SlideBar: Analysis of a linear input device

Leslie E. Chipman, Benjamin B. Bederson, Jennifer A. Golbeck

University of Maryland, College Park
Department of Computer Science
Human Computer Interaction Laboratory
{gchipman, bederson, golbeck}@cs.umd.edu

ABSTRACT

The SlideBar is a physical linear input device for absolute position control of one degree of freedom, consisting of a physical slider with a graspable knob positioned near or attached to the keyboard. Its range of motion is directly mapped to a one dimensional input widget such as a scrollbar. The SlideBar provides absolute position control in one dimension, is usable in the non-dominant hand in conjunction with a pointing device, and offers constrained passive haptic feedback. These characteristics make the device appropriate for the common class of tasks characterized by one-dimensional input and constrained range of operation. An empirical study of three devices (SlideBar, mouse controlled scrollbar, and mousewheel) shows that for common scrolling tasks, the SlideBar has a significant advantage over a standard mouse controlled scrollbar in both speed and user preference and an advantage over the mousewheel in user preference.

KEYWORDS: two-handed input, absolute, scrolling mechanisms, input devices, passive constrained haptic feedback

BACKGROUND

Personal computers have come to be so popular largely because they are multi-function devices. They consist of general-purpose processors, general-purpose input devices, and general-purpose output devices. This is a fundamental design feature of today's personal computers, and it works remarkably well. However, this design represents one side of a trade-off that results in a computer that can do many tasks well, but does not necessarily do each individual task as well as a specialized device could. Certain very common tasks, such as scrolling, are a case in point. It is no surprise then, that design, implementation and evaluation of scrolling mechanisms and pointing devices are well-studied areas.

Scrolling is one example of a common class of computer input tasks characterized by having one dimension of freedom and a constrained range of operation. Some other examples are selecting from menus, zooming in and out of documents and operating onscreen widgets, e.g., slider widgets. Even if there is not an explicit mouse controlled widget, the task can still exist and in current applications is often controlled via keyboard or with specific selections from menu items. For instance, photo browsing applications commonly have a menu for controlling the magnification with common values such as 50%, 100% or 200%, or in specific modes, the user may increment or decrement at predefined values with a mouse click, giving limited control of a task that actually has a smooth, broad range of input values.

Since this common class of tasks is broad and well defined, it seems logical to consider an input device tailored specifically for this purpose. In this paper, we introduce the SlideBar, a linear input device for absolute position control, which we built with the intention of providing users with an input device better suited for linear tasks. We delve into the reasons we expect this device to be well suited for linear tasks and give the results of a study comparing it to common input devices for the specific linear task of scrolling.

As computer users, we have become accustomed to using the standard mouse and keyboard as input devices and so it may be difficult to recognize some of their inherent drawbacks. There is more willingness to incorporate new devices into the interface, especially with the increasing popularity of the laptop computer where the limitations of a mouse become more pronounced. We believe the SlideBar offers a large enough advantage to justify becoming a standard input device.

The SlideBar

The SlideBar is a physical linear input device that can be used to control absolute position of a one degree of freedom parameter (figure 1). It is intended primarily for scrolling, although it can also be used to control any other one-degree of freedom application. It is designed for non-dominant hand use, to be used in tandem with a mouse or other general-purpose pointing device in the dominant hand.

[Insert figure 1 here]

In traditional graphical user interfaces (GUI) interfaces, a document such as a spreadsheet, text file, or file list in a folder is often larger than the viewing window. Thus, the user views only part of the document at a time. To view portions of the document outside the viewing window, one must move the window relative to the document. This behaviour, called scrolling, is an extremely frequent task in GUIs, and it becomes even more frequent in browsing World Wide Web pages.

Because the SlideBar is an absolute positioning device, the user can immediately move through the entire document, from top to bottom and through all the intermediate positions, although with limited resolution. Because the document being scrolled has a top, bottom and a length, and the SlideBar has a top, bottom and a length, there is always a direct linear mapping between the SlideBar and the scrolled document.

One issue with absolute positioning is that if a document is scrolled by a mechanism other than the SlideBar, the position of the SlideBar may no longer correspond to the viewing window of the document. The control software has been designed so that as soon as the SlideBar is moved at all, the document viewing windows jumps to the position that corresponds to the SlideBar. This is important since we hope that the prototype SlideBar can take advantage of the fact that it has a physical position in space, which allows users to develop awareness of what portion of the document they are viewing.

Scrolling

We have observed that users scroll primarily in two modalities. The first modality is to read a document, where the document is slowly scrolled a few lines at a time so that the user can easily track the line being read. We will refer to this as read mode. The second modality is to scan a document, typically when a user is searching the document to find a specific part of it. In this mode, the user typically scrolls the document back and forth through large areas. We will refer to this as scan mode. After the specific area sought for is found, the user typically switches to read mode, and reads the document line by line.

Scrolling Mechanisms

The most basic scrolling mechanism is the standard scrollbar controlled by a mouse. Other common methods include the mousewheel, keyboard, touchpad or joystick.

The scrollbar: A scrollbar is an on-screen widget that the user controls with a general pointing device, such as a mouse, joystick, or touchpad. It supports absolute motion by acquiring and dragging the movable “thumb”, which allows the user to scroll through the document to any given position in a simple motion, independent of the document length. This is most appropriate for scan mode and the thumb provides lower resolution control for longer documents. Scrollbars also support relative motion by clicking on the arrows at the ends of the scrollbar to scroll one line at a time, or on the remaining space between the arrows and the handle, which scrolls several lines at a time. This is appropriate for read mode.

The scrollbar seemingly is an excellent control mechanism for scrolling since it provides support for relative and absolute motion and for both modalities of use. Of course, it does work well since we all use it regularly. However, scrollbars do have several shortcomings. The primary one results from the fact that the user must use a general-purpose pointing device to control it. Each movement of the user’s hand between the keyboard and the mouse takes time. A study by Douglas and Mithal found it takes 0.67 seconds for users to acquire a mouse and 0.44 seconds for a keyboard joystick [5]. We would expect a time somewhere in this range for acquiring a keyboard mounted SlideBar.

Even if the user's hand is already on the mouse, the user must then move the pointer on the screen to the appropriate part of the scrollbar. Since common scrollbars are only about fifteen pixels wide, acquiring the scrollbar takes up to two seconds [17], and impacts performance, in accordance with Fitt’s Law [7]. Another shortcoming of scrollbars is that the user must take his or her eyes off the document and focus on the scrollbar to know where they are in the document.

Finally, for ongoing scrolling tasks that last a long time, the user must hold down the mouse button for the full length of the scrolling task (if using the thumb), or regularly click (if using the trough or arrows). For users with Repetitive Stress Injury (RSI), this continuous force required of the fingers can be destructive.

These actions are in conflict with a basic goal of good user interfaces: unobtrusiveness, transparency and ease of use. A good interface diverts a minimal amount of the user's attention and effort away from the primary task of viewing a document in order to explicitly manipulate GUI widgets.

The mousewheel: A mousewheel is a relatively new device, which is commonly built into new mice. It is a wheel between the two primary input buttons, which can be mapped to control the scrollbar with relative motion. It can be used to scroll a document a small amount with a corresponding wheel movement. For this reason its supports read mode well. Because the range of motion is limited the user must repetitively spin the wheel to scroll more than a few lines, making it ineffective for scan mode in all but the smallest documents.

Some applications support rate scrolling with a mousewheel. With rate scrolling, the user presses and holds the mousewheel down, then the direction of movement of the mousewheel provides the direction of scrolling and the amount of displacement controls the rate of scrolling. In a comparison of various scrolling devices and mechanisms, Zhai et al. [17] showed the performance of a mousewheel to be similar to that of the standard mouse. They also show rate scrolling to be more effective than standard scrolling when using an isometric joystick and indicate that the isometric nature of the device to be an important factor in rate scrolling.

Arrow and page up/page down keys: Most keyboards have specialized keys for scrolling. The up and down arrow keys move the cursor up and down one line with a document. When the edge of the screen is reached, the document is scrolled. The Page Up and Page Down keys scroll several lines at a time. The actual amount the Page Up and Page Down keys scroll is specified by the application, but typically the document is scrolled some significant portion of the screen height. Though arrows keys are useful in read mode, they provide only relative control and so are very limited for use in scan mode. With the exception of a specialized function key such as Home, they provide no method for moving quickly to an absolute position in the document.

Touchpad: Some laptops with touchpads can be used for controlling the scrollbar. Sliding the finger along the right edge of the touchpad provides relative control of the scrollbar. This supports read mode well. It is limited in scan mode by its relative nature, much as a mousewheel is, though constantly repositioning one's finger on a touchpad may be less fatiguing that working a mousewheel over and over. It is possible to support rate scrolling with this input device, but as mentioned above, lack of isometric feedback is an issue.

Previous work

Proprioception is the ability of people to sense the position and movement of their bodies, and has been studied for several decades. It has been shown that humans exhibit a remarkable ability to remember the position of their limbs [2]. The SlideBar's absolute linear mapping of document position should benefit from these effects.

The use of haptic feedback in computer input devices is also a well-studied area. Passive haptic feedback refers to a mechanism where the user can control and feel the position of the input device but where there is no response or control from the computer, in contrast to force or active feedback. Note that this terminology differs from that used in psychology where passive positioning means that ones body was moved externally by an apparatus and active refers to positioning done by voluntary muscle control. Fitzmaurice et al. [8,9] showed the advantages of having more specialized, graspable input devices that operate via passive haptic feedback. Studies [13,14,16] have also shown the importance of placing constraints on object movement for passive haptic feedback. The SlideBar should benefit from this as well since it is a constrained input device, physically moving only within the range of the task.

User studies dealing with comparative analysis of input devices [11,12,17] have shown that given the chance to use both hands, people will naturally use the non-dominant hand for rough positioning and the dominant hand for fine positioning. To study two-handed input, Buxton and Myers used linear input devices in participants' non-dominant hand [3]. For a scaling and positioning task, a physical linear slider with relative control for the scaling portion of the task was used. In a scrolling task, two touchpads were used for linear control. One gave absolute position control and the other gave relative control of the scrolling task. This gave users a different linear device corresponding to the different scrolling modes of scanning and reading. The SlideBar differs from these input devices in that only the SlideBar provides both absolute positioning and passive haptic feedback with physical constraints. Our purpose is to establish the benefit of the SlideBar, which is different from their purpose of studying the effect of two-handed input. We do expect two-handed input to be one of the SlideBar's advantages and their study establishes this effect.

The combination of two handed input, absolute positioning and constrained passive haptic feedback makes the SlideBar a unique computer input device for one degree of freedom tasks, which we believe it will excel at.

IMPLEMENTATION OF THE SLIDEBAR

The current implementation of the SlideBar is a proof of concept prototype that consists of a linear potentiometer that is wired to generate an analog voltage as the potentiometer is moved (figure 2). A graspable knob is attached to the potentiometer and it is placed to the side of the keyboard. The SlideBar has a range of motion of 4.5 cm, which is designed so that moving just the wrist and fingers without moving the forearm can access the full range. The analog voltage generated by the potentiometer is then digitized through a 12-bit A-D converter.

[Insert figure 2 here]

We wrote driver software to continuously read the SlideBar position via the serial port and generate events that are applied to scrolling. Position data for small changes is averaged, allowing for smoother control at slow scrolling rates, yet immediately applied for large position changes, maintaining the SlideBar's ability to move anywhere within a document without delay. The amount of averaging applied varies and is inversely proportional to the position change (up to a limit), so that the averaging used does not suddenly change at an arbitrary boundary. We also wrote custom applications that listen to SlideBar events and control scrolling within those applications.

The A-D converter limits the resolution of this version of the SlideBar to 4096 positions. As a comparison, scrollbars controlled by a mouse are limited in resolution to one pixel, determined by the resolution of the screen. Every time the scrollbar moves one pixel, the document scrolls a corresponding amount. For a high-resolution display of 1024 vertical pixels, a mouse controlled scrollbar provides one-fourth the resolution provided by the SlideBar prototype. With the same pixel resolution, a 40-page document and a 10-point font, the SlideBar prototype resolution is about one line of text.

However, the resolution of human movement may easily be less than that of the SlideBar. Numerous studies in joint proprioception show error to usually only a few degrees [2] and that the finger usually has higher error than other joints [1]. But studies of finger position are finger alone, not finger and thumb grasping an object and references hint at this potential cause of inferior finger performance [1,6]. None of these studies can be easily applied to calculate the error resolution of the SlideBar due to the various methods employed. These studies do show the error to be based on angle, not position, and here the mouse controlled scrollbar may regain some lost advantage. Because the distance of movement for a mouse is much greater than that of the prototype SlideBar the angles traversed during scrolling are greater. Rosenbaum et al. found that the optimal movement amplitude for the finger is about 45 degrees [15]. A study with positioning of sliders did show that errors in positioning are only a few percent for slider distances greater than about 5 to 10 centimeters, which is comparable to the full range of the prototype SlideBar [2]. But again methods are not directly applicable, in this case largely because most participant movements were very large and involved primarily elbow and shoulder movements. One interesting result was that participants tended to overestimate short ranges and underestimate longer ones. These findings will have some bearing on future SlideBar implementations.

While the prototype implementation of the SlideBar is a separate device that is placed next to the keyboard, a production version would be built into the keyboard. In addition to reducing the complexity and cost of the SlideBar, it will also improve its usability by providing stability to the device as well as putting in a consistent and accessible position. We expect it would require only a small amount of additional area on a keyboard, making it especially desirable for laptop computers where space is a premium and a mouse is often not available.

The prototype implementation of the SlideBar would be too expensive for commercial production. Instead, it is expected that a production implementation of the SlideBar would use similar technology to that used in the common mouse where a slotted wheel interrupts an infrared LED transmitter/receiver pair. A production version of the SlideBar could be implemented using the same approach but with a slotted linear stick that interrupts the infrared light. Alternatively, a slotted wheel similar to the ones used in mice could be used with a linear stick that turns the wheel. This would have the advantage that the turning rate could be geared up to increase resolution.

Some software design issues exist as well. We expect the SlideBar driver would automatically switch to control the vertical scrollbar of the focused window, however, other modes may be appropriate. The SlideBar will also likely be out of synch with a document when it moves into focus, as that document will not necessarily be scrolled to the depth corresponding to the SlideBar's current position. Our intuition is to have the document snap to the SlideBar's position as soon as that position starts to change, but there may be other ways to deal with this concern.

METHODS

Conditions and Materials

We conducted an experiment to evaluate the prototype SlideBar device on a laptop computer (700 MHz Pentium III, 128 MB ram) with a screen resolution of 1024 x 768 pixels. There were two independent variables: scrolling device; and document length. Three scrolling devices were tested. These were the prototype SlideBar, a standard mouse controlling a scrollbar, and a mousewheel. No acceleration algorithms were used with the mouse or mousewheel. The three documents lengths were 3 pages, 20 pages and 70 pages corresponding roughly to SlideBar resolutions of; less than one pixel, half a line and two lines, respectively. Dependent variables were the time to complete a scrolling task, the index of performance [4] for a scrolling task and subjective satisfaction. Participants were given three practice tasks, one with each device, prior to beginning the actual tasks.

Tasks

Participants performed two types of tasks. For each task, a document of random words formatted in paragraphs was presented to the participant. Approximately one-inch square, brightly coloured icons were placed randomly in the document as targets. Possible icons were circles, squares, rectangles, ovals and triangles. Colours and icons were chosen at random for each task. Prior to each task, a full screen page presented instructions for the upcoming task including what graphical icon to look for. A button to begin the task was placed near the center of the screen. The task document was then presented when the participant clicked the button, ensuring the starting position of the mouse was near the center of the screen. When the document was presented, it was scrolled all the way to the top and targets were never placed on the first page. The devices were presented in random order for each participant. During a trial for a specific device the other devices were physically present, but disabled.

The first type of task was the common scrolling task of scanning a document for a target in an unknown location. A single icon target was placed randomly in the document. We refer to this as the target location task. We wished to study the effects of device resolution, especially for the SlideBar, so we used a second independent variable, document length, with three values, leading to 9 total trials for this task. Participants were timed from the moment the start button was clicked until they found and clicked on the target.

The second task type used a 20-page document of random words, which contained two graphical icons placed within it. The location of the first icon was randomly placed somewhere in the first half of the document. The second icon was always a fixed 10 pages from the first. Participants were asked to find and select the first target then the second target, then the first again, repeatedly, for six clicks on each target. This is similar to Fitts' reciprocal tapping task [6]. Participants were timed from the beginning of the task to the selection of the first target and from each target selection to the next.

Subjects

There were a total of 24 participants in these experiments, eleven male and thirteen female. All but three were right handed, and everyone used the mouse in their dominant hand and SlideBar in the non-dominant hand. They were drawn from a general university population, ranging in age from 18-30. Most did not have a computer science background, but used computers every day.

Hypothesis

It was our hypothesis for the target location task that the SlideBar would outperform the standard mouse controlled scrollbar due to its advantages of two-handed input that eliminates the time necessary to acquire the scrollbar with the mouse. It was also expected that the SlideBar would outperform the mousewheel due to its nature as an absolute positioning device vs. the relative nature of the mousewheel. This effect may be minimized for the shorter document length.

It was our hypothesis for the reciprocal tapping task that the SlideBar would be faster than the other two devices. Only the SlideBar takes advantage of the haptic memory function of the human body and the task of repeatedly scrolling between previously located targets should take advantage of this effect.

Measures

In the target location task, time for completion was measured. The index of performance (IP) [4] was then calculated:

$$IP = \log_2(D/W + 1)/MT$$

where D is depth of target, W is width of target (fixed for all our trials) and MT is movement time. Because the target depths were varied at random for this task, it was not possible to use movement time directly as the performance measure. Index of performance was the best measure for normalizing the movement time based on target depth [10]. Three cases were discarded when participants attempted to use the incorrect device for the trial.

For reciprocal tapping tasks, 12 times were recorded. The first half of these was discarded to allow the participant to learn the location of the targets. Since the target distances and widths were fixed for every trial, movement time could be use directly as a measure of performance.

After the completion of the study tasks, participants completed a post-questionnaire where they were asked to rate each device for speed of use, clarity of use and satisfaction of use. They were also asked to choose a preferred device and given the opportunity to comment in detail.

RESULTS

Our results show that in performance the mousewheel was significantly better than the scrollbar. The SlideBar had a higher mean performance than the scrollbar but the effect was not significant. In subjective preference, the SlideBar offers significant benefits over the scrollbar. It also was preferred over the mouse wheel, but there was no statistically significant difference in performance between SlideBar and mouse wheel.

Locating a Target

For the target location task, univariate analysis of variance showed device to be a significant factor, F(2,165) = 7.139, $p = 0.003$. A post hoc Bonferroni comparison of the factors showed the mousewheel to perform significantly better than a scrollbar, $p = 0.002$, with mean IP difference of 0.3791 (figure 3). The SlideBar was not significantly better than a scrollbar, $p = 0.17$, nor was it significantly worse the mousewheel, $p = 0.333$.

[Insert figure 3 here]

The analysis also showed document length to be a significant factor for the target location task, F(2,165) = 17.59, $p < 0.001$. The interaction between device and document length was marginally significant at $p = 0.061$.

Reciprocal Tapping

For the reciprocal tapping task, univariate analysis of variance also showed device to be a significant factor, F(2,499) = 7.79, $p < 0.001$. A post hoc Bonferroni comparison of the factors again showed the SlideBar to be significantly faster than a scrollbar, $p < 0.001$, with mean times of 5.8 seconds for the SlideBar versus 8.7 seconds for the scrollbar (figure 4). Again, there was no significant difference in speed between the SlideBar and mousewheel.

[Insert figure 4 here.]

User Preference

Users preferred the SlideBar. Participants were asked to choose their preferred device and 14 of the 24 chose the SlideBar. Users also gave subjective rankings on a scale of 1 (lowest) to 9 (highest) for speed, clarity of use, and satisfaction. For speed, the SlideBar significantly outranked the other two devices, F(2,69) = 15.5, $p < 0.001$; on average the SlideBar ranked 8.09, the mousewheel ranked 5.57, and the mouse with scrollbar ranked 5.87. Users found no significant difference in clarity of use. Users were significantly more satisfied with the SlideBar than the mousewheel, F(2,69) = 3.28, $p < 0.044$; on average the SlideBar ranked 7.04, the mousewheel ranked 5.65, and the mouse with scrollbar ranked 6.26.

ANALYSIS

The results support our hypothesis that the SlideBar is faster than a standard mouse controlled scrollbar for the reciprocal tapping task. It does not support our hypothesis that the SlideBar is faster than a mousewheel for reciprocal tapping task. It does not support our hypothesis that the SlideBar would outperform the other two devices at the target location task.

The interaction between device and document length was marginally significant. For shorter documents the mousewheel starts to show an increasing gain in performance over the other devices (figure 3), while the scrollbar starts to show an increasing loss in performance. The decrease in performance for the scrollbar can be attributed to the fact that there is constant time required to acquire the scrollbar with the mouse, and this time becomes a larger percentage of the overall movement time for short documents. The increase in performance for the mousewheel is likely because, for short documents, it is often possible to scroll to the target in a single motion of the mousewheel, eliminating time spent re-positioning the finger on the mousewheel.

For the reciprocal tapping task, the scrollbar clearly under performed. This is very likely due to the overhead of acquiring the scrollbar with the mouse at each target switch. Neither the mousewheel nor SlideBar suffers from this drawback.

CONCLUSIONS

We designed the SlideBar to take advantage of the potential benefits of two-handed use and constrained passive haptic feedback to give it an advantage over traditional scrolling devices. We tested these devices with two common tasks; scrolling a document while scanning for a target at an unknown location, and switching between two targets an known locations.

This study showed that the SlideBar was significantly faster for a reciprocal tapping task than a conventional mouse controlled scrollbar. Even though not significant, the SlideBar showed faster times than a mousewheel for a reciprocal tapping task. This weakly supports the advantage of constrained passive haptic feedback.

All devices showed roughly the same index of performance, except in the case of the short document, which clearly starts to favor the mousewheel over the scrollbar. The SlideBar tends to show improving performance relative to the other two devices for longer documents, but the effect is not significant and the structure of this study is not designed for determining this effect.

More work will be necessary to improve the SlideBar and to better understand what tasks it is best suited for. It would also be useful to determine what parameters are necessary to improve its use, such as range of motion and force of motion required. Adding features for control of relative position, such as touch sliders or some isometric rate device, may also enhance its ability as an input device. Studies would be needed to define and optimize these benefits.

We expect that scrolling will continue to be such a common task performed by computer users, that the expense of adding a linear input device such as the SlideBar is well worth the small cost increase required to include it in production systems. The improvement in performance and preference indicates that users would benefit from its introduction. There are also a number of other linear tasks for which the SlideBar could be the ideal input device, such as controlling zoomable interfaces, and controlling on screen widgets, especially for potentially sensitive tasks like dynamic queries.

REFERENCES

1. Balakrishnan, R. and MacKenzie, IS., 1997, Performance Differences in the Fingers, Wrist, and Forearm in Computer Input Control. Proceedings of CHI'97, pp. 303-310.

2. Boff, K.R., Kaufman, L. and Thomas, J.P., 1986, The Handbook of Perception and Human Performance, Volume I. Wiley, New York, chapter 13.

3. Buxton, W. and Myers, B.A., 1986, A Study in Two-Handed Input, CHI'86 Proceedings, pp. 321-326.

4. Card, S.K., Moran, T.P., Newell, A., 1983, The Psychology of Human-Computer Interaction. Hillsdale, New Jersey, Lawrence Erlbaum Associates.

5. Douglas, Sarah A. and Mithal, Anant Kartik, 1994, The Effect of Reducing Homing Time on the Speed of a Finger-Controlled Isometric Pointing Device, Proceedings of CHI'94 Human Factors in Computing Systems, pp. 411-416.

6. Fitts, P.M., 1954, The information capacity of human motor system in controlling the amplitude of movement. Journal of Experiemental Psychology, **47**, 381-391.

7. Fitts, P. and Peterson, J., 1964, Information Capacity of Discrete Motor Responses. Journal of Experimental Psychology, **67**, 103-112.

8. Fitzmaurice, G.W., Ishii, H. and Buxton, W., 1995, Bricks: Laying the Foundations for Graspable User Interfaces. Proceedings of CHI'95, pp. 432-449.

9. Fitzmaurice, G.W. and Buxton, W., 1997, An Empirical Evaluation of Graspable User Interfaces: towards specialized, space multiplexed input. Proceedings of CHI'97, pp. 43-50.

10. Hinckley, K., Cutrell, E., Bathiche, S. and Muss, T., 2002, Quantitative Analysis of Scrolling Techniques. Proceedings of CHI'02 Human Factors in Computing Systems, pp. 65-72.

11. Kabbash, P., MacKenzie I.S. and Buxton W., 1993, Human Performance Using Computer Input Devices in the Preferred and Non-Preferred Hands. Proceedings of CHI'93 Human Factors in Computing Systems, pp. 474-481.

12. Kabbash, P., Buxton W. and Sellen A., 1994, Two-handed Input in a Compound Task. Proceedings of CHI'94 Human Factors in Computing Systems, pp. 417-423.

13. Lindeman, R.W., Sibert, J.L. and Hahn, J.K., 1999, Towards usable VR: An empirical study of user interfaces for immersive virtual environments. Proceedings of CHI'99, pp. 64-71.

14. MacKenzie, C.L., 1992, Making contact: Target surfaces and pointing implements for 3D kinematics of humans performing a Fitts' task. Society for Neuroscience Abstracts, **18**, 515.

15. Rosenbaum, D.A., Slotta, J.D., Vaughan, J. and Plamondon, R., 1991, Optimal movement selection. Psychological Science, **2**, 86-91.

16. Wang, Y. and MacKenzie, C.L., 2000, The Role of Contextual Haptic and Visual Constraints on Object Manipulation in Virtual Environments. Proceedings of CHI'00, pp. 532-539.

17. Zhai, S., Smith, B.A. and Selker, T., 1997, Improving Browsing Performance: A study of four input devices for scrolling and pointing tasks. Proceedings of Interact'97, pp. 286-292.

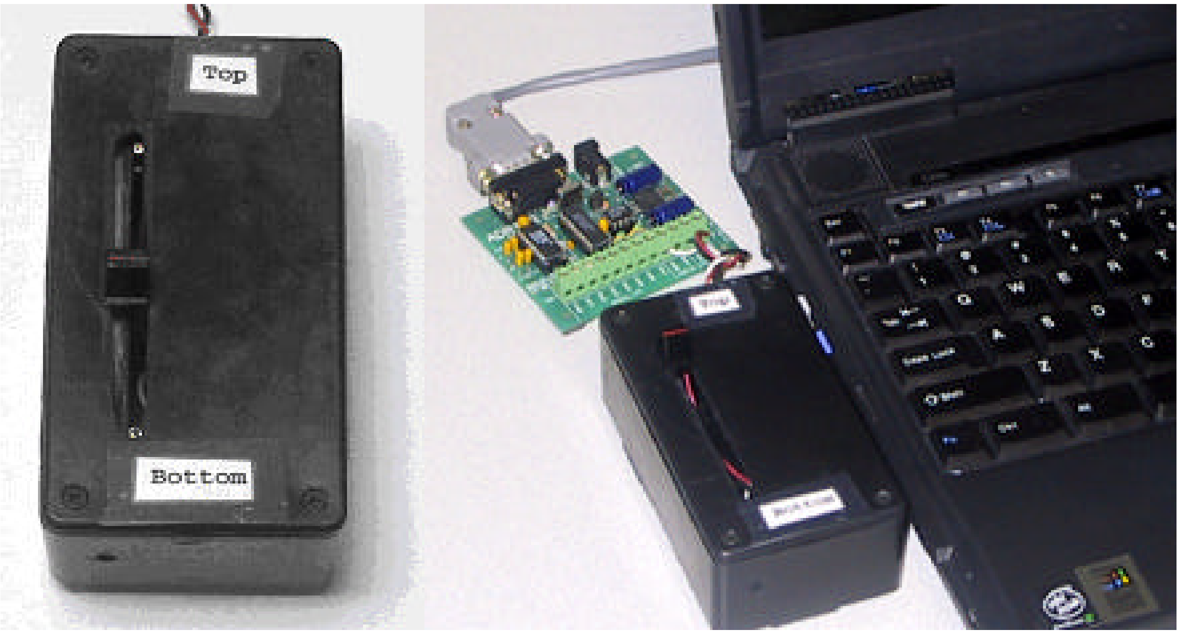

Figure 1: The Prototype SlideBar.

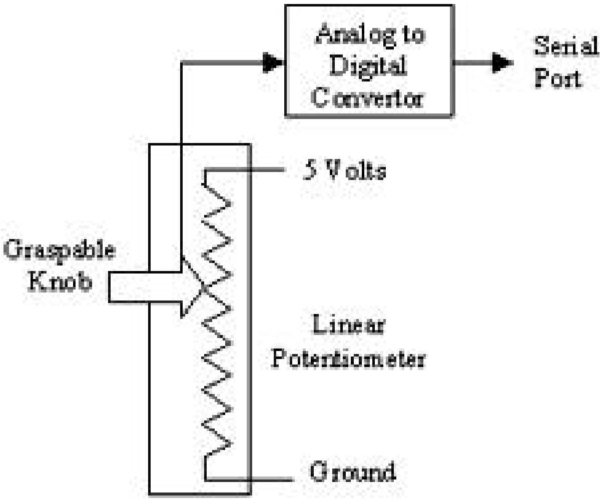

Figure 2. Block diagram of the SlideBar.

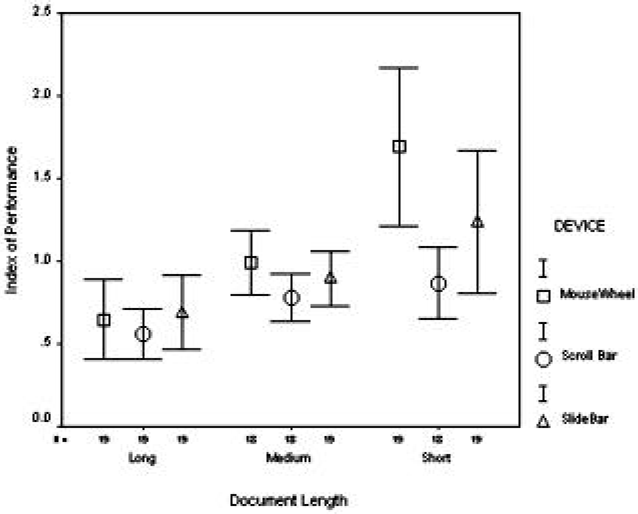

Figure 3. Mean index of performance locating a target with bars representing 95% confidence interval.

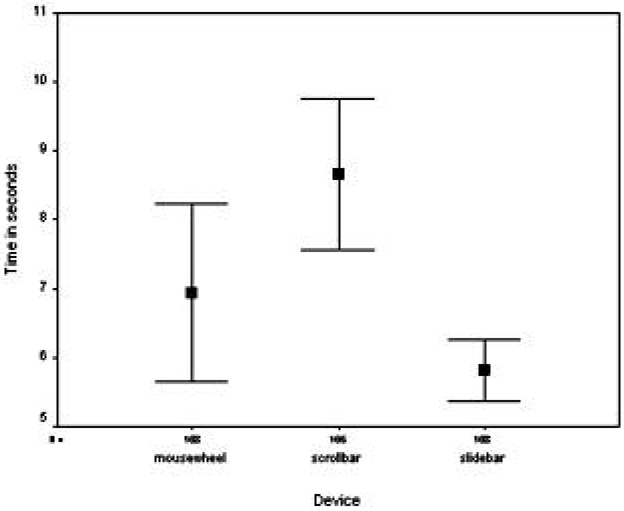

Figure 4. Mean time to switch between known targets with bars representing 95% confidence interval.