Navigating Documents with the Virtual Scroll Ring

Tomer Moscovich, John F. Hughes
Department of Computer Science
Brown University, Box 1910
Providence, RI 02912, USA
tm@cs.brown.edu, jfh@cs.brown.edu

ABSTRACT

We present a technique for scrolling through documents that is simple to implement and requires no special hardware. This is accomplished by simulating a hardware scroll ring—a device that maps circular finger motion into vertical scrolling. The technique performs at least as well as a mouse wheel for medium and long distances, and is preferred by users. It can be particularly useful in portable devices where screen-space and space for peripherals is at a premium.

Categories and Subject Descriptors: H.5.2 [**User Interfaces**]: Graphical User Interfaces (GUI), Input devices and strategies

Additional Keywords and Phrases: Interaction Techniques, Scrolling

INTRODUCTION

Computer users spend a large amount of their time reading and editing documents. Since documents are generally too long to fit on one screen, users must frequently scroll to other parts of the document. This common, repetitive task has inspired the creation of a variety of hardware devices aimed at improving the scrolling experience [8]. But the world is full of computers lacking this special hardware, laptop users may not want to carry an additional wheel mouse, and not every hand-held device has room for a dedicated scrolling mechanism. We present a scrolling technique that uses an existing general positioning device such as a touchpad, stylus, or standard mouse.

The technique is a software simulation of a hardware device known as a *scroll ring.* The scroll ring maps circular motion of the user's finger into vertical scrolling motion. A study by Wherry [6] shows that ring scrolling can be faster than using a mouse wheel. Furthermore, users prefer the continuous motion and precise control afforded by the scroll ring. Do these benefits persist without the limited tactile feedback of a physical ring? To find out we created the *virtual scroll ring (VSR).*

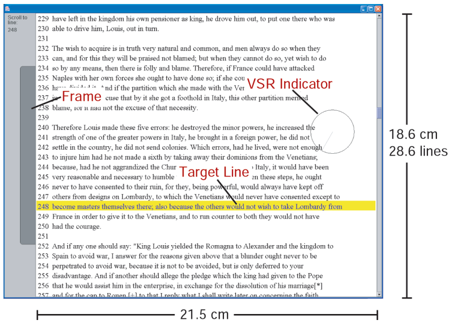
To use the virtual scroll ring the user activates scrolling mode and makes continuous circular movements with the pointing device. Clockwise motion scrolls the view down, while counterclockwise motion scrolls up. The scrolling activation method depends on the pointing device and the scrolling context. For example, a user can activate touchpad scrolling by tapping a "hot-spot" on the pad. Alternatively, a single mouse button can be multiplexed using techniques in the style of Zeleznik's UniCam controls [7]. For example, clicking and moving would begin scrolling, while clicking and hovering briefly would bring up a context menu.

The use of circular motion for adjusting a continuous scalar value is not new. Newman proposed the Light Handle [5] as an alternative to a shaft encoder. Guimbretière et al.'s FlowMenus [3] adjust zoom level relative to the angle traversed around the menu's origin. These techniques are based on absolute positioning relative to a fixed point, and so require more visual attention from the user. Our technique uses position relative to an adaptive center of revolution, leaving the user free to look at the document rather than at an obstructing widget. This aspect is similar to Evans et al.'s stirrer [2], which rotates an object by a function of the the pointer trajectory's curvature. Our implementation can use many more sample points than Evans without introducing a lag. This makes it less susceptible to noise or aliasing in the input.

Unlike previous systems, which vary parameters based on the angle swept around a point, the VSR uses the distance traveled along the circumference of a circle. Scrolling by angle is a contrary mapping for the task, as slow, small, circles would cause fast scrolling, while fast, large, circles would cause slow scrolling. We believe that for repetitive circular motion amplitude and frequency provide a more easily grasped conceptual model than the traditional angle and radius model. With the VSR, large or fast movements produce fast scrolling, while small or slow movements yield slow scrolling. By making the scrolling rate directly proportional to the speed of the pointer we allows the user to choose a size for the circle that is most comfortable for the current scrolling speed. This may be an improvement over the physical scroll ring, as it is often easier to increase a motion's amplitude than its frequency.

One advantage of the VSR over the mouse wheel is the continuous nature of the motion; it does not require the user to release and re-engage the input device. It also allows for more precise control over the scrolling speed, an important asset for scanning documents. By adjusting the radius of the circle while keeping the frequency of the motion constant, the user can span a wide continuum of scrolling rates. Con-

UIST '04, October 24–27, 2004, Santa Fe, New Mexico, USA.



229 have left in the kingdom his own pensioner as king, he drove him out, to put one there who was
230 able to drive him, Louis, out in turn.
231
232 The wish to acquire is in truth very natural and common, and men always do so when they
233 can, and for this they will be praised not blamed; but when they cannot do so, yet wish to do
234 so by any means, then there is folly and blame. Therefore, if France could have attacked
235 Naples with her own forces she ought to have done so; if she cou
236 ha nd if the partition which she made with the Ver
237 cuse that by it she got a foothold in Italy, this other partition merited
238 blame, for it had not the excuse of that necessity.
239
240 Therefore Louis made these five errors: he destroyed the minor powers, he increased the
241 strength of one of the greater powers in Italy, he brought in a foreign power, he did not
242 settle in the country, he did not send colonies. Which errors, had he lived, were not enough
243 to injure him had he not made a sixth by taking away their dominions from the Venetians;
244 because, had he not aggrandized the Chur Italy, it would have been
245 very reasonable and necessary to humble en these steps, he ought
246 never to have consented to their ruin, for they, being powerful, would always have kept off
247 others from designs on Lombardy, to which the Venetians would never have consented except to
248 become masters themselves there; also because the others would not wish to take Lombardy from
249 France in order to give it to the Venetians, and to run counter to both they would not have
250 had the courage.
251
252 And if any one should say: "King Louis yielded the Romagna to Alexander and the kingdom to
253 Spain to avoid war, I answer for the reasons given above that a blunder ought never to be
254 perpetrated to avoid war, because it is not to be avoided, but is only deferred to your
255 disadvantage. And if another should allege the pledge which the king had given to the Pope
256 that he would assist him in the enterprise, in exchange for the dissolution of his marriage[*]

Figure 1: A screen-shot of the setup used in our study. The subjects' task was to scroll the target line so it lies within the marked frame.

versely, an advantage of the mouse wheel over the VSR is the haptic feedback provided by the notched wheel. The clicking of the notches help the user determine the distance scrolled, and snap the motion to integer lines. With the VSR the user is limited to visual and kinesthetic feedback.

The VSR also has several advantages over the most common software scrolling technique, the scrollbar. While scrollbars can rapidly bring a user to any point in the document, they lose precision in very long documents. More importantly, they interrupt the user's workflow: the user must look away from the document to guide the pointer to the scrollbar "thumb." Acquiring a small screen widget like the scrollthumb may take as long as two seconds [8].

Another hardware independent scrolling technique requires the user to click on a point in the document, and then move the pointer in the desired scrolling direction. The distance from the pointer to the original point controls the scrolling rate. An advantage of this method is that it allows scrolling in two dimensions simultaneously (useful for maps and large graphics). Unfortunately, this method can be confusing for rapid scrolling, as it provides no kinesthetic feedback of the scrolling rate or the distance covered.

SIMULATING A SCROLL RING

There are several properties we want in a virtual scroll ring. Since it lacks tactile feedback, it should not constrain the motion to a specific circle; it should accommodate the user's preferred motion amplitude as well as any drift in the location of the circle. It should also exhibit an immediate response to changes in scrolling speed and direction. To accomplish this, our system infers the circle the user is drawing from the most recent segment of the pointer trajectory. It then scrolls the document by the length of the arc traversed on the circle since the last scroll event.

We find a linear least-squares fit for the circle with a technique from Arvo et al. [1], which we briefly describe here. Let $\mathbf{X}$ and $\mathbf{Y}$ denote column vectors of the last n mouse positions, and $\mathbf{1}$ a column vector of ones; let $\mathbf{X}^2$ denote the termwise square of $\mathbf{X}$. To find the circle, we solve the overdetermined system

$$\left[\begin{array}{ccc} 2\mathbf{X} & 2\mathbf{Y} & \mathbf{1} \end{array}\right]\left[\begin{array}{ccc} c_1 & c_2 & c_3 \end{array}\right]^T = \left[\begin{array}{c} \mathbf{X}^2+\mathbf{Y}^2 \end{array}\right]$$

The circle's center is then (c_1, c_2), and its radius is $r = \sqrt{c_3+c_1^2+c_2^2}$. Given the angle θ swept since the last scroll event, the distance to scroll is simply $2\theta r$. The sign of θ determines the scrolling direction.

This method instantly responds to changes in scrolling speed and direction. However, it does take a fraction of a second to accommodate sudden changes in radius or position. The length of this delay depends on n and on the rate at which the system processes scrolling events. A shorter history makes for snappier adaptation, but increases sensitivity to pixel grid aliasing errors which may become noticeable at low scrolling rates. Our system used $n = 30$, and handled 55 events per second on a 1.6 GHz Pentium M Processor.

Using this mapping the scrolling speed is approximately twice the speed of the pointer (the error becomes smaller as the user's motion approaches an ideal circle). It is possible to apply various multiplicative factors or gain functions to either r or θ to adjust scrolling behavior independently from the mouse setting. In practice we found that a factor of two yields a wide enough range of scrolling speeds.

EVALUATION

Experiment One

Procedure and Design. We conducted a limited user study comparing VSR scrolling characteristics to those of a well studied standard—the mouse wheel. The VSR may be controlled with a wide variety of positioning devices; our study was limited to control using a mouse and using a touchpad. We compared VSR and mouse wheel performance at short, medium, and long range scrolling tasks, and at two levels of precision.

Closely following Hinckley et al. [4] we evaluate our technique using a reciprocal framing task. Subjects alternate scrolling to one of two marked lines, positioning each line so that it lies within the boundaries of a marked vertical frame. When the subject reaches the specified line she presses the space-bar and continues to the next line. If the specified line lies outside the frame the system sounds a bell, and the subject continues to the next line. This setup is shown in Figure 1.

The experiment design crossed device × scrolling-distance × frame height. The devices tested were the VSR using a mouse, the VSR using a touchpad, and the mouse scroll wheel. We chose three representative scrolling distances: 6, 24, and 192 lines, and two frame heights: 6 and 18 lines. Note that when targets are 6 lines apart no scrolling is needed for an 18 line frame, so the factors were not fully crossed. Subjects completed nine sequential repetitions for each condition. Condition order was balanced with respect to device, and condition order was randomized within each device.

Ten right-handed subjects participated in the experiment. Each was allowed a short practice session to become familiar with the three scrolling techniques. Once comfortable with the task and techniques, they completed the timed trials in three blocks corresponding to scrolling device. Subjects ended the session by completing a short survey.

Setup. We performed the experiment on an IBM Thinkpad equipped with a 6×4 cm touchpad and a Microsoft Wheel

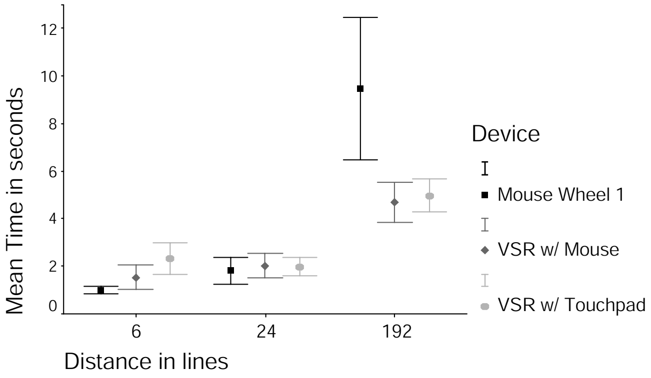


Figure 2: Experiment One – Mean completion times and 95% confidence intervals for the scrolling task. Mouse wheel set at one line per notch.

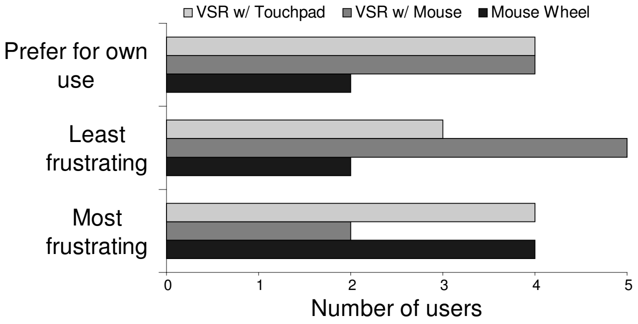


Figure 3: Experiment One – Subjective ranking for each device. Ten users indicated which device they found most and least frustrating to use, and which they would choose for their own use. Mouse wheel set at one lines per notch.

Mouse Optical. We disabled software acceleration for the mouse and scroll wheel, but were unable to prevent the touchpad's low-speed gain control that increases precision for slow movements. The mouse wheel was set to scroll at one line per notch. Clicking either the left or right button activated the VSR.

The visible portion of the document was 18.6 cm by 21.5 cm wide. Each of the 28.6 lines visible was 0.65 cm (32 pixels) high. The document was 650 lines of text.

Results. An analysis of variance of the fully crossed subset of the data (two distances × two frame heights) revealed a significant main effect for device ($F_{1.09,9.84} = 14.68, p < 0.05, \hat{\varepsilon} = 0.55$, with a Greenhouse-Geisser correction for sphericity violations) as well as a significant interaction between device and distance ($F_{1.02,9.15} = 21.60, p < 0.05, \hat{\varepsilon} = 0.51$). This interaction can be seen in Figure 2. For medium scrolling distances no significant difference was found between either VSR type and the mouse wheel. For long distance scrolling both VSR types outperformed the mouse wheel (mouse VSR $t_9 = 4.41, p < 0.006$, touchpad VSR $t_9 = 4.12, p < 0.006$ with a Bonferonni correction for nine comparisons). This was mirrored in subjects' perception. All agreed that the mouse wheel was slowest; one wrote in the survey "Scrolling long distances was a chore." An acceleration gain as described by Hinckley et al. [4] would likely bring long-distance mouse wheel performance closer to that of the VSR. For short distances most users preferred the mouse wheel, saying it was "More accurate, stopping where I wanted it to." Several mentioned that they found the tactile feedback very helpful. Indeed, for short distances mean scrolling time with the mouse wheel was 1.3 seconds shorter than using the touchpad VSR ($t_9 = 4.95, p < 0.006$).

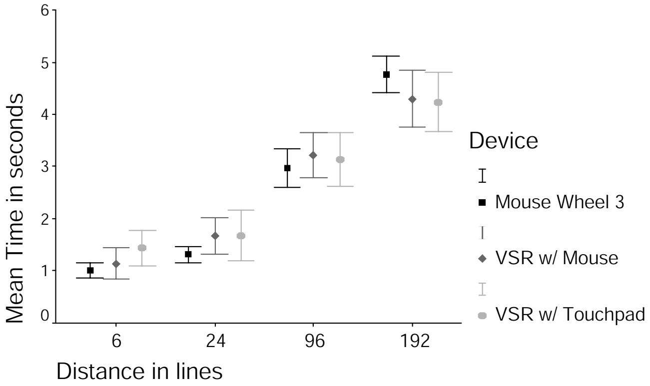


Figure 4: Experiment Two – Mean completion times and 95% confidence intervals for the scrolling task. Mouse wheel set at three lines per notch.

While we found no significant difference between the VSR with a mouse and the mouse wheel for short distances, we did find a difference for the VSR with touchpad. We attribute this behavior to the touchpad's low-speed gain setting. At low stroke speeds this setting causes the pointer to move very slowly. When the recent pointer trajectory lies in a region that is small relative to the pixel grid our system cannot determine a robust estimate for the traversed circle. This condition may be further aggravated by the confused user's attempt to compensate. Despite the touchpad's poor performance at very short distances more users preferred it to the mouse wheel (see Figure 3), saying "it was intuitive and fast." Frustration with the scrolling devices varied greatly between subjects. This appears to be a personal preference that is not associated with previous mouse wheel or touchpad use.

While our analysis did reveal a significant main effect for frame height, no significant interaction was found between device and frame height.

Experiment Two

Procedure and Setup. Scrolling long distances with the mouse wheel set at one line per notch is slow, and some users may be willing to give up a small amount of precision for a gain in scrolling speed. To see how the VSR compares to a faster scroll wheel, and how it affects users' perception of the devices, we re-ran our first experiment with the mouse wheel set at three lines per notch. We also added a fourth scrolling distance; users scrolled 6, 24, 96, and 192 lines. Eight right-handed subjects participated in the experiment. All other factors remained the same as in the first experiment.

Results. As can be seen in Figure 4, VSR performance was comparable that of the mouse wheel set at three lines per notch. While our experiment could not distinguish a signifi-

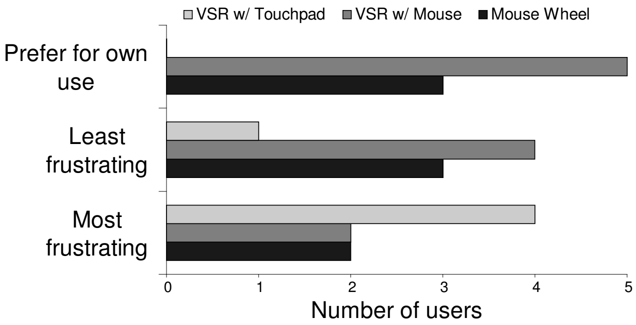


Figure 5: Experiment Two – Subjective ranking for each device. Mouse wheel set at three lines per notch.

cant effect for device, we did find a significant interaction between device and distance ($F_{4,24} = 7.06, p < 0.05$). However, multiple comparisons revealed no significant differences between devices at each distance.

Although the scrolling time was the same, more subjects preferred the VSR with mouse to the other techniques (see Figure 5). Increased scrolling rate did reduce subjects' frustration with the mouse wheel, but subjects still found the wheel to be tedious when scrolling long distance. This is in contrast to the VSR, which subjects found "more fun and entertaining" to use. One said that "for long distances circular motion [is] sort of relaxing." Several asked if the VSR was available anywhere. As in experiment one, subjects did find the touchpad VSR frustrating to use for very short distances.

DISCUSSION

The virtual scroll ring is a tenable scrolling alternative. This is especially true when most scrolling actions are expected to be longer than half a page. For shorter distances involving slow movement extra care should be taken to ensure that enough data is collected for a robust estimate of the circle. This may be done by disabling low-speed gain during scrolling, or increasing the length of the the trajectory history at low speeds.

Since the VSR scrolls the view smoothly, in increments as small as one pixel, it allows users to read the text while they scroll. However, for some applications users may prefer to have the view snap to integer lines. This is easily implemented by scrolling the text only when the pointer travels a distance equal to the line height.

The VSR may also work well in conjunction with other scrolling techniques. For example, users may turn the mouse wheel to scroll short distances and depress the wheel (which doubles as a button) to activate the VSR for scrolling long distances. The VSR could also be used for one-dimensional dragging tasks (such as using a slider). Rather than repetitively re-clutching the pointing device, the user may begin with a linear sliding motion, and continue with a circular motion, essentially converting a slider into a dial.

REFERENCES

1. James Arvo and Kevin Novins. Fluid sketches: Continuous recognition and morphing of simple hand-drawn shapes. In *Proceedings of the 13th annual ACM symposium on User interface software and technology*, pages 73–80. ACM Press, 2000.

2. Kenneth B. Evans, Peter P. Tanner, and Marceli Wein. Tablet-based valuators that provide one, two, or three degrees of freedom. *SIGGRAPH*, 15(3):91–97, 1981.

3. François Guimbretière and Terry Winograd. Flowmenu: Combining command, text, and data entry. In *Proceedings of the 13th annual ACM Symposium on User Interface software and technology*, pages 213–216. ACM Press, 2000.

4. Ken Hinckley, Edward Cutrell, Steve Bathiche, and Tim Muss. Quantitative analysis of scrolling techniques. In *Proceedings of the ACM CHI conference on Human factors in computing systems*, pages 65–72. ACM Press, 2002.

5. W.M. Newman. A graphical technique for numerical input. *Computer Journal*, 11(1):63–64, 1968.

6. Elaine Wherry. Scroll ring performance evaluation. In *Proceedings of the ACM CHI extended abstracts on Human factors in computing systems*, pages 758–759. ACM Press, 2003.

7. Robert Zeleznik and Andrew Forsberg. Unicam – 2D gestural camera controls for 3d environments. In *Proceedings of the 1999 symposium on Interactive 3D graphics*, pages 169–173. ACM Press, 1999.

8. Shumin Zhai, Barton A. Smith, and Ted Selker. Improving browsing performance: A study of four input devices for scrolling and pointing tasks. In *Proceedings of INTERACT'97*, pages 286–292, 1997.