BEHAVIOUR & INFORMATION TECHNOLOGY, JANUARY–FEBRUARY 2004, VOL. 23, NO. 1, 1–9

SlideBar: Analysis of a linear input device

LESLIE E. CHIPMAN, BENJAMIN B. BEDERSON and JENNIFER A. GOLBECK

Department of Computer Science, Human-Computer Interaction Laboratory, University of Maryland, College Park, Maryland, USA; e-mail: {gchipman, bederson, golbeck}@cs.umd.edu

Abstract. The SlideBar is a physical linear input device for absolute position control of 1° 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 user preference. In addition, users tended to prefer it over the mousewheel (without statistical significance).

1. 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.

In traditional graphical user interfaces (GUIs), 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. Scrolling behaviour is an extremely frequent task in GUIs, and it becomes even more frequent in browsing World Wide Web pages.

Scrolling is just 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 goal of providing users with an input device better suited for linear tasks. In this paper, 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 laptop and even smaller computers where the limitations of a mouse become more pronounced. We believe the SlideBar

Behaviour & Information Technology
ISSN 0144-929X print/ISSN 1362-3001 online © 2004 Taylor & Francis Ltd
http://www.tandf.co.uk/journals
DOI: 10.1080/01449290310001638487

offers a large enough advantage to justify being built in to some computers.

1.1. *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.

1.2. *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.

1.2.1. *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 appears to be 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, and we all use it regularly. However, scrollbars do have several shortcomings. The primary one results from the fact that users 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 s for users to acquire a mouse and 0.44 s for a keyboard joystick (Dougland *et al*. 1994). 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 15 pixels wide, acquiring the scrollbar takes up to two seconds (Zhai *et al*. 1997), and impacts performance, in accordance with Fitts' Law (Fitts and Peterson 1954). 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 several common goals 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.

1.2.2. *The mousewheel*: A mousewheel is commonly built into 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, it 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*. (1997) 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.

1.2.3. *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 specialized keys such as the 'Home' key, they provide no method for moving quickly to an absolute position in the document.

1.2.4. *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.

1.3. *The slidebar*

The SlideBar is a physical linear input device that can be used to control absolute position of a 1° 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.

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.

1.4. *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 (Boff *et al*. 1986). 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

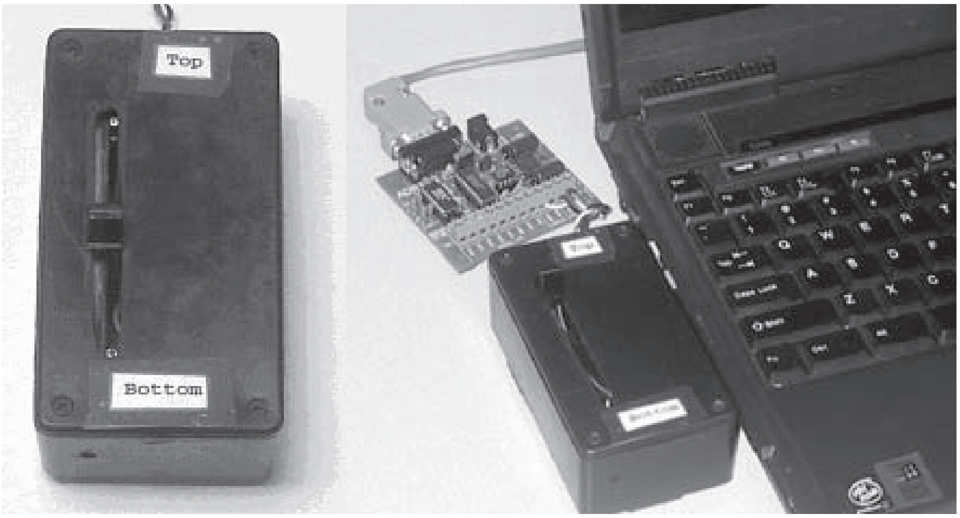

Figure 1. The prototype slidebar.

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*. (1995) and Fitzmaurice and Buxton (1997) showed the advantages of having more specialized, graspable input devices that operate via passive haptic feedback. Studies (MacKenzie 1992, Lindeman *et al*. 1999, Wang and MacKenzie 2000) 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 (Kabbash *et al*. 1993, 1994, Zhai *et al*. 1997) 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 (1986) used linear input devices in participants' non-dominant hand. 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 1° of freedom tasks, which we believe it will excel at.

2. 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. 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.

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 (Boff *et al*. 1986) and that the finger usually has higher error than other joints (Balakrishnan and MacKenzie 1997). 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 (Fitts 1954, Balkrishnan and MacKenzie 1997). 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*. (1991) found that the optimal movement amplitude for the finger is about 45° Rosenbaum *et al*. 1991). 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 cm, which is comparable to the full range of the prototype SlideBar (Boff *et al*. 1986). 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, we envision a production version that would be built into the keyboard. In addition to reducing the complexity and cost of the SlideBar, this would also improve its usability by providing stability to the device as well as putting it 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.

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 sync 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.

3. Methods

3.1. *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 1024×768 pixels. There were two independent variables: scrolling device; and document length. Three scrolling devices were tested: 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 three 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 for a scrolling task and subjective satisfaction. The study was set up as a within-subjects design, so each subject performed both tasks with all three devices.

Participants were first presented with a description of each device and how it worked. The study administrator demonstrated the devices and tasks on a practice document. Participants were instructed on how each exercise was to be completed, and then given unlimited time to practice device use and familiarize themselves with the tasks until they felt comfortable enough to proceed. On both the mousewheel and SlideBar, the two least familiar devices, subjects on average spent over a minute practicing, with several subjects spending several minutes. On the scrollbar, a device familiar to all of the subjects, average practice time was still nearly 45 seconds, ensuring that the use of the device in context of the tasks was familiar.

Subjects were administered a pre-test to gather demographic information and data about their previous experience with computers and various input devices. For the actual experiment, users completed all the tasks for a given device before proceeding to the next device. The order of the devices was randomized for each subject.

After completing the tasks for the last device, users completed a post-test questionnaire that gathered their preferences and comments about the devices. They were also interviewed to gather any additional opinions about the experiment. The entire process took approximately 1 h.

3.2. *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 centre of the screen. The task document was presented when the participant clicked the button, ensuring the starting position of the mouse was near the centre 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 nine total trials for this task. Participants were automatically timed from the moment the start button was clicked until they found and clicked on the target.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.