

Tuning and Testing Scrolling Interfaces that Automatically Zoom

Andy Cockburn[†], Joshua Savage[‡], Andrew Wallace[†]

[†]Human-Computer Interaction Lab
Department of Computer Science and Software Engineering
University of Canterbury, Christchurch
New Zealand
andy@cosc.canterbury.ac.nz

[‡]LeftClick Ltd.
Canterbury Innovation Incubator
Christchurch
New Zealand
josh.savage@leftclick.co.nz

ABSTRACT

Speed dependent automatic zooming (SDAZ) is a promising refinement to scrolling in which documents are automatically zoomed-out as the scroll rate increases. By automatically zooming, the visual flow rate is reduced enabling rapid scrolling without motion blur. In order to aid SDAZ calibration we theoretically and empirically scrutinise human factors of the speed/zoom relationship. We then compare user performance with four alternative text-document scrolling systems, two of which employ automatic zooming. One of these systems, which we term 'DDAZ', is based on van Wijk and Nuij's recent and important theory that calculates optimal pan/zoom paths between known locations in 2D space. van Wijk and Nuij suggested that their theory could be applied to scrolling, but did not implement or test their formulaic suggestions. Participants in our evaluation (n=27) completed scrolling tasks most rapidly when using SDAZ, followed by DDAZ, normal scrollbars, and traditional rate-based scrolling. Workload assessments and preferences strongly favoured SDAZ. We finish by examining issues for consideration in commercial deployments.

Categories & Subject Descriptors: H5.2. [Information Interfaces and Presentation]: User Interfaces—Input devices and strategies.

General Terms: Human Factors; Experimentation.

Keywords: Scrolling, automatic zooming, visual flow, rate control

INTRODUCTION

Scrolling and panning allow users to move through information spaces that are too large for convenient display within a single window. Byrne et al. [4] observed that web users spend a surprisingly large proportion of their time

(13%) manipulating scrollbars, and they commented that scrolling was "an obvious case where widget design could make a difference". Since this study, several scrolling enhancements have been deployed in mainstream desktop environments. Examples include scroll-wheel mice and rate-based scrolling in which scroll-speed is a function of the distance dragged with the middle mouse button.

This paper scrutinises a promising scrolling refinement called 'speed-dependent automatic zooming' or SDAZ. SDAZ automatically zooms away from the document as the scrolling velocity increases: the faster you scroll, the 'higher' you fly (see Figure 1). Although SDAZ was first formally described by Igarashi and Hinckley in 2000 [11] a similar concept was used in the computer game 'Grand Theft Auto' in 1997. The game gave users a plan view of their car in a city street, which automatically zoomed to show progressively more city blocks on acceleration. The need for zooming in the game is clear: without it, the rate of display change (the speed of pixel movement) exceeds that the human visual system can process, inducing 'motion blur'. Zooming-out decreases the rate of pixel movement, allowing higher speeds in the information space without overloading the visual system. Standard desktop systems suffer similar problems: when seeking a visual target users wish to find it quickly, but the faster they scroll the greater the impact of motion-blur. Contemporary software developers are grappling with the close relationship between scrolling, panning and zooming as demonstrated by systems such as Adobe Reader 6 which includes a 'Dynamic Zoom' tool that allows a single mouse to control simultaneous panning and zooming.

Igarashi and Hinckley's preliminary evaluation of SDAZ (n=7) found that it allowed comparable performance to other scrolling techniques. In our previous work (n=12) we found positive results for SDAZ, both in user performance and in subjective preferences, when compared to document and map navigation using traditional scrollbars[6]. More recently, van Wijk and Nuij [19] proposed an adaptation of SDAZ based on a strong theory that yields optimal animations in pan/zoom space. Their formulae calculate a smooth and efficient camera flight between two x,y coordinates, each at different zoom levels. Although the theory is primarily intended for pre-determined starting and

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHI 2005, April 2–7, 2005, Portland, Oregon, USA.

Copyright 2005 ACM 1-58113-998-5/05/0004...\$5.00.

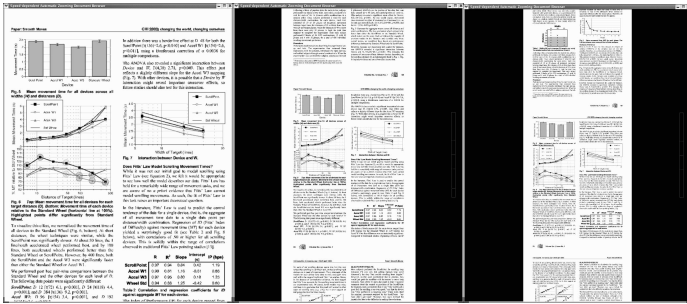


Figure 1. Automatic zooming with increasing scroll speed (slow, medium, fast).

finishing locations, van Wijk & Nuij suggest how it could be deployed in scrolling. Until now the scrolling theory has not been empirically tested.

This paper describes our attempts to better understand human perceptual foundations of systems that automatically change zoom-level with scroll speed. We review human-factors of visual processing and use it to predict a perceptual relationship between speed and zoom. These predictions are compared with empirical results ($n=20$), and the results are used to calibrate a text-based SDAZ system. We also implement and test van Wijk and Nuij's three suggestions for theoretically-based automatic zooming, and we empirically compare performance ($n=27$) of the most promising system with three other scrolling interfaces—our calibrated SDAZ system, normal scrollbars, and traditional rate-based scrolling. Finally, we discuss issues involved in developing commercial deployments of SDAZ.

BACKGROUND

Document editors and browsers such as Microsoft Word and Adobe Reader support a wide range of tools for scrolling. For example, Word's vertical scrollbar has been extended (shown right) to include shortcuts for scrolling to semantic items such as pages, sections, tables, figures, and keywords. Zooming is another standard user interface feature that influences scrolling behaviour because it alters the proportion of the document displayed within each window. The 'Dynamic Zoom' feature of Acrobat Reader 6 is an interesting exploitation of the scroll/zoom relationship. It allows simultaneous control of scrolling and zooming by binding scroll-wheel actions to zooming while dragging the mouse with the wheel depressed controls rate-based scrolling. Despite the availability of sophisticated tools such as these, the extent to which they are used remains largely unexplored.



Scrolling studies

Zhai et al. [22] conducted a comparative evaluation of four input devices for scrolling and pointing in ten page web documents. The four devices were a standard two button mouse, a scroll wheel mouse, a 'JMouse' which had a built-in isometric joystick for scrolling rate control, and

bimanual input with a keyboard-mounted isometric joystick controlling scrolling rate in one hand and a standard mouse in the other. Results showed that while the scroll wheel did not improve performance, rate-based scrolling with the mouse and keyboard mounted joysticks improved performance by 31% and 35% over the standard mouse.

More recently, Hinckley et al. [10] compared scrolling performance using the IBM ScrollPoint mouse (similar to the JMouse) against that of three different scroll wheel configurations: a standard three-lines per notch setting, and two settings that applied scrolling acceleration on rapid manipulation. Results indicated that scrolling with the ScrollPoint's isometric joystick was comparatively poor over short distances, but that it performed equally well to the best scroll wheel acceleration setting for long distances. Interestingly, they also showed that the acquisition of off-screen targets with scrolling is reliably modelled by Fitts' Law [8], which is normally applied only to targets within a single visible field.

Speed and zoom coupling

Igarashi and Hinckley [11] described five systems that automatically modify zoom-level with scrolling speed—a web browser, a map viewer, an image browser, a dictionary, and a sound editor. Only the web browser and map viewer were evaluated as the other applications did not seem promising. Their evaluation ($n=7$) showed similar performance between SDAZ and traditional interfaces. They stressed the desirability of further and larger scale evaluations.

Tan, Robertson and Czerwinski [17] produced positive results for coupling zoom level (or 'flying altitude') with movement speed. Participants were able to acquire and move targets in a 3D virtual world more rapidly when using a 'speed-coupled flying and orbiting' interface than when using more traditional mechanisms for 3D navigation. This result builds on that of Ware and Fleet [20] who showed improved user performance in 3D 'fly-by' navigation when speed was automatically coupled to flying altitude (essentially the inverse of SDAZ).

Our previous evaluation ($n=12$) showed positive results for SDAZ [6]. We re-implemented and evaluated interfaces similar to the web-browser and map-browser demonstrated by Igarashi and Hinckley. Our OpenGL interfaces provided high frame-rates and fluid interaction with real documents and maps, rather than the synthesized ones used in Igarashi and Hinckley's study. Results showed that SDAZ significantly out-performed traditional interfaces. Although the results are positive, there are validity concerns arising from the fact that neither of the competing interfaces supported rate-based scrolling. There is a risk that rather than scrutinising the impact of automatic zooming, the experiments actually discriminated between rate-based scrolling and traditional scrollbars. Because SDAZ is essentially rate-based scrolling (with automatic zooming)

there is a risk that the performance benefits resulted from users not having to acquire the scrollbar before scrolling.

Optimal Pan-Zoom Paths

In recent work, van Wijk and Nuij [19] present a theoretical analysis of panning and zooming in 2D information spaces. Their formulae calculate optimal camera paths that produce smooth and efficient animations between discrete start and finish locations in ‘zoom/pan space’ (i.e., from one coordinate at one level of zoom to another coordinate at another level of zoom). Their formulae are parameterised by two constants, V and ρ . V constrains the ‘information flow’ and is measured in units of screen-widths per second. The second parameter ρ is a measure of the human-perceptual trade-off between zooming and panning, with higher values of ρ producing animations with more zooming, and lower values producing relatively more panning. van Wijk and Nuij suggest values for V using empirical methods, and for ρ using both theoretical and empirical methods.

Interestingly to us, van Wijk and Nuij suggest three schemes for deploying their formulae within scrolling variants of speed-dependent automatic zooming. Although primarily intended for navigation between pre-determined locations in pan/zoom space they suggest adaptations allowing their formulae to work with scrolling where the final destination is normally dynamically determined by the user. None of these schemes were empirically tested, but van Wijk and Nuij’s theoretical analysis is inspiring.

We implemented all three of their suggested scrolling systems, as described below.

Human processing of visual flow

When scrolling and zooming, users must visually track moving targets. Research into human visual perception shows that visual signals are summated over a period of approximately 120-125ms in daylight [3, 5]. Visual blurring of moving objects is dramatically less than might be expected from a 120ms ‘exposure’ time [3, 21]. Morgan and Benton [15] showed that images can move across the retina at up to 3 degrees per second without an effect on image acuity. At higher speeds moving images are stabilised on the retina by tracking them with ‘smooth-pursuit’ conjugate eye movements [7]. Smooth-pursuit eye-tracking of objects succeeds up to angular velocities of approximately 100° per second [13]. These values are used in our theoretical approximation of SDAZ calibration.

CALIBRATING THE BEHAVIOUR OF AUTOMATIC ZOOMING

All scrolling systems are controlled through a mapping from the user’s manipulation of the input device to the resultant scrolling behaviour. In zero-order control, such as manipulation of the scroll-thumb, the mapping is from position to position—as the mouse position changes, the scroll position changes proportionately (control-display

gain determines the magnitude and acceleration of the mapping [14]). In first-order control, such as that in rate-based scrolling, the mapping is from position (or some other scalar such as force on an isometric input device) to speed—as the mouse is displaced further, the scroll speed increases.

Igarashi and Hinckley used an inverse relationship between speed and scale, shown in Equations 1 & 2: $v0$ is a predefined speed below which no zooming occurs, $s0$ is a predefined minimum scale, $d0$ is the mouse movement required before zooming begins, and $d1$ is a predefined maximum mouse movement beyond which further displacement has no effect on speed or zoom.

$$speed = \frac{v0}{scale} \quad \text{Equation 1.}$$

$$scale = s0^{(dy-d0)/(d1-d0)} \quad \text{Equation 2.}$$

Our SDAZ systems used a linear speed-scale relationship (Equations 3&4). Y_{ip} and Y_{cp} represent the mouse-down and current y-coordinate of the mouse, k is a constant governing the rate of zooming, and $threshold$ is a constant governing the minimum scroll-speed prior to zooming.

$$speed = |Y_{ip} - Y_{cp}| \quad \text{Equation 3.}$$

$$scale = k \times (speed - threshold) \quad \text{Equation 4.}$$

Neither Igarashi & Hinckley nor Cockburn & Savage report the values for their constants, and both papers report that several heuristics are applied to limit the adverse affects of rapid changes in zoom-level. In our experience, ‘tweaking’ the calibration of the device-speed-zoom relationship is one of the hardest parts of implementing SDAZ systems.

In order to aid SDAZ calibration we theoretically and empirically analyse the maximum visual flow rates at various levels of magnification.

Theoretical limits of smooth-pursuit

Smooth-pursuit visual tracking is limited by the angular rate that objects move before the eyes, which is dependent on the speed of object movement and the distance between the object and the viewer. In all the experiments described in this paper viewers sat approximately 50cm from a 19inch Compaq monitor, giving a viewport size of 35×27cm and angular dimensions at the eye of 42×31°.

To simplify discussion of visual flow rates at different levels of magnification, we discriminate between screen movement rate (SMR) and document movement rate (DMR), both measured in cm/sec. The screen movement rate is the speed that pixels move across the screen, while the document movement rate is the speed the document moves in 3D space. At 100% magnification (we use magnification mag as a percentage of full-size), the screen and document movement rates are identical. At a constant

document movement rate, the screen movement rate is halved when magnification is altered from 100% to 50%: see Equation 5.

$$DMR = \frac{SMR}{(mag / 100)} \quad \text{Equation 5.}$$

By applying the 100°/sec limit of the smooth-pursuit visual system [13] to the screen viewing distance of 50cm, gives an upper-bound for SMR of 87cm/sec. The topmost line of Figure 2 shows the theoretical document movement rates corresponding to the limit of smooth pursuit at different levels of magnification, calculated using Equation 5.

People’s ability to track and identify data such as items of text is likely to be substantially lower than the upper-bound of visual tracking. We assume that targets must remain on-screen long enough for users to carry out three actions: initiate the smooth-pursuit visual system (*InitSPTime*), visually process the image or read the text (*IdentTime*), and recognise the target (*RecogTime*). For simplicity, we assume that these steps are completed in series, however, in reality some are likely to be completed in parallel. Blohm and Schreiber [1] estimate that initiating the smooth pursuit visual system, *InitSPTime*, takes between 100-150ms. Reading a three-word heading takes approximately 600ms assuming reading speeds of approximately 280 words per minute [12]. Once the heading is processed, it must be recognised as the target, *RecogTime*, for which we use Card et al.’s [5] 200ms estimation for one cycle of the visual and cognitive systems. Summing these values gives an estimated time of 950ms for reading and identifying a moving three letter heading. Finally, these limiting on-screen times can be converted into document and screen movement rates using equations 5 and 6, where *AppSize* is the height of the application window:

$$SMR = \frac{AppSize}{OnScreenTime} \quad \text{Equation 6.}$$

Assuming a maximised window size of 27cm vertically, Figure 2 plots the estimated maximum document scroll rates for tracking and identifying three word text headings.

Empirical analysis

The empirical analysis, like the theoretical approximation above, attempts to answer the question “how fast is too fast for comfortable perception when scrolling at different magnification levels?”

Participants were asked to find either images or two/three word headings within a document set at a specific magnification level. Software continually logged their scrolling velocity. Rather than ask participants to set a maximum scroll speed, we wished to observe their natural scrolling behaviour and use that to characterise their maximum scrolling speeds.

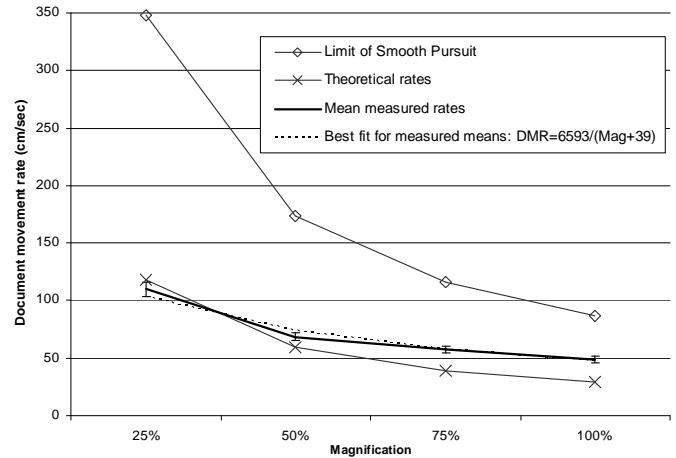


Figure 2. Theoretical and empirical measures of the speed/magnification relationship.

Apparatus

The experiment ran on Athlon 2200+ computers with 512MB of RAM running Windows XP, outputting to 19inch Compaq monitors at 1024×768 resolution, driven by Geforce 4 ti4600 video cards. All software was written in C/OpenGL and it was executed in full-screen game mode at 70fps. Input was provided through an optical Logitech three button mouse.

The interface used for the tasks was our SDAZ document browser (described later), but with automatic zooming disabled. Documents were displayed within a maximised window at one of four magnification levels (100%, 75%, 50% or 25%). At 100% magnification all document text was clearly legible. At the 25% level the target text (section headings) was ‘barely legible’, basing the assessment of legibility on Tullis et al.’s study [18] which found that 7.5point Arial is ‘just readable’. Full font metrics used in the experiment are described by Savage [16]. Scroll speed was controlled through rate-based scrolling with a linear relationship between mouse displacement and velocity. The maximum document movement rate was set to exceed the theoretical limit for the smooth pursuit visual system, and was attained at a mouse displacement of 170pixels. Only vertical scrolling was supported. Scroll speeds were logged by software at 100ms intervals.

Two A4 documents were used in the tasks. For practice tasks (data discarded) the document was a 32 page scientific report, and for logged tasks it was a 157 page Masters thesis.

Method and Procedure

The participants’ tasks involved scrolling in a specified direction (up or down) for an image or a two/three word text heading. Each task was cued within a small window in the upper-right hand corner of the screen. Example image and heading tasks are “locate the first bar-graph up from here” and “locate the ‘Mapping Attributes’ heading down from here”. To familiarise participants with the

experimental procedure four practice tasks were completed prior to the logged ones. Following the practice tasks all participants completed two tasks of each type (locate image and locate heading) at all four levels of magnification.

The logged scroll-speed provides the source data for our analysis, but our interest lies with the maximum comfortable document movement rate at each magnification level, and not the mean speed. For this reason the analysed dependent measure is the upper-quartile (UQ: the median of the upper half of the data-set) of the scroll-rate per task. To inspect this decision we plotted the scroll-rates for five pre-test participants together with the upper-quartile measure (see Figure 3). The plots assured us that the UQ measure was successfully approximating the fastest sustained scroll-speed.

Participants

Twenty volunteer second-year Computer Science students participated in the study (14 male, 6 female). We collected a variety of background demographics including their experience with computer games.

Results

The upper-quartile scroll speeds per task were analysed in a 4×2 repeated-measures analysis of variance (ANOVA) for factors magnification-level and target-type. Although the existence/absence of statistically reliable differences between levels of these factors is interesting, it is not our primary objective. The primary objective is to characterise the maximum comfortable scroll speeds at various levels of magnification.

Over all magnification levels, the mean UQ document scroll-rate was 71.2cm/sec (standard deviation 34.6cm/sec). As expected, there is a reliable difference between UQ scroll-rates at different levels of magnification ($F_{3,57}=77.2$, $p<0.01$), which essentially means that decreasing magnification allows users to scroll the document more quickly. There was no reliable difference between the means for locating images and text headings ($F_{1,19}=0.3$, $p=0.6$), which suggests that a single calibration value may be robust for different target types. Finally, there was no reliable interaction between factors magnification- level and target-type ($F_{3,57}=0.83$, $p=0.48$).

Regression analysis shows accurate modelling through an inverse relationship between document movement rate and magnification: $DMR=6593/(Mag+39)$, $R^2=0.97$, $p<.05$. Figure 2 shows the theoretical and measured mean document movement rates at different levels of magnification. It also shows the limiting rates for smooth-pursuit and the calculated line of best fit for the means.

Calibration settings

Our SDAZ implementation maps from mouse displacement to document scroll rate, and then from document scroll rate to magnification level, using a linear-relationship of

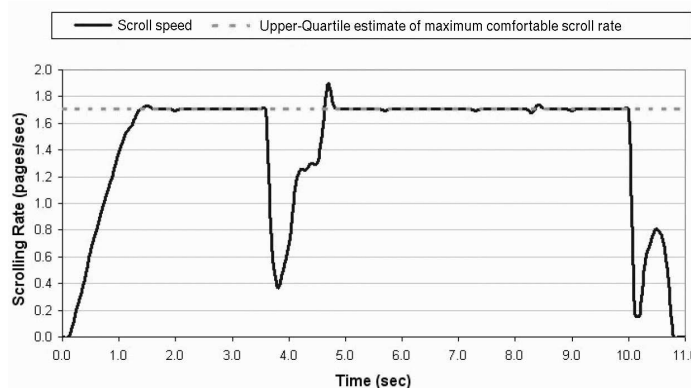


Figure 3. Scroll-rate (pages per second, 1 page=27cm) against time. The dashed line shows the upper-quartile estimate of maximum comfortable scroll rate.

$Mag=(DMR-120)/-0.78$. (We used the linear relationship, $R^2=0.86$, because we had not noticed the better inverse model; the difference between calibrations is small, except at low magnifications.) Our system constrains maximum and minimum magnifications to 100% and 10%. Document movement rate is set by a linear mapping with cursor displacement. A maximum displacement of 170pixels corresponds to a maximum document movement rate of 112cm/sec (and consequently a magnification of 10%).

Three further parameters are necessary to constrain the rate of change of magnification. Without these constraints there can be disturbingly rapid zooming during the onset of scrolling, when changing scroll direction, and when terminating scrolling. Igarashi and Hinckley also described these constraints, but did not provide parameter values. The three parameters and recommended values (determined by extensive informal trials) are as follows:

- Maximum ascent rate—this parameter prevents the excessive decrease in magnification (zoom-out) that can be caused by a sudden large mouse-drag movement. We recommend 150mag/sec, where *mag* is measured in magnification percentage.
- Maximum descent rate—this parameter prevents the disturbing experience of ‘slamming into the document’ (rapid increase in magnification) that occurs when changing scrolling direction. We recommend 40mag/sec. This constraint is very important. Users frequently scroll in one direction only to rapidly change direction on identifying a candidate target, characterised by “there it is, whoops overshoot, bring it back”.
- Maximum fall rate—this parameter constrains the rate of increase in magnification when the user releases the mouse button to cease scrolling. This rate is best set higher than the maximum descent rate as the user typically wants to ‘look at the target as fast as possible’. We recommend 175mag/sec.

COMPARING VAN WIJK & NUIJ'S SUGGESTIONS WITH SCROLLBARS, RATE-BASED SCROLLING, AND SDAZ

Having established calibration metrics for SDAZ we wished to compare its performance with that of the techniques suggested by van Wijk and Nuij. We also evaluated standard rate-based scrolling and traditional scrollbars for baseline comparison.

We implemented all three of van Wijk and Nuij's suggestions, based on the formulae they present in Equation 9 and Section 5.1 of their paper. Full details of these formulae are beyond the scope of this paper, but we refer precisely to them to aid those wishing to replicate our study. The three techniques differ in how user input is sampled, resulting in markedly different interaction experiences:

- *Positional sampling*, which we term *DDAZ* for 'displacement dependent automatic zooming'. The location of a virtual cursor is periodically sampled (approximately 50 times per second in our system) and input to the formulae. The user's experience is that they control and move a virtual scroll-thumb, and after each discrete thumb movement a pan/zoom animation quickly moves between the original and new document locations. While this works well for discrete movements it causes substantial 'view bouncing' if the user attempts to sustain a smooth and continuous scroll speed.

The values recommended by van Wijk and Nuij for parameters V and p resulted in scrolling speeds that were far too slow for our preliminary testers. We increased V from 0.9 to 1.5, but left p at the recommended value of 1.42. van Wijk and Nuij note that parameter modification may be necessary to compensate for lag in DDAZ.

- *Velocity sampling*, which we term *VDAZ* for 'velocity dependent automatic zooming'. The velocity of the virtual cursor is continually sampled and input into the formulae. Mouse control of this technique proved awkward and counter-intuitive. For example, it seems reasonable to expect a single large but fast upwards mouse movement to scroll a substantial distance towards the beginning of the document. However, as the formulae induce more zooming than panning during initial motion, the resultant effect is a rapid zoom-out and back in with little scrolling. We discarded VDAZ from this study, but intend to investigate its effectiveness with rate-control input devices (such as an isometric joystick) in the future.
- *Combined position and velocity sampling*, which we term *CDAZ*. DDAZ suffers from a lag between user action and resultant view animation. VDAZ suffers from a lack of fine position control and an excess of mouse movement. CDAZ combines the two techniques in an attempt to ease their deficiencies. In our experience, however, the technique is awkward to use, with several trial users complaining that it was 'unpredictable'.

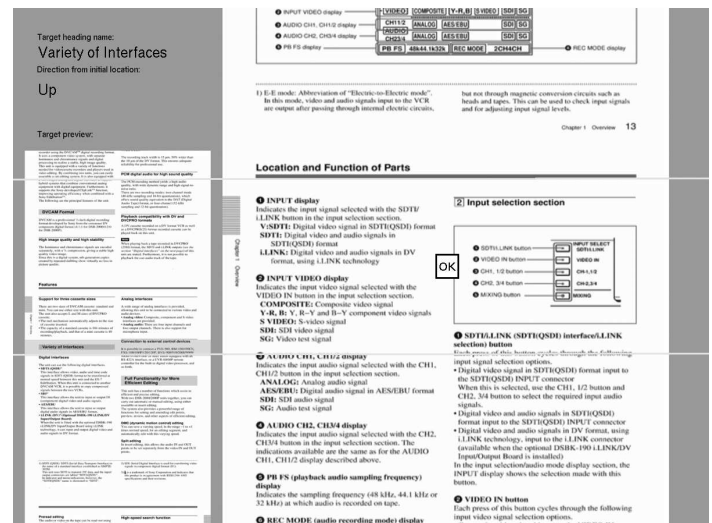


Figure 4. Evaluation interface. The pane on the left cues the tasks, showing the target text, initial direction, and target preview. The main window contains the scrolling interface with the target region bounded by yellow bars.

Of the three van Wijk & Nuij systems we felt that DDAZ was a viable competitor to SDAZ. The discarded variants may improve with isometric input devices, but we wanted to focus on mouse input as it remains the de-facto standard pointing device.

Evaluation

This evaluation compares visual-search scrolling performance using traditional scrollbars, rate-based scrolling, DDAZ and SDAZ. The rate-based, DDAZ and SDAZ interfaces were all controlled by dragging with the left mouse button. The traditional scrollbar interface was the only one that required a visible GUI component to be directly manipulated—before beginning to scroll the users had to acquire the scroll-thumb, trough, or arrow. A video demonstrating the interfaces and the experimental method is available at: www.cosc.canterbury.ac.nz/~andy/tuning-n-testing.wmv

Apparatus

The experiment ran on Athlon 1600+ computers with 256MB of RAM running Linux 9, outputting to 19inch Compaq monitors at 1280×1024 resolution, driven by Geforce 2MX video cards. The systems ran in full-screen game mode at 70fps. Input was through an optical Logitech three button mouse. All interfaces were created from the same core C/OpenGL program.

Method and Procedure

Similar to the calibration experiment, the participants' tasks involved vertically scrolling through a document seeking document section headings. Tasks were cued within a window (left hand side of Figure 4) which showed the text of the target heading, the direction of the target from the starting location (up or down), and a picture-preview of the target within the document. Tasks were complete when the

target text was at full-zoom and stationary (mouse up) within the middle third of the scrolling window. The middle third was revealed by horizontal yellow lines across the screen. Each new task was automatically presented by software when the preceding task was complete. Participants were encouraged to carefully study the cued target prior to clicking the “OK” button to begin each task. Software timed all tasks and logged scroll-speed and zoom-level at 100ms intervals.

All participants used all four interface types in a random order, with training for each interface immediately preceding the logged tasks. Training consisted of five minutes undirected browsing, followed by six practice tasks that were cued in precisely the same way as the logged ones. The documents used for practice and logged tasks were both manuals for Sony digital camcorders with the index and table of contents removed, as follows: practice tasks, 40 pages for model DSR2000; logged tasks, 137 pages for the model DSR-500WSL.

Nine logged tasks were completed with each interface: three at each of short, medium, and long distances, which had targets four, ten and twenty pages from the starting location respectively. Participants were not informed that targets were a discrete level of distance away, and the order of exposure to the various distances was random per task.

While we expected most tasks to be completed fairly rapidly, we anticipated that users would sometimes become lost in the relatively long document (137 pages). To avoid data from ‘lost’ tasks skewing the results we planned to cap task completion at 30seconds, replacing lost tasks with the 30second value. Participants were allowed to continue searching for up to two minutes per task, after which the experimenter pressed a key-sequence to advance to the next task. In the results we analyse capped data, the number of capped tasks per interface, and log-transformed uncapped data. Log-transformation for uncapped tasks is used to stabilise the naturally wide variance of performance in ‘lost’ tasks.

Prior to beginning the experiment we collected background demographics of the participants, particularly scrutinising game-playing experience. After completing all tasks with each interface the participants completed NASA-TLX worksheets [9] to rate the associated workload. They were also asked to comment on the interface. At the end of the experiment they were asked to rank the interfaces by overall preference.

Participants

Twenty seven undergraduate Computer Science students participated in the study (20 male, 7 female). None had previous experience with automatic zooming interfaces. We classified fifteen participants who played interactive computer games for less than one hour per week as ‘non-gamers’ and twelve as ‘gamers’ with between one and 30 hours of play per week.

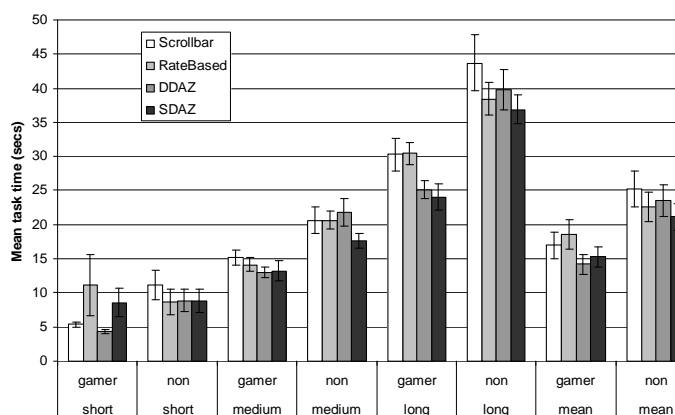


Figure 5. Uncapped mean task times for the four interfaces by gamers and non-gamers across the three distances. Cross distance means are also shown. Error bars show the mean ± one standard error.

Experimental Design

The primary dependent variable is task completion time: the time taken to place the target heading within the middle third of the screen. We analysed this data using a 4×3×2 mixed factors ANOVA. Within-subjects factor ‘interface-type’ has four levels: scrollbars, rate-based, DDAZ and SDAZ. Within-subjects factor ‘distance’ has three levels: short, medium and long for tasks that are 4, 10 and 20 pages from the starting location. Between-subjects factor ‘gaming’ has two levels: gamer and non-gamer.

Subjective measures of workload and overall preference are also analysed.

Results

Target acquisition

Of the 972 tasks (27 participants using 4 interfaces at 3 distances, and three tasks per distance), 14 were abandoned after 120 seconds: 7 with DDAZ, 5 with Ratebased, 2 with Scrollbars and none with SDAZ. Data from these tasks are discarded. The analyses of capped tasks and of log-transformed uncapped tasks reveal the same results in terms of statistical significance. Figure 5 shows mean performance across all factors for uncapped analyses.

With the 30-second cap on task completion, there is a reliable difference between mean times taken to scroll to targets using the four interface types ($F_{3,75}=3.6, p<0.05$), with SDAZ supporting fastest acquisition (15.9secs, s.d. 8.4) followed by DDAZ (16.8secs, s.d. 9.3), scrollbars (17.5secs, s.d. 9.1) and rate-based (17.6secs, s.d. 9.2). However, the thirty second time-cap introduces a potential confound: if, for instance, SDAZ allowed faster acquisition for successful tasks, but encouraged participants to become lost, this would not be apparent from the comparison of capped means. Inspecting the number of capped tasks shows that SDAZ also had the lowest proportion (44 tasks, 18%) followed by DDAZ (54, 22%), scrollbars (57, 23%)

Table 1. Summary of the subjective measures: NASA-TLX workload ratings and for overall preference rankings.

	NASA Task Load Index: Mean (s.d.)						Preference Ranking: counts (cumulative %)			
	Mental Load	Physical Load	Temporal Load	Performance	Frustration	Effort	1 st	2 nd	3 rd	4 th
Scrollbars	3.2 (1.2)	3.4 (1.0)	3.3 (0.8)	3.3 (0.9)	3.1 (0.9)	3.1 (0.8)	0 (0)	8 (30)	10 (67)	9 (100)
Ratebased	3.0 (1.1)	2.5 (1.2)	3.0 (0.9)	2.9 (0.9)	2.8 (0.9)	2.7 (1.1)	6 (22)	11 (63)	5 (81)	5 (100)
DDAZ	3.4 (1.1)	3.4 (1.1)	3.2 (1.0)	3.4 (1.1)	3.4 (1.2)	3.2 (1.2)	3 (11)	2 (19)	9 (52)	13 (100)
SDAZ	2.8 (1.0)	2.3 (1.0)	2.4 (1.0)	2.6 (1.3)	2.4 (1.1)	2.6 (1.2)	18 (67)	6 (89)	3 (100)	0 (100)
***<.01; *<.05		***	***	***	*	*				

and rate-based (60, 25%). The analysis of log-transformed uncapped means also shows the same interface order ($F_{3,75}=2.94, p<0.05$).

Both capped and uncapped analyses show a reliable difference between mean performance by gamers and non-gamers, with capped means of 14.8secs (s.d. 8.5) and 18.7secs (s.d. 9.1) respectively ($F_{1,25}=36.4, p<0.01$); uncapped means 16.2 (s.d. 10.8) and 23.1secs (s.d. 15.2), $F_{1,25}=38.2, p<0.01$. Despite the difference in overall performance, we were surprised to see that there is no interface×gaming interaction (capped $F_{3,75}=1.5, p=0.23$, uncapped $F_{3,75}=1.7, p=0.18$). We had suspected that gamers might be better able to handle the combination of panning and zooming in SDAZ and DDAZ than non-gamers, but the absence of an interaction provides no supporting evidence, suggesting that the utility of automatic zooming extends beyond those familiar with rapidly interactive systems.

As expected, there is a reliable main effect for ‘distance’ in both capped and uncapped analyses (capped $F_{1,25}=863.9, p<0.01$). No other interactions were significant in either capped or uncapped analyses.

Subjective workload assessments and preferences

After using each interface the participants used the NASA-TLX worksheets to rate six dimensions of ‘workload’ using 5-point Likert scales. Responses show that SDAZ received the lowest (best) mean workload assessment on all dimensions, with statistical differences between the four interfaces (Friedman χ^2 tests) on all dimensions except Mental Effort (see Table 1). Despite the relatively good task acquisition performance of DDAZ it was rated poorly under most workload measures.

Overall preferences for the interfaces (also shown in Table 1) were strongly in favour of SDAZ. Sixty seven percent of participants ranked it first, and 89% ranked it first or second; none ranked it last. DDAZ, by comparison, performed poorly, with 81% ranking it either worst or 2nd worst. Several of the participants explained that they disliked DDAZ because it was “too bouncy”, “too difficult to maintain a constant level of zoom”, “annoying to have to keep moving the mouse to scroll”, and “bad with fine adjustments”. Despite the negative comments about DDAZ, the automatic zooming of SDAZ was generally liked: “it’s nice, smooth and fast”, “I think I am good at this one

because it zooms slowly so that I can have time to look at the words and pictures in each page”, and “I could fix the zoom to a comfortable level easily, and that also matched the scroll speed to my liking”. SDAZ also received several negative comments, with one participant commenting that “sometimes it’s not easy on the eyes” and another stating “initially it kept haring off all over the place”. Three comments provide insight into a SDAZ usability issue that we wish to further investigate: “sluggish for short distances”, “hard to stop once you overshoot”, and helpfully “required a lot of concentration to slow down because you want to make sure it doesn’t go the other way”. These comments suggest that the parameter setting for Maximum Descent Rate may need further investigation. Precise positioning also remains tricky with SDAZ, and with rate-based scrolling in general, because displacement maps to velocity rather than location. Hinckley et al. [10] also noted that rate-based input with ScrollPoint performed relatively poorly for short distances.

Despite its relatively poor target acquisition performance rate-based scrolling was the second preferred interface, with 63% of participants ranking it first or second. Comments showed that users found the physical load to be low because they did not having to continually move the mouse or acquire the scrollbar. Like SDAZ, several participants noted the problem of precise position control caused by the displacement to speed mapping. Finally, several participants commented that the rate-based system induced eyestrain, suggesting that zooming is not the sole cause of the eyestrain reported with SDAZ and DDAZ.

DISCUSSION

Results of the experiment suggest that for visual search tasks our calibrated implementation of speed-dependent automatic zooming outperforms other scrolling techniques. Both automatic zooming techniques (SDAZ and DDAZ) had lower mean task times than traditional scrolling interfaces. Subjective preferences and workload assessments were strongly in favour of SDAZ. This raises the question, should major software vendors include SDAZ in their systems? And if they do, how should they do it, and what implementation issues are they likely to encounter?

If SDAZ is included in document browsers/editors such as Adobe Reader or Microsoft Word it should be provided as an optional behaviour of rate-based scrolling. Rate-based

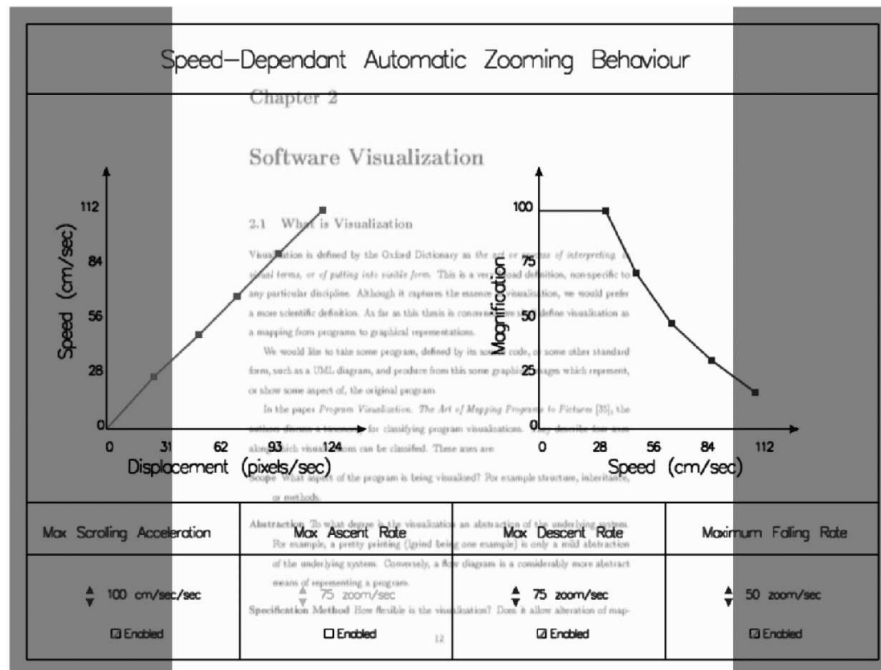


Figure 6. An interface for the precise calibration of SDAZ. Settings can be dynamically adjusted and tested against the document under the translucent calibration overlay.

scrolling is already a standard facility that co-exists with normal scrollbars, each serving distinct purposes with the scrollbar excelling at spatial-location tasks, while many prefer rate-based scrolling for visual searches. Our experiments indicate that SDAZ enhances rate-based scrolling, but many users will dislike the visual effect of automatic zooming. Anecdotal evidence suggests that rate-based scrolling already polarises users' opinions with many loving it and many others hating it. Similar polarisation is evident in users' reaction to the visual 'fisheye' effect present in MacOSX's 'Dock' icon-panel. If provided, SDAZ should be enabled and disabled through a check-box similar to Adobe Reader's toggle for 'Dynamic Zoom'.

System performance is another critical issue. SDAZ needs to be rapidly and smoothly animated to work effectively. Acrobat Reader's 'Dynamic Zoom' barely achieves this, despite using extensive clipping and text-greeking to reduce rendering times. Our SDAZ system relies on graphics hardware acceleration to achieve high frame rates, but despite the fact that almost all desktop computers include powerful graphics cards few office systems exploit them.

Our study of visual flow yields calibration settings based on 'average' performance settings, but expert users are likely to want to adjust the settings. Such an interface is non-trivial because of the relatively complex mapping from mouse displacement to scroll speed to zoom level. Figure 6 shows our current solution, which allows all parameters to be adjusted via two direct-manipulation graphs, and four other parameter entry widgets. The customisation interface lies on a transparent window above the document browser,

allowing users to dynamically reconfigure and test their modified settings. A commercial deployment would almost certainly want to hide such a complex set of parameter settings behind an 'advanced' button, preferring to allow users to select between named pre-compiled parameter settings such as "zoom early", "zoom late", etc.

Finally, the scrolling interfaces compared in our experiment either supported no zooming (scrollbars and rate-based) or automatic zooming (SDAZ and DDAZ). The question remains whether users would prefer and perform better when given separate, parallel controls for scroll and zoom. Prior studies have shown that bimanual separation of controls for scrolling and zooming can improve over one-handed input [2]. Similar results have been shown for bimanual separation of scrolling and pointing [22].

We recently completed a study comparing SDAZ with bimanual control of rate-based scrolling (mouse input) and manual zooming (keyboard input). Participants (n=35) performed significantly better using SDAZ, rated workload dramatically lower with SDAZ, and strongly preferred it. Full details of this experiment will follow.

CONCLUSIONS

The results of this paper strongly suggest that scrolling performance in visual search tasks can be improved by systems that automatically zoom away from the document as scrolling speed increases. Calibrating the relationship between speed and zoom is complex, and the identification of parameters and associated values should aid others wanting to implement speed-dependent automatic zooming interfaces.

There are many areas for further work. These include analysis of 2D scrolling through document types such as maps and images, and investigation of how different implementations of SDAZ compare when controlled through alternative input devices such as isometric joysticks.

ACKNOWLEDGMENTS

Thanks to the experiment participants and to the anonymous reviewers for their helpful comments

REFERENCES

1. Blohm, G. and Schreiber, C. The Smooth Pursuit System, www.auto.ucl.ac.be/EYELAB/neurophysio/perception_action/SP.html 2002.
2. Bourgeois, F. and Guiard, Y., Multiscale Pointing: Facilitating Pan-Zoom Coordination. in *CHI '02 extended abstracts of CHI2002*, (Minneapolis, Minnesota, USA, 2002), 758-759.
3. Burr, D. Motion Smear. *Nature*, 284 (13). 1980, 164-165.
4. Byrne, M., John, B., Wehrle, N. and Crow, D., The Tangled Web We Wove: A Taskonomy of WWW Use. in *Proceedings of CHI'99 Conference on Human Factors in Computing Systems*, (Pittsburgh, 1999), 544-551.
5. Card, S., Moran, T. and Newell, A. *The Psychology of Human-Computer Interaction*. Lawrence Erlbaum Associates, 1983.
6. Cockburn, A. and Savage, J., Comparing Speed-Dependent Automatic Zooming with Traditional Scroll, Pan and Zoom Methods. in *People and Computers XVII (Proceedings of the 2003 British Computer Society Conference on Human-Computer Interaction.)*, (Bath, England, 2003), 87-102.
7. Eckert, M. and Buchsbaum, G. The Significance of Eye Movements and Image Acceleration for Coding Television Image Sequences. in Watson, A. ed. *Digital Images and Human Vision*, MIT Press, 1993, 90-98.
8. Fitts, P. The Information Capacity of the Human Motor System in Controlling the Amplitude of Movement. *Journal of Experimental Psychology*, 47. 1954, 381-391.
9. Hart, S. and Staveland, L. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. in Hancock, P.a.M., N ed. *Human Mental Workload*, 1988, 139-183.
10. Hinckley, K., Cutrell, E., Bathiche, S. and Muss, T., Quantitative Analysis of Scrolling Techniques. in *Proceedings of CHI'2002 Conference on Human Factors in Computing Systems*, (Minneapolis, Minnesota, 2002), 65-72.
11. Igarashi, T. and Hinckley, K., Speed-dependent Automatic Zooming for Browsing Large Documents. in *Proceedings of the 2000 ACM Conference on User Interface Software and Technology*, (San Diego, California., 2000), 139-148.
12. Kang, T. and Muter, P. Reading Dynamically Displayed Text. *Behavior and Information Technology*, 8 (1).1989, 33-42.
13. Krauzlis, R. and Stone, L. Tracking with the Mind's Eye. *Trends in Neurosciences*, 22.1999, 544-550.
14. MacKenzie, I. and Riddersma, S. Effects of Output Display and Control-Display Gain on Human Performance in Interactive Systems. *Behaviour and Information Technology*, 13 (5). 1994, 328-337.
15. Morgan, M. and Benton, S. Motion-deblurring in human vision. *Nature*, 340.1989, 385-386.
16. Savage, J. The Calibration and Evaluation of Speed-Dependent Automatic Zooming Interfaces *Masters Thesis*, University of Canterbury, Christchurch, 2004.
17. Tan, D., Robertson, G. and Czerwinski, M., Exploring 3D Navigation: Combining Speed-coupled Flying with Orbiting. in *Proceedings of CHI'2001 Conference on Human Factors in Computing Systems*, (Seattle, Washington, 2001), 418-425.
18. Tullis, T., Boynton, J. and Hersh, H., Readability of fonts in teh windows environment. in *Extended Abstracts of CHI'95*, (1995), 127-128.
19. van Wijk, J. and Nuij, W. A Model for Smooth Viewing and Navigation of Large 2D Information Spaces. *IEEE Transactions on Visualization and Computer Graphics*, 10 (4). 2004, 447-458.
20. Ware, C. and Fleet, D., Context Sensitive Flying Interface. in *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, (Providence, RI, 1997), 127-130.
21. Zagier, E. A Human's Eye View: Motion Blur and Frameless Rendering. *ACM Crossroads*, 3 (4). 1997, 8-12.
22. Zhai, S., Smith, B. and Selker, T., Improving Browsing Performance: A Study of Four Input Devices for Scrolling and Pointing Tasks. in *Proceedings of INTERACT'97: the sixth IFIP conference on Human Computer Interaction*, (1997), 286-292.