Zoofing! Faster List Selections with Pressure-Zoom-Flick-Scrolling

Philip Quinn and Andy Cockburn
Department of Computer Science and Software Engineering
University of Canterbury
Christchurch, New Zealand
philip.quinn@canterbury.ac.nz, andy@cosc.canterbury.ac.nz

ABSTRACT
The task of list selection is fundamental to many user interfaces, and the traditional scrollbar is a control that does not utilise the rich input features of many mobile devices. We describe the design and evaluation of *zoofing*—a list selection interface for touch/pen devices that combines pressure-based zooming and flick-based scrolling. While previous flick-based interfaces have performed similarly to traditional scrolling for short distances, and worse for long ones, *zoofing* outperforms (and is preferred to) traditional scrolling, flick-based scrolling, and OrthoZoom. We analyse experimental logs to understand how pressure was used and discuss directions for further work.

Author Keywords
Zooming, scrolling, flicking, list selection, pressure.

ACM Classification Keywords
H.5.2 Information interfaces and presentation: User interfaces (*Interaction styles, screen design*).

INTRODUCTION
Mobile devices are continuing to grow in both computing power and consumer popularity; following these is the demand for these devices to support complex, desktop-like interfaces and applications. Mobile devices are limited in their form factor and input techniques—typically equipped with a physically small screen at a low resolution, and either a keypad, or more recently, a finger or stylus pen for input. Repackaging the desktop metaphor onto these devices without consideration of their constraints can result in tiresome and tedious interaction with them.

Selecting an item from a list is a fundamental task for many user interfaces: for example, choosing a country, font, or song from among hundreds or thousands of candidates. On the desktop this is typically facilitated by a scrollbar and often a mouse wheel (sometimes supplemented with keyboard shortcuts in alphabetically sorted lists). Mobile devices have different constraints: the mouse wheel is unavailable, scrollbar widgets can clutter the limited screen real estate, and can be difficult to acquire and manipulate. Despite these constraints, mobile devices also enable new interaction styles through stylus/touch input and pressure sensing.

OZCHI 2009, November 23-27, 2009, Melbourne, Australia.

OZCHI 2009 Proceedings ISBN: 978-1-60558-854-4

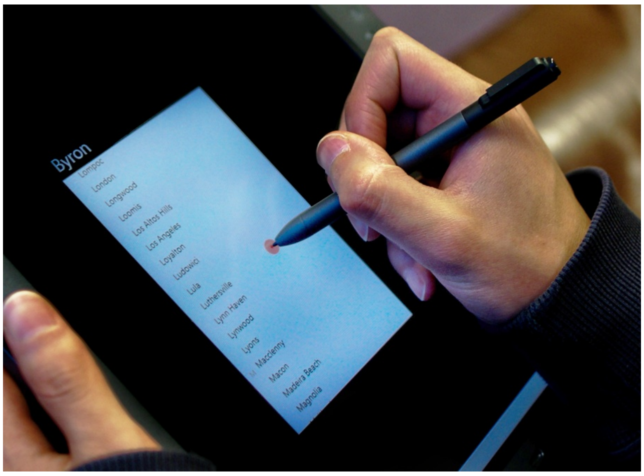


Figure 1: Zoofing: An interface where a user zooms and pans to select an item in the list.

Recently, designers and researchers have explored *flicking* as an alternative to the scrollbar on mobile devices (Aliakseyeu, Irani, Lucero & Subramanian 2008, Reetz, Gutwin, Stach, Nacenta & Subramanian 2006), using a familiar and natural throwing gesture to scroll the viewport. Some implementations also simulate physical effects such as inertia and friction to make the interaction feel more natural. However, it is difficult to find a robust mapping function between flick action and scroll length that is appropriate for all list lengths (Aliakseyeu et al. 2008): aggressive functions aid rapid movement but are imprecise, and passive functions demand excessive flicking to traverse long distances.

Zooming allows users to control the effective mapping between action and effect—a small scroll action will traverse the entire list when zoomed-out, or only a few items when zoomed-in. Several projects have demonstrated that performance in long distance scrolling can be improved by combining zooming and scrolling (Appert & Fekete 2006, Cockburn, Savage & Wallace 2005, Igarashi & Hinckley 2000).

We coupled pressure and zoom to complement the natural behaviour of flick scrolling in a new technique called *zoofing* (pressure-*zoo*m-*f*lick-scroll*ing*). It eliminates the scrollbar through pressure-based zooming and flick-based scrolling—using the list itself as the control surface. Starting from an overview of the entire list, users fluidly select discrete items by applying pressure to accelerate towards the list, then dragging and flicking to final aquisition. Evaluation results show that *zoofing* is comparable to scrolling in short lists (and significantly faster than both flicking and OrthoZoom (Appert & Fekete 2006)), and that it performs significantly better than scrolling with long lists.

BACKGROUND

Zoofing combines aspects of both zooming and scrolling interfaces. Here we review the relevant work from these areas that provide us with insight into potential design issues.

Scroll-Zoom Relationships

Zooming and panning (two dimensional scrolling) are often paired in navigation systems: zooming controls the scale of information shown, while panning controls the focus. Interaction with zooming and panning controls are typically distinct—users complete an operation with one control before using the other (Appert & Fekete 2006). Zooming and panning have also been instrumental in multiscale navigation (Perlin & Fox 1993, Furnas & Bederson 1995, Cockburn, Karlson & Bederson upcoming 2009), where information is available at multiple scales and resolutions in a two dimensional information space.

Speed-Dependent Automatic Zooming (SDAZ) dynamically adjusts the zoom level in response to the user's scroll speed in order to overcome the disorientation that can be caused by fast rate-based scrolling, and maintains a constant perceptual scrolling speed (Igarashi & Hinckley 2000). Evaluations of SDAZ (Igarashi & Hinckley 2000, Cockburn et al. 2005) found that users preferred the SDAZ technique over non-zooming rate-based scrolling, and performance was comparable to that of other scrolling techniques.

OrthoZoom uses two degrees of input freedom to allow users to concurrently control scroll location and zoom level—horizontal position controls the zoom level, and vertical position controls the scroll position (Appert & Fekete 2006). Empirical evaluation revealed OrthoZoom to perform better than SDAZ when information is segmented in a multiscale fashion. Analysis of their results found that users typically performed their zoom actions separately from their panning actions, despite being able to control both simultaneously.

Mobile Interaction

Stylus or pen devices are commonly used on mobile devices, allowing precise targetting via the small stylus/pen tip. In more contemporary touch-screen devices, the use of a finger for interaction is becoming a preferred method, due to its habitual and convenient nature. However, finger-based touch has several limiting issues: the friction between the finger and the device surface (Buxton, Hill & Rowley 1985); and the low resolution of the human finger make it difficult to perform accurate selections or manipulations (Potter, Weldon & Shneiderman 1988). However, there has been investigation into techniques to imrpove the accuracy of selection on touchscreen devices (Potter et al. 1988, Albinsson & Zhai 2003, Benko, Wilson & Baudisch 2006).

Flicking

Flicking (also known as *momentum–* or *kinetic scrolling*) is a method observed on contemporary mobile devices wherein the user performs a 'flick' gesture on the list to cause it to scroll with a velocity proportional to the flick impulse velocity.[1] Aliakseyeu et al. (2008) present and evaluate several different mapping functions for flick-based interfaces. They found performance with flicking to be comparable with scrollbar-based panning.

Pressure

Herot and Weinzapfel's (1978) early exploration of the ability of the human finger to apply pressure to a touchscreen showed users could quickly learn direct manipulation pressure bindings on touchscreens (for example, pushing/pulling objects or rotation). More recently, Ramos et al. (2004) explored 'pressure widgets', finding that users have a reduced ability to control low levels of pressure, and suggesting that a transfer function be used to accommodate for it.

Zliding is a technique that allows for continuous parameter adjustment by using pressure to adjust the range of the parameter space, while using spatial location to select a value (Ramos & Balakrishnan 2005). Applying pressure to the widget 'zooms' the parameter space that can be selected spatially. Empirical evaluation of their widget revealed it to perform comparably with alternative interfaces.

PRESSURE-ZOOM-FLICK-SCROLLING (*ZOOFING*)

Zoofing uses pressure controlled zooming, dynamic target steering while zooming, and flick scrolling to acquire targets. It is primarily intended for use on small mobile devices, where long lists are prevalent (such as address books or song lists), where constrained screen real estate demands as few interface controls as possible, and where pressure sensitive input is common. While designing `zoofing`, we considered the following goals:

- *Rapid traversal.* The user should be able to perform rapid open-loop navigation that allows them to quickly traverse towards arbitrary locations in the list.
- *Reduce the need for precise motor control.* The interface should not require the user to make precise acquisitions or manipulations in order to control their navigation. Such demands are susceptible to slips and lengthy closed-loop navigation phases, particularly in noisy and unstable mobile settings.
- *Natural and familiar navigation.* The interface should appeal to natural and familiar metaphors when controlling zooming and panning parameters

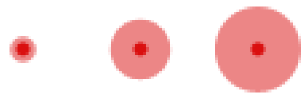
[1] `http://support.apple.com/kb/HT1636`

INTERFACE DESIGN AND INTERACTION

Zoofing initially presents a zoomed-out overview of every item in the list from where the user can begin to zoom into any region of the list (illustrated in Figure 2a); as the items are at an unreadable resolution, landmarks along the left side are shown to denote where particular regions within the list begin, but are not manipulable (landmarks remain at a constant visual size, irrespective of zoom). Navigation through to selection consists of three fluid steps, further described below—zoom steering into a particular list area, flick scrolling to refine the view, and tapping to select an item.

(a) **(b)** **(c)**

Figure 2: Applying pressure and zooming into a particular region of the list.



(a) **(b)** **(c)**

Figure 3: The cursor's visual feedback on the user's pressure through low (a), medium (b), and high (c) levels of pressure.

Zoom steering

Users initiate zooming towards an item by pressing on the touch sensitive display. Zooming is centred on the leftmost edge of the item lying directly under the pressure point: for example, pressing at the cursor location in Figure 2a will zoom towards items beginning with 'E'. The two relevant parameters in this action are the user's vertical position, and the amount of force they are applying on the device. Their position determines where the zoom action is centred (dynamically adjusted as the input device is dragged on the list), and pressure determines zoom acceleration towards the list.

To accommodate the variation in user's ability to control pressure levels (Ramos, Boulos & Balakrishnan 2004), we use a non-linear transfer function to describe the relationship between pressure and acceleration:

$$v = \sqrt{h} \cdot a \cdot p^b$$

Where h is the height of the entire list (in units—which correspond to pixels), p is the pressure applied (in the range $[0 \ldots 1]$, as reported by the device), and a & b are empirically derived constants accounting for varying sensitivities in different input devices (pilot studies determined values of 10 and 4 respectively for our device). Scaling the velocity with the height of the list helps to avoid lengthy waiting periods when zooming into larger lists.

Zooming stops if the user lifts the stylus, and the zoom level automatically falls back to the fully zoomed out state after a 3 second pause without touch. There is no manual control for zooming out.

In order to provide the user with adequate feedback about their zoom steering actions, the cursor is represented with a red dot within a translucent red ring (shown in Figure 3). The external ring expands with pressure, providing the user with real-time, fluid feedback about their pressure level on the device (not scaled with the transfer function).

Flick scrolling

Flick scrolling is only enabled once fully zoomed in, allowing users to move through the list with panning or flicking. We prohibit flick scrolling during the zoom steering phase because the two techniques conflict: during zoom steering dragging the stylus upwards on the device causes motion towards the start of the list; but an upwards flick gesture (similar to a drag) causes a scrolling action in the opposite direction. Furthermore, prior research has suggested that people naturally separate their zooming and panning actions (Appert & Fekete 2006).

Drag-based panning actions perform a 1:1 scrolling action on the list in the opposite direction of the drag action (equivalent to 'grabbing' the list surface). Drags are discriminated from flicks by the velocity and distance of the action. Flicks establish a scroll velocity that is gradually retarded through simulated friction. If the drag velocity exceeds a constant threshold,[2] a two second scroll animation is started for one second of distance at the velocity calculated, with smooth deceleration over the entire length of the animation. The animated scrolling can be stopped at any time by placing the pointing device back on the list. If either end of the list is reached, a 'bounce' animation stablises the display to the bounds of the list.

This implementation of flicking is a partial combination of the 'multi-flick-friction' and 'compound-multi-flick' techniques presented by Aliakseyeu et al. (2008)—utilising the smooth deceleration of the multi-flick-friction technique and elements of visual feedback in compound-multi-flick.

After three seconds of no activity (no active animations and no contact from the input device) a one second animated, interruptable effect returns the list to its original zoom level.

Tap selection

To select an item, users lift the stylus from the surface and tap it. Selections can be completed at any zoom level, thus preempting the need to reach a fully zoomed in state.

[2] A threshold was used to avoid confusing a dragging action with a flick; in our evaluations, this was set to half the height of the control.

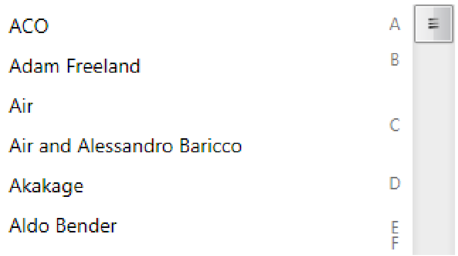


Figure 4: Section of the scrolling interface.

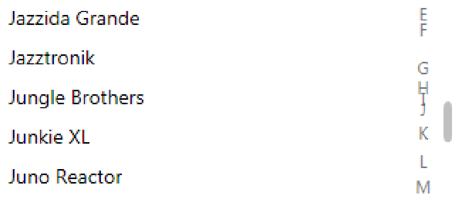


Figure 5: Section of the flicking interface.

(a) (b)

Figure 6: The OrthoZoom interface.

ALTERNATIVE INTERFACES

For evaluation pursposes, we considered three other techniques as alternatives to zoofing, described below.

Scrollbar

The traditional scrollbar is the definitive baseline interface for viewport manipulation. Our version placed a scrollbar down the right-hand side of the control (Figure 4). The scrollbar had a width of 30 units to assist in acquiring the scroll thumb; the vertical height of the scroll thumb scaled with the size of the list to a minimum of 3.4 units. Tapping in the troughs allowed users to scroll 95% of the control's height in either direction; the scroll arrows, traditionally at either end, were removed to give the scroll thumb as much granularity and precision as possible.

Landmarks were provided to the left of the scrollbar, and aligned to correspond with where the particular range started with respect to the position of the scroll thumb. Landmarks were visually identical to those provided in the zooming interface, and were not manipulable.

Flick scrolling

Flick scrolling is widely used on current mobile devices. By using the document surface for control, it negates the need to dedicate screen real estate to scroll widgets. Our flicking interface was designed to operate similarly to contemporary flick scrolling interfaces, and behaved identically to the flick scrolling interaction in zoofing. The interface was visually similar to the scrollbar interface; however, the scrollbar replaced with a narrow (10 unit) non-manipulable bar styled as an indicator of the current position (Figure 5).

OrthoZoom

Prior studies have shown OrthoZoom to outperform other scrolling interfaces (Appert & Fekete 2006). We include it in the evaluation as a 'state of the art' research comparison.

We implemented an interface that used OrthoZoom-style interaction, but redesigned to fit within the constraints of a mobile device.

The list started maximally zoomed-in at the top of the list, as shown in Figure 6(a). When the user placed the input device on the list, the x co-ordinate was mapped to a particular zoom level in a non-linear fashion:

$$(\frac{x^a}{w^a} \cdot (h \cdot r) - w) + w$$

Where w is the width of the control, h is the height the control, r is the aspect-ratio ($width/height$) of the control, and a is a scaling factor (set to 4 from pilot studies). Because our list was not segmented in a multiscale fashion (except for the alphabetic landmarks), this non-linear mapping allowed users more precise control over the zoom levels where the list was still at a readable resolution.

The list would always immediately jump to the zoom level the current x co-ordinate mapped to; when the input device was lifted, the list would jump back to being fully zoomed-in. Zooming was always centred around the current, or last-known, vertical position of the cursor.

Dragging the cursor along the y axis allowed the user to pan the list. There was a 1:1 mapping between the dragging distance and the list at its current zoom level, in the direction of cursor movement. The original OrthoZoom interface used a fixed rate of scrolling when the cursor moved outside the window. To create a similar control for a mobile device (where there is no space outside the window) we used a fixed rate of scrolling when the stylus entered a shaded area consuming 10% of the window's height at either the top or bottom of the window. The constant scroll rate was $b \cdot z$ Where z is the current zoom factor and b is a constant value of 10 based on pilot studies. This scrolling adjustment was applied approximately every 0.15 seconds.

EVALUATION

Two experiments investigated interaction with *zoofing* and assessed its performance and preference. The first compared four designs in lists of 300 items: *zoofing*, flick scrolling, traditional scrolling, and OrthoZoom. The second focused on traditional scrolling and *zoofing* in long lists of 1500 items. We were particularly interested in how these interaction techniques compared to one another, but also how each interface performed with respect to the location of the item to be selected within the list. Finally, we wanted to characterise and understand how users chose to interact with *zoofing*.

Apparatus

The experimental software ran on an HP Compaq tc4400 Tablet PC running Windows Vista; the device operated in portrait mode (12.1″/106ppi screen, 768×1024 screen resolution), and was controlled using the wireless stylus.

The experimental application was developed using WPF/C#. The application ran full screen, with the list interface displayed at a size of 320×480 (7.67×11.51cm), centred.

Task and Stimuli

Tasks involved repeatedly selecting cued items. Correct selections were briefly highlighted green, incorrect ones red. Each trial continued until correctly completed.

Participants received two minutes training with each interface, and completed at least ten trials (data discarded). They then completed 46 trials (with the initial trial discarded as a dummy trial to allow adjustment to the new list data) with each interface (balanced using a Latin square), with fifteen randomly distributed through each of the first, second, and third 100 items (for short, medium, and long distances in the list). Participants completed a NASA-TLX workload questionnaire (Hart & Staveland 1988) after each interface, and ranked them in order of preference after the completion of all interfaces. Trial completion time was measured from the time they were prompted with the item to select, to the time they made the correct selection.

Items were rendered in 14pt *Segoe UI*, landmarks were rendered in 12pt, at 75% opacity. Each item had a bounding box of 320 × 28, within which it could be selected—text was rendered left-aligned and vertically centred within each box. When fully zoomed in, 17 items were shown within the viewport with all four interfaces.

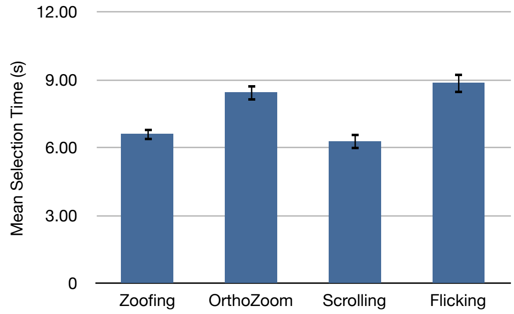
Experiment One

Participants

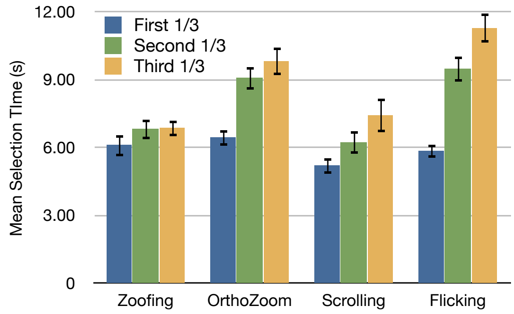
Sixteen (fifteen male, one female; thirteen right-handed) volunteers participated in the experiment. Most had little or no experience using a stylus or pressure sensitive devices. Participants received a $10 gift-certificate as compensation.

Design and Procedure

A 4 × 3 within-participants design was used. The independent variables were *interface* (Scrolling, Flicking, zoofing, and OrthoZoom) and *selection range* (first third, second third, and last third of the dataset); the primary dependent variable was *selection time*, and errors were also analysed.



(a) Mean selection time (s) per interface condition (+/- SE).



(b) Mean selection time (s) per selection range per interface condition (+/- SE).

Figure 7: Experiment one selection time results.

The lists were composed of a random selection of 300 artists to reach number one on the Billboard Hot 100 since 1955. A different list was used for each interface, but the same lists were used across all participants.

Results

We removed outliers from the data set. A trial was considered an outlier if the trial completion time was beyond three standard deviations from the mean trial completion time for that combination of interface and selection range. A total of 137 outlier tasks were discarded (4.8% of the data collected).

Analysis of variance showed a significant main effect for interface ($F_{3,45} = 39.6$, $p < 0.001$). Scrolling had the best performance (6.29s, s.d. 3.05), followed by zoofing (6.6s, s.d. 1.72), OrthoZoom (8.45s, s.d. 2.4), and flicking (8.87s, s.d. 3.0), shown in Figure 7(a). A post-hoc Tukey test produced an HSD of 1.35s ($\alpha = 0.05$), revealing significant differences between all interface pairs except for OrthoZoom/flicking and scrolling/zoofing. Error analysis showed significantly more errors with flicking, but otherwise no pairwise effects.

As expected, there was a significant main effect for *selection range* ($F_{2,30} = 137.3.6$, $p < 0.001$), but more interestingly there was also a significant *interface*×*range* interaction ($F_{6,90} = 23.56$, $p < 0.001$), which is best explained by *zoofing* showing relatively stable performance across *range* in contrast all other interfaces (see Figure 7b). This suggests *zoofing* successfully meets our design goal of supporting rapid traversal independent of target location.

Subjective Results

Subjective workload assessments and preference rankings were positive for scrolling and *zoofing* (Table 1) and particuarly negative for OrthoZoom, with many users criticising it as "hard to control". Its poor performance may be explained by the short training period or by differences in the data-space structure (our experiment used a flat list; the original experiment used a hierarchical structure providing natural landmarks at varying scales).

Experiment Two

We were also interested in how the performance of these interaction techniques scaled with the size of the list they were operating upon. Given the performance of the flicking and OrthoZoom techniques in experiment one, we decided to exclude them from experiment two, focussing our attention on the scrolling and *zoofing* interaction techniques.

Participants

Thirteen (ten male, two female—eleven right-handed) volunteers participated in the experiment. Most had little, or no experience using a stylus or pressure sensitive devices. A $10 gift-certificate was provided as compensation.

Design and Procedure

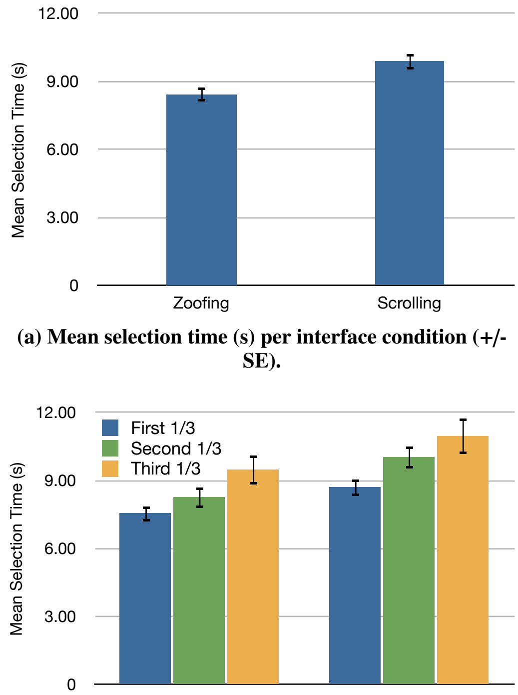
A 2×3 within-participants design was used. The independent variables were *interface* (Scrolling and zoofing) and *selection range* (first third, second third, and last third); the dependent variable was *selection time*. The same methodology used as for experiment one.

The lists were composed of a selection of 1500 artists from the author's personal music collection. Partitioned identically to experiment one.

Results

As with experiment one, we removed outliers from the data set (79 trials discarded, 6.8% of the data).

Analysis of variable showed a significant main effect for interface ($F_{1,12} = 41.2$, $p < 0.001$), with participants performing 15% faster with zoofing (8.4s, s.d. 1.9) than with scrolling (9.9, s.d. 2.2). Like experiment one, *range* showed a significant main effect ($F_{2,23} = 17.3$, $p < 0.001$), but unlike experiment one, there was no *interface*×*range* interaction ($F_{2,24} < 1$). Figure 8 summarises these results, further discussed in 'Future Work'.



(a) Mean selection time (s) per interface condition (+/- SE).

(b) Mean selection time (s) per selection range per interface condition (+/- SE).

Figure 8: Experiment two selection time results.

Subjective Results

Participants strongly preferred *zoofing*, with only one of the thirteen perferring scrolling. All NASA-TLX measures (Table 2) were better for *zoofing*, with significant differences for physical load, frustration, and overall effort.

DISCUSSION

Results show that user performance with *zoofing* is at least comparable to traditional scrolling, and is significantly more efficient than flicking and OrthoZoom-style list interaction. Our second experiment shows that as the size of the list in-

	Scroll	Flick	OZ	zoofing	Sig?
Mental Load	2.0	2.6	4.1	2.7	⋆ ⋆ ⋆
Physical Load	2.7	4.3	3.5	2.4	⋆ ⋆ ⋆
Temporal Demand	2.4	2.8	3.0	2.5	
Performance	4.4	3.3	3.5	4.0	⋆ ⋆ ⋆
Frustration	2.3	3.1	3.7	2.3	⋆⋆
Effort	2.4	3.8	3.9	2.7	⋆ ⋆ ⋆
Preference Rank	1.9	2.9	3.4	1.8	⋆ ⋆ ⋆

Table 1: Experiment one mean NASA-TLX responses (1 low, 5 high) and preference rankings. Friedman χ^2 tests: ⋆ ⋆ ⋆ < 0.001, ⋆⋆ < 0.005, ⋆ < 0.05.

	Scroll	zoofing	Sig?
Mental Load	3.1	2.7	
Physical Load	3.2	2.0	⋆⋆
Temporal Demand	2.8	2.6	
Performance	3.5	3.5	
Frustration	3.2	2.3	⋆
Effort	3.8	2.7	⋆ ⋆ ⋆
Preference Rank	1.9	1.1	⋆ ⋆ ⋆

Table 2: Experiment two mean NASA-TLX responses (1 low, 5 high) and preference rankings. Wilcoxon tests: ⋆ ⋆ ⋆ < 0.001, ⋆⋆ < 0.005, ⋆ < 0.05.

creases, performance with the scrollbar does not scale as well as *zoofing*—significantly out-performing scrolling.

In this section, we will discuss participant feedback, analyse some of the navigation data collected by our experimental software, and discuss some recommendations for some of the parameters that can be adjusted and manipulated for future interfaces and studies.

Participant Feedback

Response from participants to zoofing was positive, with comments that it was "very easy" to understand and manipulate. They liked the natural feel of flicking gestures over short distances in zoofing, but disliked the flick interface for long distances, stating it was "very strenuous". OrthoZoom was generally disliked and criticised as 'very annoying' and 'difficult to control'.

Some participants had problems manipulating the scrollbar on a tablet device—when lifting the stylus from the screen, the action would often drift, causing the scroll thumb to move slightly. This occurred more frequently in the second experiment, where the scroll thumb control was smaller—participants commented that it was 'fiddly'.

They had mixed feelings about the use of pressure to control zoom acceleration. Although generally liked, some felt zoom acceleration was "too sensitive" and that they had difficulty slowing the zoom rate down.

Characterising Scrolling and Flicking

Figure 9 graphs the average vertical distance to a target item over a selection (scaled to the relative average selection) from experiment one. Using a scrollbar, users were able to conduct an open-loop navigation phase that positioned them close to the target item (for example, quickly dragging towards the end of the list to acquire 'ZZ Top'). In contrast, the flicking interface hindered participants' ability to perform an equivalent open-loop phase of movement, leading to a predominantly linear progression towards the target.

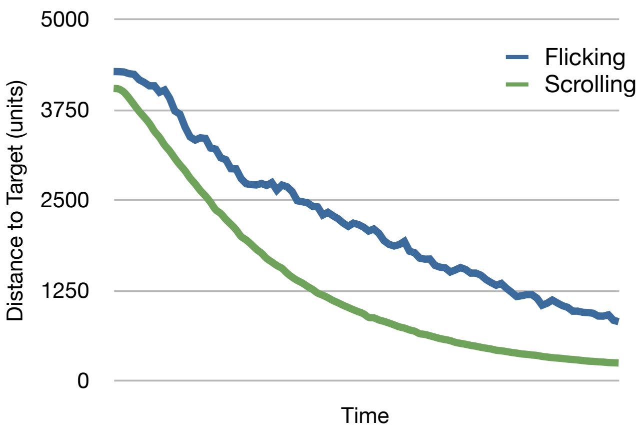


Figure 9: Average vertical distance (units) to target, scaled to the average selection time against scrolling.

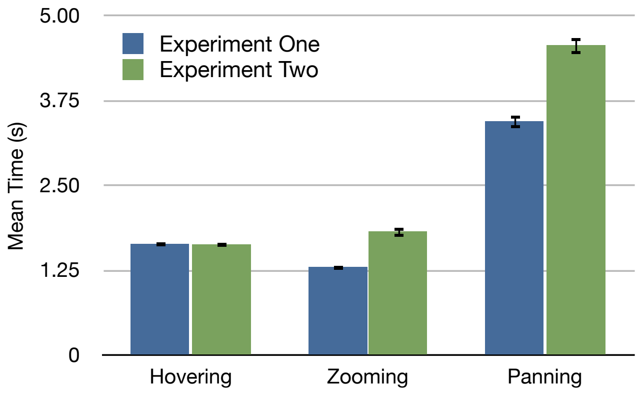


Figure 10: Average time spent (s) per selection, hovering, zooming, and panning in *zoofing* (+/- SE).

Characterising Zoofing

To better understand how pressure and zoom were used, we divided interactions into three phases: (1) *preparation-hovering* above the device surface following the target prompt; (2) *zooming-steering* towards a particular area of the list; and (3) *flicking* at full-zoom to correct the list position. Figure 10 shows the average time spent in each of these phases for both experiment one, and two.

Participants roughly spent the same amount of time deciding where to zoom into the list before initiating that action, regardless of list size. Most time was spent in the *flicking* stage, indicating their accuracy with *zoom steering* and suggesting that users quickly attained a full-zoom before initiating corrective action. However, the logs also show that *zoom steering* was used to refine the zoom centre, with a mean vertical cursor movements of 161 pixels (s.d. 155) in experiment one, and 164 (s.d. 157) in experiment two. Our observations suggest that *zoom steering* was most useful for view refinement in sparse landmarked regions (such as around '*T*', which was heavily populated with items starting with '*The*').

Figure 11 shows the average pressure and zoom distance for experiment one as participants zoomed towards the list for each selection. From the graph, we can see that participants reached and maintained a high level of pressure and zoom acceleration over the zooming portion of their navigation—not utilising variable pressure to control their navigation.

The lack of pressure manipulation may have been due to the over-sensitivity of the stylus or pressure transfer function (either perceived, or real). Alternatively, the linear acceleration towards the list and high average pressure level, may indicate that participants could not accelerate fast enough, with the transfer function limiting their performance.

We believe that a good pressure transfer function needs to embody these characteristics: allowing for a slower, less sensitive motion towards the list for unfamiliar items, but a faster and more sensitive transfer function when users have more precise knowledge about where they want navigate.

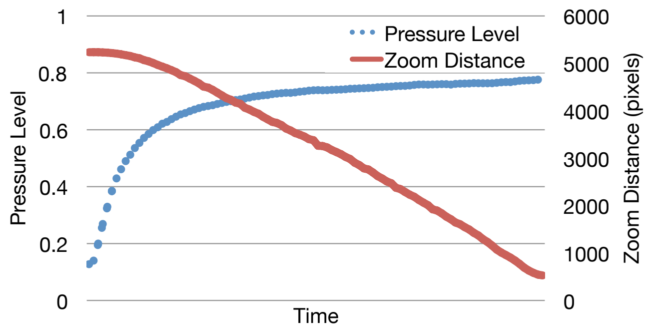


Figure 11: Average pressure level (left) and zoom distance (right) over the course of a single selection with zoofing.

FUTURE WORK

Several issues warrant further investigation. First, we want to understand why performance with *zoofing* was influenced by *selection range* in experiment two more than in experiment one. From a mechanical perspective, performance with *zoofing* should be independent of range. We suspect this may be caused by people finding searches easier early in the alphabet (for example, searching around a, b, c rather than o, p, q) and that this effect is more pronounced in longer lists. Second, we need to investigate improved and active landmark support. All interfaces in our experiment used passive landmarks to depeict letter range locations. Active landmarks would allow users to immediately select a particular data region for scrutiny. Third, we want to explore *zoofing* when used for dataspaces that use multiscale segmentation or 'semantic' zooming. Finally, our experiment used an emulated device in a controlled experimental setting. This is the appropriate experimental norm for research on novel interactive mobile interfaces. Furthermore, no commercial mobile device currently exists that can support our technique: for example, the iPhone does not support pressure interaction, and the Nokia N800 does not have software drivers for its graphics hardware to support *zoofing*. However, with rapid advances in mobile technology we can anticipate new and better devices soon, and we will then evaluate *zoofing* in more ecologically sound experimental conditions.

CONCLUSION

Researchers and developers are seeking natural, subjectively satisfying, and efficient ways to select list items. Prior work on flick scrolling has demonstrated that it is both natural and satisfying, but harms performance in long lists that are often present on mobile devices. By adding pressure-based zooming in *zoofing*, we have substantially improved flick scrolling performance, with evaluations showing comparable performance to scrollbars in lists of 300 items, and 15% better performance in 1500 item lists. Users' emotional response to interfaces is increasingly recognised as a pivotal success factor, and importantly almost all participants perferred *zoofing*, and rated it as less effortful than alternatives.

ACKNOWLEDGEMENTS

This work was partially funded by New Zealand Royal Society Marsden Grant 07-UOC-013.

REFERENCES

Albinsson, P.-A. & Zhai, S. (2003). High precision touch screen interaction, *CHI '03*, ACM, New York, NY, USA, pp. 105–112.

Aliakseyeu, D., Irani, P., Lucero, A. & Subramanian, S. (2008). Multi-flick: an evaluation of flick-based scrolling techniques for pen interfaces, *CHI '08*, ACM, New York, NY, USA, pp. 1689–1698.

Appert, C. & Fekete, J.-D. (2006). OrthoZoom scroller: 1d multi-scale navigation, *CHI '06*, ACM, New York, NY, USA, pp. 21–30.

Benko, H., Wilson, A. D. & Baudisch, P. (2006). Precise selection techniques for multi-touch screens, *CHI '06*, ACM, New York, NY, USA, pp. 1263–1272.

Buxton, W., Hill, R. & Rowley, P. (1985). Issues and techniques in touch-sensitive tablet input, *SIGGRAPH Computer Graphics Newsletter* **19**(3): 215–224.

Cockburn, A., Karlson, A. & Bederson, B. B. (upcoming 2009). A review of overview+detail, zooming, and focus+context interfaces, *ACM Computing Surveys* .

Cockburn, A., Savage, J. & Wallace, A. (2005). Tuning and testing scrolling interfaces that automatically zoom, *CHI '05*, ACM, New York, NY, USA, pp. 71–80.

Furnas, G. W. & Bederson, B. B. (1995). Space-scale diagrams: Understanding multiscale interfaces, *CHI '95*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, pp. 234–241.

Hart, S. & Staveland, L. (1988). Development of NASA-TLX (task load index): Results of empirical and theoretical research, *Human Mental Workload* **1**: 139–183.

Herot, C. F. & Weinzapfel, G. (1978). One-point touch input of vector information for computer displays, *SIGGRAPH Computer Graphics Newsletter* **12**(3): 210–216.

Igarashi, T. & Hinckley, K. (2000). Speed-dependent automatic zooming for browsing large documents, *UIST '00*, ACM Press, New York, NY, USA, pp. 139–148.

Perlin, K. & Fox, D. (1993). Pad: an alternative approach to the computer interface, *SIGGRAPH '93*, ACM Press, New York, NY, USA, pp. 57–64.

Potter, R. L., Weldon, L. J. & Shneiderman, B. (1988). Improving the accuracy of touch screens: an experimental evaluation of three strategies, *CHI '88*, ACM, New York, NY, USA, pp. 27–32.

Ramos, G. & Balakrishnan, R. (2005). Zliding: fluid zooming and sliding for high precision parameter manipulation, *UIST '05*, ACM, New York, NY, USA, pp. 143–152.

Ramos, G., Boulos, M. & Balakrishnan, R. (2004). Pressure widgets, *CHI '04*, ACM Press, New York, NY, USA, pp. 487–494.

Reetz, A., Gutwin, C., Stach, T., Nacenta, M. & Subramanian, S. (2006). Superflick: A natural and efficient technique for long-distance object placement on digital tables, *Graphics Interface*, pp. 163–170.