1999, pp. 1–9 (1999)
12. Karat, C., Vergo, J., Nahamoo, D.: Conversational Interface Technologies. In: Jacko, J., Sears, A. (eds.) The Human-Computer Interaction Handbook, pp. 169–186. LEA, NJ (2003)
13. Karimullah, A.S., Sears, A.: Speech-based cursor control. In: Proceedings of Assets 2002, pp. 178–185 (2002)
14. Sears, A., Feng, J., Oseitutu, K., Karat, C.: Hands-free, speech-based navigation during dictation: difficulties, consequences, and solutions. Hum.-Comput. Interact. 18(3), 229–257 (2003)
15. Kamel, H., Landay, J.: A study of blind drawing practice: Creating graphical information without the visual channel. In: ASSETS 2000 (2000)
16. Kamel, H., Landay, J.: Sketching images eyes-free: A grid-based dynamic drawing tool for the blind. In: ASSETS 2002 (2002)