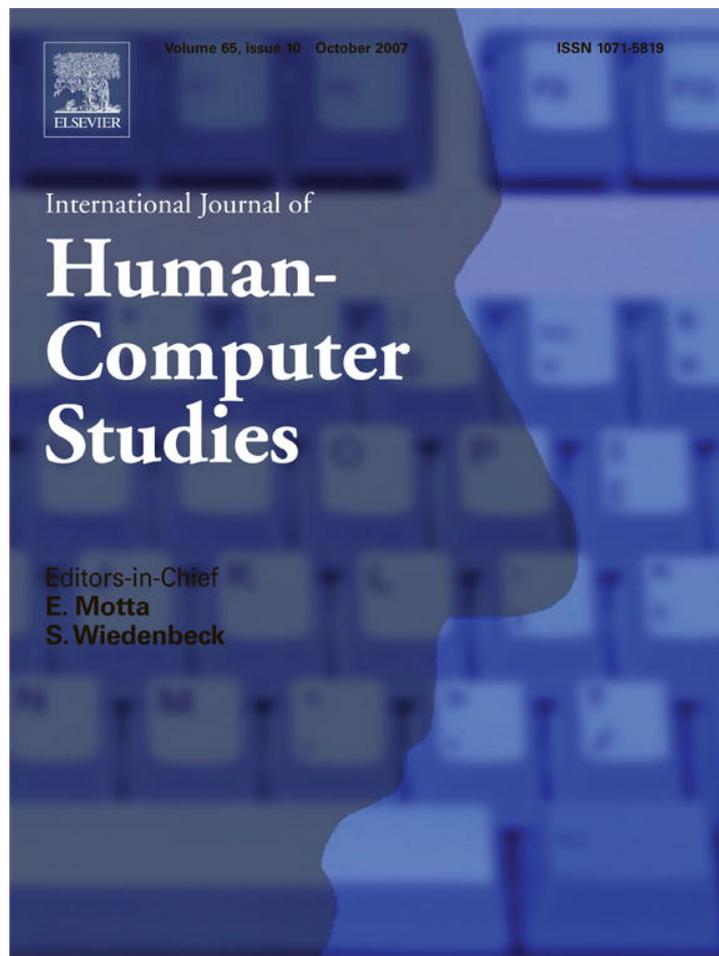


Provided for non-commercial research and education use.
Not for reproduction, distribution or commercial use.



This article was published in an Elsevier journal. The attached copy is furnished to the author for non-commercial research and education use, including for instruction at the author's institution, sharing with colleagues and providing to institution administration.

Other uses, including reproduction and distribution, or selling or licensing copies, or posting to personal, institutional or third party websites are prohibited.

In most cases authors are permitted to post their version of the article (e.g. in Word or Tex form) to their personal website or institutional repository. Authors requiring further information regarding Elsevier's archiving and manuscript policies are encouraged to visit:

<http://www.elsevier.com/copyright>



Input techniques that dynamically change their cursor activation area: A comparison of bubble and cell cursors

Morten Hertzum^{a,*}, Kasper Hornbæk^b

^aComputer Science, Roskilde University, Building 42.2, P.O. Box 260, DK-4000 Roskilde, Denmark

^bUniversity of Copenhagen, Denmark

Received 6 April 2006; received in revised form 2 February 2007; accepted 8 March 2007

Communicated by E. Motta

Available online 24 May 2007

Abstract

Efficient pointing is crucial to graphical user interfaces, and input techniques that dynamically change their activation area may yield improvements over point cursors by making objects selectable at a distance. Input techniques that dynamically change their activation area include the bubble cursor, whose activation area always contains the closest object, and two variants of cell cursors, whose activation areas contain a set of objects in the vicinity of the cursor. We report two experiments that compare these techniques to a point cursor; in one experiment participants use a touchpad for operating the input techniques, in the other a mouse. In both experiments, the bubble cursor is fastest and participants make fewer errors with it. Participants also unanimously prefer this technique. For small targets, the cell cursors are generally more accurate than the point cursor; in the second experiment the box cursor is also faster. The cell cursors succeed in letting participants select objects while the cursor is far away from the target, but are relatively slow in the final phase of target acquisition. We discuss limitations and possible enhancements of input techniques with activation areas that contain multiple objects. © 2007 Elsevier Ltd. All rights reserved.

Keywords: Input techniques; Interaction techniques; Cell cursors; Touchpad; Bubble cursor; Box cursor; Pie cursor

1. Introduction

Pointing operations are intrinsic to graphical user interfaces. Consequently pointing devices and the associated input techniques are of considerable importance to users' overall task performance and to their subjective experience of systems. Recently, Grossman and Balakrishnan (2005) presented the bubble cursor, which elegantly extended previous work on area cursors and performed very well in a user study. The bubble cursor (Fig. 1) dynamically adjusts the radius of the cursor activation area such that it always contains exactly one object. Thus, to select the closest object the user need merely click, no cursor movement is required. To select any other object the user need only move the cursor closer to the target object than to any other object, making selection easier whenever there is space between objects.

Grossman and Balakrishnan (2005) evaluated the bubble cursor for mouse-operated cursors and scenes in which the effective width of targets was formally controlled through the placement of non-target objects, so-called distracters. In the experiments presented in this paper we evaluate the usability of bubble cursors when operated with a mouse as well as a touchpad, and we base our evaluation on scenes with randomly placed distracters. Such replication is relatively rare in Human–Computer Interaction but necessary to validate important results and assess how they transfer to related but not identical circumstances.

As selection of the closest object is optimal with the bubble cursor, improvements over the bubble cursor must provide more efficient ways of reaching beyond the closest object. We suggest two new input techniques, the box cursor and the pie cursor, which extend the cursor activation area until it contains not only the object closest to the cursor but a small set of objects in the vicinity of the cursor. We expect that this will further reduce the need for moving the mouse in the initial part of pointing operations,

*Corresponding author. Tel.: +45 4674 3077; fax: +45 4674 3072.

E-mail address: mhz@ruc.dk (M. Hertzum).

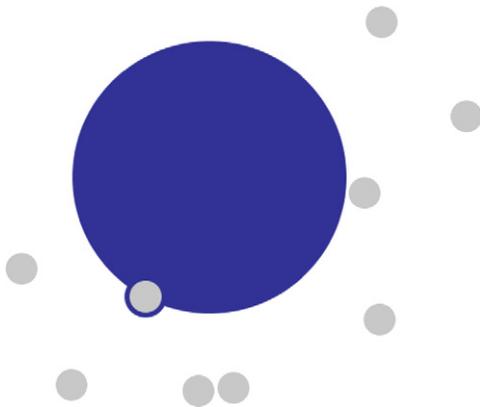


Fig. 1. The bubble cursor.

particularly in densely populated displays, at the expense of an overhead during the final part of pointing operations. The final selection of the target from the set of objects within the cursor activation area can be accomplished in several ways, of which we explore two: indicating the *position* of the target within the cursor activation area and indicating the *direction* from the centre of the cursor activation area toward the target. To explore these two possibilities, we conduct two experiments.

In our first experiment the input techniques are operated with a touchpad and for the box and pie cursors the final selection of the target is accomplished by tapping in the touchpad region that corresponds to the position of the target in the cursor activation area. This is similar to the TouchGrid (Hertzum and Hornbæk, 2005), which replaces moving the cursor through dragging a finger on the touchpad with tapping in different regions of the touchpad. Whereas the TouchGrid recursively maps touchpad regions to smaller display regions until a target is uniquely specified, we only use the TouchGrid technique for the final selection of the target.

In our second experiment the input techniques are operated with a mouse and for the box and pie cursors the final selection of the target is accomplished by gesturing with the mouse in the direction of the target. The final selection resembles object pointing (Guiard et al., 2004), which extrapolates the direction of the initial cursor movement to skip empty space between objects. Again, whereas object pointing recursively applies this process to skip from object to object until the target is reached, we merely use the directional gesture for the final selection of the target from the small set of objects in the vicinity of the cursor.

Thus, this study evaluates the usability of the bubble cursor and compares the bubble cursor with two input techniques that explore ways of extending it. In the following, we review related work on input techniques (Section 2), explain the design of the box and pie cursors (Section 3), provide empirical evidence about the use of bubble, box, pie, and conventional point cursors based on our two experiments (Sections 4–9), and discuss the

implications of our experimental findings (Section 10). The discussion revolves around the prospects of shifting a larger portion of pointing operations to a final selection of the target from a cursor activation area containing multiple objects.

2. Related work

The bubble cursor is a recent result of the extensive work on pointing devices and input techniques. While mice and, to some extent, trackballs are generally faster than joysticks and touchpads in experimental comparisons (Card et al., 1978; Epps, 1986; MacKenzie et al., 1991, 2001; Cohen et al., 1993; Accot and Zhai, 1999), other factors may impact the choice of pointing device in practical situations. For example, mice require a flat surface for their operation whereas touchpads remain usable in cramped spaces with unstable support and have become a standard component of laptop computers. The basic functionality of pointing devices has been refined, enhanced, and modified by various input techniques, most of which can be modelled with Fitts's law.

2.1. Fitts's law

Fitts's law (Fitts, 1954) concerns basic aiming movements such as moving a cursor to a target by means of a mouse, a touchpad, or another device with which continuous cursor movement is achieved by continuous finger and/or hand movement. During such pointing movements users initially perform a large and fast sub-movement to get close to the target. The remaining part of pointing movements consists of acquiring the target and involves one or several slower and more precise sub-movements to position the cursor over the target. While the initial sub-movement is governed by the distance to the target (D), the subsequent sub-movements are governed by the width of the target (W). Fitts (1954) showed that the time needed to perform a pointing movement could be expressed in terms of these two variables:

$$MT = a + b \times \log_2 \left(\frac{D}{W} + 1 \right), \quad (1)$$

where a and b are empirical constants determined by linear regression. Whereas the logarithmic term characterizes the difficulty of the pointing task, a and b depend on the pointing device and have been used extensively to compare such devices (MacKenzie, 1992; Guiard and Beaudouin-Lafon, 2004). Fitts's law suggests that more efficient pointing operations can be achieved by input techniques that increase target width, reduce the distance to targets, or both (Balakrishnan, 2004).

2.2. Expanding the size of targets

One way of expanding the effective target width is to increase the size of the cursor activation area. Area cursors

(Zhai et al., 1994; Kabbash and Buxton, 1995) replace the one-pixel activation area of the conventional point cursor with a larger, but fixed-size region. Worden et al. (1997) find that area cursors are faster than point cursors, particularly for small targets. However, the fixed size of the cursor activation area is arbitrarily set and whenever the cursor activation area contains more than one object, area cursors revert to operating as point cursors.

The bubble cursor (Grossman and Balakrishnan, 2005) extends area cursors by dynamically adjusting the size of the cursor activation area based on the proximity of surrounding objects. The activation area is circular with a radius equal to the distance to the object that is currently closest to the cursor. That is, the cursor activation area always contains exactly one object, and pointing at an object merely involves getting closer to this object than to any other object. Grossman and Balakrishnan (2005) note that formally this consists of expanding objects until their effective size is equivalent to their corresponding region in a Voronoi diagram. In discussing Voronoi diagrams, Gold (1994) made the same observation and suggested their potential for making pointing operations more efficient: “it is not necessary to... point precisely at a[n] existing object in order to select it for use... Merely pointing nearer to that object than to any other will suffice. (This is the same as pointing anywhere within an object’s bubble.)”

In Grossman and Balakrishnan’s (2005) implementation the bubble cursor is visually rendered as a semitransparent circular area corresponding to its activation area (Fig. 1). This circular area always intersects the closest object but in order not to intersect the second closest object the circular area may not fully contain the closest object. In this case the bubble cursor is morphed by extending it with a second circular area that expands from the intersection and fully envelops the closest object.

Empirical testing shows that the bubble cursor is faster than both a point cursor and object pointing, and has a lower error rate (Grossman and Balakrishnan, 2005). Specifically, the bubble cursor is faster in both a condition with no distracters and conditions with up to 200 distracters. Grossman and Balakrishnan (2005) also find that the bubble cursor can be accurately modelled with Fitts’s law if the actual target width is replaced by its effective width. That is, selection with the bubble cursor is as fast as if a point cursor was used and targets had an actual width equal to the maximum distance from which the target can be selected with the bubble cursor.

Two additional types of input technique make pointing operations more efficient by increasing target size. The first type visually expands targets close to the cursor. McGuffin and Balakrishnan (2002) find that users benefit from target expansion even if it does not occur until the cursor has travelled 90% of the distance toward the target. Zhai et al. (2003) replicate this finding and extend it to situations in which users do not know in advance whether targets expand when the cursor gets close to them. Cockburn and Firth (2003) investigate selection of small targets and find

that expanding targets are selected faster than unexpanding targets but that goal crossing (Accot and Zhai, 2002) is even faster.

A second type of technique enlarges targets by dynamically adjusting the control:device (C:D) gain to make the cursor stick to objects. Blanch et al. (2004) term this semantic pointing. Semantic pointing increases target size in motor space without the potentially distracting effects of dynamically expanding and shrinking objects in display space. Blanch et al. (2004) find that semantic pointing is faster and more accurate than pointing operations performed at a constant C:D gain. Keyson (1997) finds that making objects sticky by dynamic C:D gain adaptation improves performance and is virtually unaffected by the absence or presence of distracters. In the absence of distracters, performance is further improved by using tactile feedback to enhance object stickiness.

2.3. Reducing the distance to targets

Two types of input technique make pointing movements more efficient by reducing the distance to targets. The first type accelerates cursor movement toward targets. Object pointing (Guiard et al., 2004) exploits that pointing operations are performed to select objects and that the cursor can thus skip empty spaces between objects. This drastically reduces the amount of input-device movement necessary to move the cursor to a target compared to conventional pointing. Object pointing skips empty space by determining the direction of cursor movement as the cursor leaves an object and extrapolating this movement until it reaches the first object located in that direction. Guiard et al. (2004) find that object pointing is faster than conventional pointing for more difficult pointing tasks. The level of task difficulty at which object pointing becomes the faster technique increases drastically with the number of distracters, implying that object pointing is mainly competitive in sparsely populated displays. Grossman and Balakrishnan (2005) find that object pointing is consistently slower than conventional pointing.

A second type of technique reduces the distance to targets by temporarily moving candidate targets closer to the cursor. With drag-and-pop and drag-and-pick (Baudisch et al., 2003) candidate targets are determined by extrapolating the initial direction of cursor movement and choosing as candidates the targets that lie in a sector extending a small angle to either side of this initial direction. Empirical testing shows that drag-and-pop speeds up long pointing operations on wall-size displays but also indicates that the direction of users’ initial cursor movement is frequently off target, forcing users to abort and restart their pointing movement to get the target included among the candidate targets. The vacuum (Bezerianos and Balakrishnan, 2005) extends drag-and-pick by enabling users to dynamically adjust the direction and width of the sector that determines the candidate targets and by facilitating operations that require multiple

manipulations of the target. Empirical testing shows that the vacuum performs better than drag-and-pick in the presence of distracters.

2.4. Cell cursors

Cell cursors replace the continuous pointing movements that can be modelled with Fitts's law with pointing movements that consist of a sequence of distinct inputs. For their implementation of cell cursors, Hertzum and Hornbæk (2005) find that selection time increases linearly with the number of inputs (N):

$$T = a + b \times N, \quad (2)$$

where a and b are empirical constants determined by linear regression. This corresponds to the model Card et al. (1978) devised for keyboard-operated pointing techniques such as arrow keys. Cell cursors are, however, characterized by three basic ideas that distinguish them from using arrow keys for pointing. First, a cell cursor overlays a region of the display and divides that region into a number of cells. A cell cursor is often a grid but other layouts are also possible, for example pie layouts. Second, each cursor cell is mapped to a discrete user input, for example to regions of a touchpad. Third, selection is accomplished by providing the input that specifies the cell containing the target. If the specified cell contains multiple objects, the cursor is recursively applied to the cell. After a sequence of one or more user inputs the target is the only object in the specified cell and selection has been accomplished.

Cell cursors relax the requirements for pointing precision in display space as well as motor space. In display space, cell cursors exploit that users need not hit the target but merely specify a cell sufficiently small to disambiguate which object is being selected. In motor space, the same inputs (e.g., taps in different touchpad regions) are used for all specifications of cells but recursively mapped to smaller display regions. A limitation of cell cursors is that selection time depends on the number of objects in the display. As the number of objects increases, more recursions are needed to select a target.

Two studies investigate the use of cell cursors for general-purpose pointing. The TouchGrid (Hertzum and Hornbæk, 2005) maps a cell cursor to touchpad regions and, thereby, replaces moving the cursor through dragging a finger on the touchpad with tapping in different touchpad regions. The TouchGrid is a grid with at most three rows and three columns. Depending on the number and spatial distribution of the objects within the cursor area individual cells may or may not contain an object. If entire rows or columns are empty they are dynamically removed. The cell cursor is visually represented by a grid showing the current division of the cursor into cells. This grid defines the division of the touchpad into regions. Thus, if the grid currently consists of three rows but only one column, the user merely needs to tap the lower third of the touchpad to

select the bottom row; the horizontal position of the tap is of no consequence. Empirical testing shows that the TouchGrid is faster than a conventional touchpad for tasks that require one tap, marginally faster for two-tap tasks, and neither faster nor slower for three-tap tasks. This suggests that the TouchGrid may mostly be suited for sparsely populated displays. Further, selection of small targets is consistently faster with the TouchGrid, but error rates are consistently higher.

Dai et al. (2005) describe a cell cursor that is visually very similar to the TouchGrid but operated by speech input rather than touchpad taps. The cursor always consists of 3×3 cells, and the user recursively specifies cells by giving their numbers as spoken commands. Cells are numbered left to right and top to bottom. Empirical testing shows that selection times with the cell cursor correlate well with the number of spoken commands and are largely independent of the distance to targets. Dai et al. (2005) informally compare performance data for their cell cursor with previous studies of conventional direction-based techniques for speech-controlled pointing and find that it is both faster and more accurate.

Cell cursors have also been used for map navigation on cell phones (Robbins et al., 2004) and text input to television-based appliances (Ingmarsson et al., 2004). In these cases the 3×3 arrangement of the digit keys 1 through 9 on the cell phone or remote control has recursively been mapped to regions of the map or subsets of the character set.

3. Two instances of cell cursors: the box cursor and the pie cursor

Compared to conventional pointing, the box and pie cursors reduce the distance users have to move the cursor to select an object and at the same time relax the requirements for precision during the final selection of both small and large objects. In this sense the box and pie cursors are similar to the bubble cursor. Contrary to the bubble cursor the final selection of objects is extended by distinguishing between taps in different touchpad regions (experiment one) or mouse gestures in different directions (experiment two). This way the user may select the object closest to the cursor, as in the bubble cursor, or one of the other objects close to the cursor. Contrary to previous cell cursors, pointing with the box and pie cursors does not involve recursive selection of still smaller cells. Rather, users move the cursor toward an object in the same way as with a conventional point cursor. We do this because Hertzum and Hornbæk (2005) find that performance with the TouchGrid decreases as the number of objects increase.

Below we describe the variant of the box and pie cursors used in experiment one. Experiment two evaluates a revised variant of the box and pie cursors, which will be described in Section 7.3.

3.1. Box cursor

The box cursor has an activation area divided into rectangular cells, each containing one object. When the cursor is moved, the size of the activation area and its division into cells change in a way determined by the objects closest to the cursor. The current arrangement of cells is mapped onto the touchpad, which is thereby divided into a corresponding arrangement of regions. Users operate the box cursor by dragging a finger along the touchpad surface to move the cursor until the target object is within its activation area and then tapping in the touchpad region that corresponds to the cell containing the target.

The arrangement of cells that constitutes the box cursor is restricted to at most 3×3 cells. Thus in tapping the touchpad the user must distinguish between at most three vertical and three horizontal regions. Often the position of screen objects relative to each other prevents the construction of a full 3×3 arrangement of cells and the user has to distinguish between fewer regions in tapping the touchpad. Fig. 2 shows an example of a box cursor and the touchpad region that must be tapped to select the black target. When the box cursor contains a centre cell the size of the corresponding touchpad region is enlarged slightly, relative to the surrounding rows and/or columns. Based on pilot testing this enlargement was set to 15%. In the absence of a centre cell the touchpad is divided into rows of equal height and columns of equal width, irrespective of the relative size of the cells that represent the box cursor on the screen.

3.2. Pie cursor

The operation of the pie cursor is similar to that of the box cursor but their visual representations differ, see Fig. 3. The pie cursor is represented by arrows radiating from a common centre. Each arrow points toward a selectable object, which is further visually indicated by a circle enclosing the object. The arrows are mapped to pie-sliced touchpad regions, each pie slice including the direction given by the arrow and extending halfway toward the two neighbouring arrows. Thus, the pie cursor has a circular activation area divided into pie slices, each containing one

object. The user moves the pie cursor by dragging a finger over the touchpad surface, as in conventional pointing. Selection is accomplished by tapping in the touchpad region that corresponds to the arrow pointing toward the target. In addition, if the centre of the pie cursor is within 12 pixels of an object, that object is selected by tapping the centre region of the touchpad. No arrow points toward such an object.

The objects selectable at any one point in time are those closest to the centre of the pie cursor, skipping only those objects that cannot be assigned a pie slice subtending an angle of at least 60° with a minimum of 18° on either side of the direction given by the arrow. This minimum size of the pie slices was determined through pilot testing and limits the number of pie slices to a maximum of 6.

4. Experiment one

The first experiment seeks to understand the benefits and relative performance of box, bubble, and pie cursors empirically by comparing them to a standard point cursor.

4.1. Participants

Eight participants (three female, five male) took part in the experiment. Participants' age ranged from 13 to 67 years with an average of 32 years. All participants had

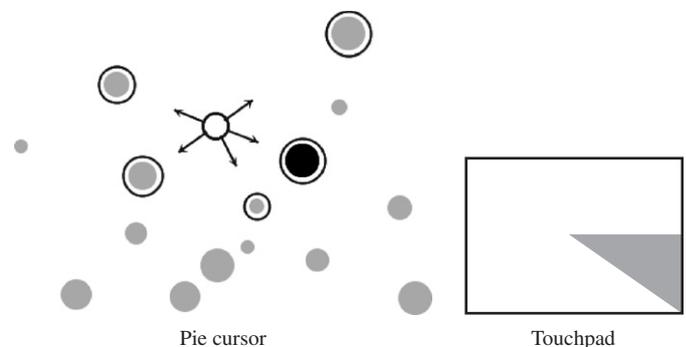


Fig. 3. Example of pie cursor in experiment one. The black target to the lower, right of the pie cursor (left) is selected by a tap in the indicated touchpad region (right).

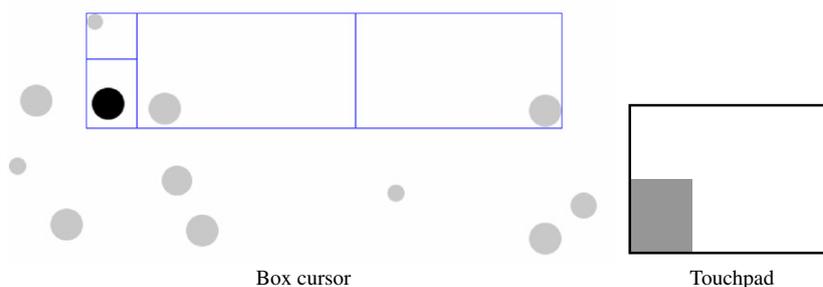


Fig. 2. Example of box cursor in experiment one. The black target in the lower, left cell of the box cursor (left) is selected by a tap in the indicated touchpad region (right).

normal or corrected-to-normal vision. Participants rated their experience using touchpads on laptop computers to an average of 3.1 on a scale from 1 (none) to 5 (very experienced); all participants had some touchpad experience. Seven participants operated the touchpad with their right hand, one with her left hand.

4.2. Apparatus

The experiment was conducted on a 1.86 GHz HP laptop with a 68 mm × 39 mm Synaptics touchpad. The display resolution was set to 1024 × 768 on a 15" screen. For all input techniques mouse acceleration was deactivated, and the C:D gain was set to the middle value in the control panel of Windows XP.

4.3. Input techniques

Four input techniques were used. The *box cursor* and the *pie cursor* worked as explained in Section 3. The *bubble cursor* was implemented following the description in Grossman and Balakrishnan (2005). The *point cursor* simply displayed a non-enhanced cursor; that is, a standard arrow pointer. All four input techniques were operated with a touchpad.

4.4. Tasks

Each task consisted of consecutive selections of 11 targets. Selection of the first target marked the start of the task, leaving ten selections for which time and other measurements were obtained. Targets were located in a scene of randomly placed objects having a certain size, distance between targets, and density of distracters. Each scene used one level of each of target sizes, distances, and density of distracters.

We used two *target sizes*: small (8 pixels) and large (32 pixels). The reason for varying target size is to investigate its impact on the performance of the bubble and cell cursors. For the point cursor, Fitts's law (Fitts, 1954) quantifies the presumed influence of varying size on movement time.

Three *distances* between targets were used: short (128 pixels), medium (320 pixels), and long (800 pixels). The reason for varying the distance between targets is that it may impact performance differently between input techniques. In addition, the distance to target will also affect the number of objects the cursor passes and this may, in turn, impact performance.

We also varied the *distracter density* of each scene. While density should not affect the point cursor, it is likely to affect the other input techniques. The values chosen for density are few (10 distracters), medium (20 distracters), and many (60 distracters). These values were chosen informally, but are comparable to the values chosen by Guiard et al. (2004). We excluded very densely packed scenes because they appeared biased against the bubble

cursor. Note that the number of distracters include targets to be selected before or after the current target; with few distracters no objects beyond the 11 targets are added to the scene, leaving one target to be selected and 10 objects that serve as distracters.

We chose against more formally controlling other factors that could impact performance, as was done in a previous study of the bubble cursor (Grossman and Balakrishnan, 2005). In that study distracters were placed at a certain distance from the target, controlling its effective width. In addition, one of these distracters was always placed on a straight line from the start object to the target (see Grossman and Balakrishnan, 2005, p. 287). In our study this would systematically disfavour the bubble cursor because the target would never be the object closest to the cursor at the start of a trial but could be within the cursor activation area of the box and pie cursors. We therefore chose random placement of the distracters. We chose against the multidirectional selection task of ISO 9241-9 (ISO, 2000) because it never involves distracters between a target and the initial cursor position.

The current target was shown as a red circle and distracters were shown as light blue circles. Upon successful selection of the current target it turned light blue and the next target became red. Targets were placed randomly, apart from ensuring that the target-to-target distance was exactly the level of distance used in the particular scene. Distracters were placed entirely at random, except those distracters that also worked as targets. The relatively long distances between targets ensure that most of the screen has to be visited. Consequently, all or most of the distracters would impact performance. See Fig. 4 for an illustration of the tasks used.

4.5. Procedure

Initially, participants were explained the purpose of the experiment, filled out a background questionnaire, and were introduced to the four input techniques. Participants were instructed to work fast but no faster than still maintaining a high level of accuracy. Participants were also instructed to use only one hand for operating the touchpad and they were offered a hand rest, which all of them used. Then, participants spent an average of 12 min familiarizing themselves with the input techniques by solving training tasks similar to the experimental tasks. Next, they used one of the input techniques to perform 180 trials and then filled out a questionnaire about their satisfaction with that input technique. Having performed with one input technique, participants proceeded to make selections and fill out questionnaires for the three remaining input techniques. After filling out a questionnaire participants rested for a moment and then performed the 180 trials with the next input technique in an unbroken sequence. Finally, participants ranked the input techniques in order of preference. For each participant, the entire experiment lasted about an hour.

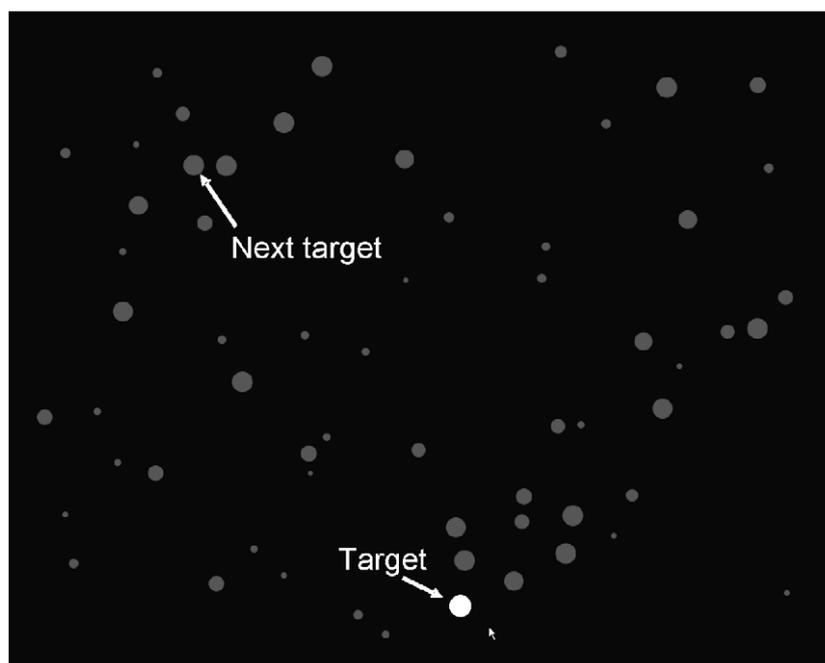


Fig. 4. An example of a scene with large targets, long distances between targets, and many distracters. In the experiment the scene occupied the full 15" screen, and the current target was red. The arrows and the text are added as illustration only.

4.6. Design

A within-subjects design was used. The independent variables were input technique (box, bubble, pie, point), target size (small, large), target distance (short, medium, long), and distracter density (few, medium, many). The design was fully crossed. Each participant used an input technique to perform 10 trials with each combination of densities, distances, and sizes. The 18 combinations of densities, distances, and sizes appeared in a random sequence. The order in which participants used the input techniques was balanced by means of a pair of Latin squares. In total, the experiment comprised:

$$8(\text{participants}) \times 4(\text{input techniques}) \times 2(\text{target sizes}) \\ \times 3(\text{target distances}) \times 3(\text{distracter densities}) \\ \times 10(\text{repetitions}) = 5760 \text{ trials.}$$

4.7. Dependent measures

We measured error rate, trial completion time, subjective satisfaction, and preference. *Error rate* was measured as the trials for which participants missed a target by clicking in an empty part of the screen or selecting a wrong object. *Trial completion time* was measured from the selection of one target to the selection of the next target. *Subjective satisfaction* was measured by 12 questions from a questionnaire based on the ISO 9241-9 standard (ISO, 2000), as modified by Douglas et al. (1999). *Preference* was measured as participants' rank-ordering of the input

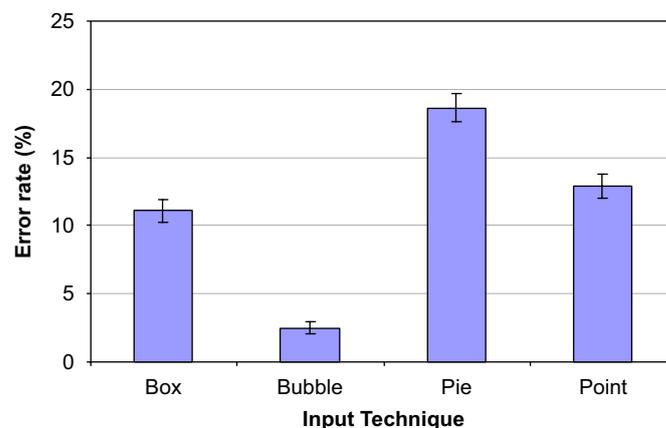


Fig. 5. Average error rates in experiment one ($N = 5664$); error bars show standard error of the mean.

techniques. In addition, we logged all touchpad events, allowing us to derive several measures of touchpad use. All analyses of the dependent measures were done using repeated-measures analysis of variance.

5. Results of experiment one

5.1. Error rate

Before analysing error rates, we removed 96 outlier trials, which were more than three inter-quartile ranges above the upper quartile in trial completion time. Fig. 5 summarizes the error rates in the remaining 5664 trials. We find a significant difference between input techniques,

$F(3, 5) = 20.85, p < .001$. Linear contrasts show that the bubble cursor has a lower error rate than any other input technique; on average, the bubble cursor has 2.5% errors while the other input techniques have 14.3% errors. Further, the box and point cursors performed similarly and had significantly lower error rates than the pie cursor.

We find a main effect of size on error rates, $F(1, 7) = 7.82, p < .05$, and an interaction between size and input technique $F(3, 5) = 20.56, p < .01$. Unsurprisingly, the largest difference seems to be that the point cursor leads to more errors with small targets ($M = .20, SD = .40$) compared to large targets ($M = .06, SD = .23$), making the box cursor significantly more accurate than the point cursor for small targets ($p < .05$). Distance to target and distracter density do not influence error rates, nor do they interact with input technique.

5.2. Trial completion time

Fig. 6 summarizes the trial completion times for the 5030 error-free, non-outlier trials in the experiment. We find a significant difference between input techniques, $F(3, 5) = 35.92, p < .001$. Linear contrasts show that the bubble cursor ($M = 1775$ ms, $SD = 646$) is significantly faster than the three other input techniques—on average the bubble cursor is 32% faster, or about 830 ms. We find no differences among the box ($M = 2549$ ms, $SD = 946$), pie ($M = 2690$ ms, $SD = 979$), and point ($M = 2475$ ms, $SD = 844$) cursors.

We find effects of both distance and size on trial completion times. A lower distance to target gives a significantly lower trial completion time compared to a higher distance to target, $F(2, 6) = 92.86, p < .001$. For target size, we find a significant average difference of 372 ms between small and large targets, $F(1, 7) = 115.06, p < .001$. Further, size and input technique interact, $F(3, 5) = 22.93, p < .001$. The point cursor is much slower for small targets ($M = 2910$ ms, $SD = 869$) than for large targets ($M = 2120$ ms, $SD = 630$), whereas trial completion times for the other input techniques are less sensitive to target size.

We also find an effect of density on trial completion time, $F(2, 6) = 15.03, p < .01$. Average trial completion times are lower when density is lower. In addition, we find an interaction between input technique and density, $F(3, 5) = 28.44, p < .001$. Contrary to the point cursor, the other input techniques have higher trial completion times for high-density scenes ($M = 2456$ ms, $SD = 1011$) compared to low-density scenes ($M = 2215$ ms, $SD = 894$), see Fig. 6c.

5.3. Distance to target when selection occurs

We calculated the average distance from which targets are selected, excluding outliers and trials with errors. There is a significant difference between input techniques in how far from the centre of targets selection occurs,

$F(3, 5) = 595.06, p < .001$. Linear contrasts show a distinct ordering of the input techniques; listed in decreasing order of distance the input techniques are the box cursor ($M = 111$ pixels, $SD = 81$), pie cursor ($M = 85$ pixels, $SD = 65$), bubble cursor ($M = 55$ pixels, $SD = 51$), and point cursor ($M = 5$ pixels, $SD = 4$). All pair-wise comparisons between input techniques are significant.

5.4. Time in approach and selection phases

Trial completion time can be split into time required to approach the target and to select it. We define the approach phase as lasting from the beginning of a trial to the point in time where the mouse is moved less than 5 pixels per second for the remainder of the trial; the selection phase comprises the remainder the trial. The threshold of 5 pixels/s was selected because no movement (i.e., 0 pixels/s) seemed too strict a delimiter of the two phases; in particular a small movement on the touchpad when beginning to make a click could have a large effect. Fig. 7 shows the average times for the approach and selection phases of non-outlier, error-free trials.

For the approach phase, we find a significant effect of input technique, $F(3, 5) = 19.23, p < .01$. Linear contrasts show that the box and bubble cursors have shorter approach phases than the pie cursor, which in turn has a shorter approach phase than the point cursor. These numbers are slightly surprising, as the pie cursor enables selection of a target from farther away compared to the bubble cursor (see Section 5.3).

For the selection phase, we also find a significant effect of input technique, $F(3, 5) = 49.02, p < .01$. Linear contrasts show that the bubble cursor has the shortest selection phase ($M = 448$ ms, $SD = 285$), followed by the point cursor ($M = 591$ ms, $SD = 388$). The pie and box cursors are significantly slower with selection phases of 1150 ms ($SD = 647$) and 1228 ms ($SD = 742$), respectively. The selection phase of the bubble cursor is 62% shorter than for the box and pie cursors and appears to be the main reason for the difference in trial completion times.

5.5. Modelling trial completion times

Trial completion times for the input techniques can be modelled with Fitts's law (Fitts, 1954; MacKenzie, 1992). Initially, we disregarded any effects of distracters and used regression on the means of error-free, non-outlier trials for the six combinations of size and distance (giving six index-of-difficulty values). For the point cursor, a good regression model is obtained, $r^2 = .96$. For the other input techniques the fits are less convincing, namely $r^2 = .78$ (box), $r^2 = .75$ (bubble), and $r^2 = .84$ (pie). A slightly more refined attempt at modelling trial completion times uses the mean of error-free, non-outlier trials for the combinations of index of difficulty and distracter densities (i.e., 18 values for each input technique). Stepwise linear regression shows that adding density to the models significantly improves

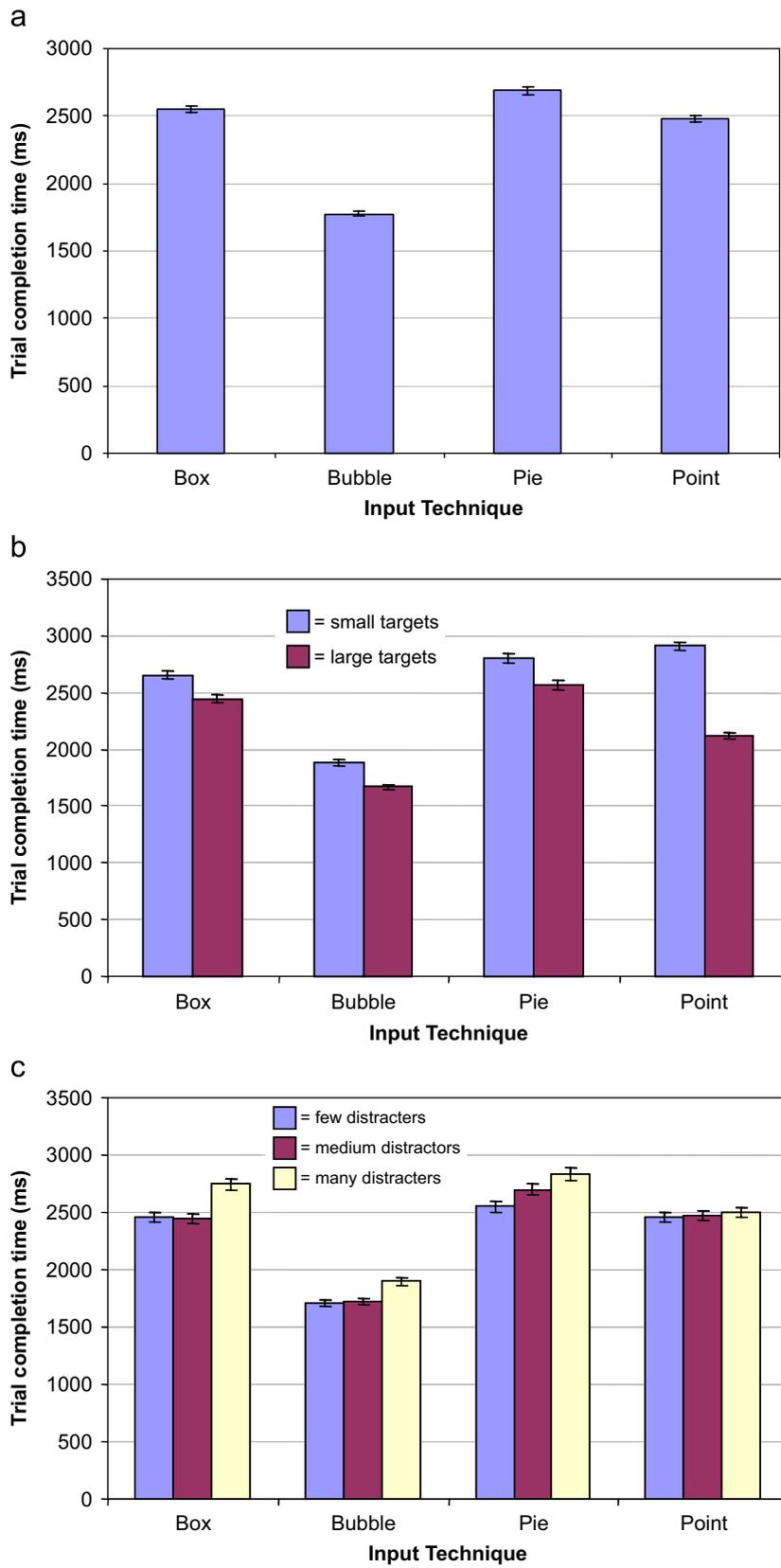


Fig. 6. Average trial completion times in experiment one for (a) input techniques, (b) input techniques given different target sizes, and (c) input techniques given different distracter densities; error bars show standard error of the mean ($N = 5030$).

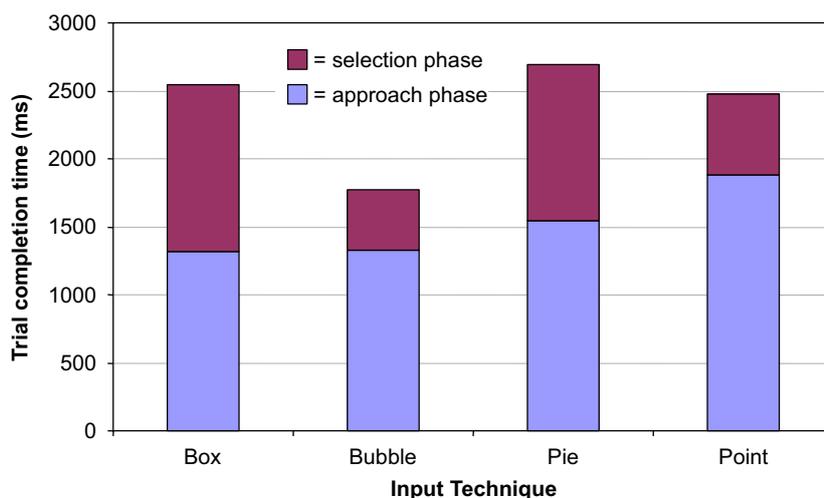


Fig. 7. Time spent in approach and selection phases ($N = 5030$).

Table 1
Participants' subjective ratings of the input techniques in experiment one ($N = 8$); significant differences between input techniques are marked with asterisks

Question	Box cursor		Bubble cursor		Pie cursor		Point cursor	
	<i>M</i>	<i>SD</i>	<i>M</i>	<i>SD</i>	<i>M</i>	<i>SD</i>	<i>M</i>	<i>SD</i>
1. Overall, the cursor was** (1: very difficult to use–5: very easy)	3.13	.99	4.88	.35	2.38	1.06	3.25	1.28
2. General comfort * (1: very uncomfortable–5: very comfortable)	3.13	.64	4.63	.52	2.38	1.30	2.63	1.06
3. Smoothness during operation was ** (1: very rough–5: very smooth)	2.88	1.13	4.50	.53	2.75	.89	3.00	1.31
4. Accurate pointing was ** (1: easy–5: difficult)	2.88	.99	1.25	.46	2.88	1.55	3.63	1.51
5. Force required for actuation was (1: appropriate–5: inappropriate)	2.63	1.19	1.50	.76	3.00	1.20	2.25	1.04
6. Mental effort required for operation was (1: appropriate–5: inappropriate)	2.63	1.19	1.25	.71	3.38	1.41	1.75	1.04
7. Physical effort required for operation was (1: appropriate–5: inappropriate)	2.00	1.07	1.63	.74	2.88	1.25	2.88	1.46
8. Finger fatigue (1: none–5: very high)	2.25	1.16	2.13	1.13	3.00	1.31	3.13	1.36
9. Wrist fatigue (1: none–5: very high)	2.50	1.51	2.50	1.20	3.00	1.41	3.38	1.60
10. Arm fatigue (1: none–5: very high)	1.88	1.13	2.25	.89	2.38	1.41	2.75	1.39
11. Shoulder fatigue (1: none–5: very high)	1.38	.74	1.88	.83	1.88	.99	2.13	1.36
12. Neck fatigue (1: none–5: very high)	1.63	.74	2.00	1.41	2.25	1.39	2.38	1.41

* $p < .05$, ** $p < .01$.

their fit for the box and pie cursors ($p < .05$) and marginally improves the fit for the bubble cursor ($p = .09$). Using *ID* to signify the index of difficulty (i.e., $\log_2(D/W + 1)$) and *DD* to signify the distracter density (with levels of 10, 20 and 60), the model for the box cursor is $MT = 1576 \text{ ms} + 191 \text{ ms} \times ID + 7 \text{ ms} \times DD$, and for the pie cursor it is $MT = 1561 \text{ ms} + 232 \text{ ms} \times ID + 5 \text{ ms} \times DD$. For comparison, the model for the point cursor is $MT = 920 \text{ ms} + 357 \text{ ms} \times ID$, and for the bubble cursor it is $MT = 704 \text{ ms} + 224 \text{ ms} \times ID$.

5.6. Subjective assessments and preference

Table 1 summarizes the answers to the questionnaires administered to participants. An overall multivariate analysis shows a significant difference between input techniques, Wilks's $\lambda = .20$, $F(3, 5) = 6.59$, $p < .05$. With the experiment-wide error thus protected, we perform analyses of variance for the individual questions. These

individual analyses of variance show significant differences for four of the 12 questions. Linear contrasts show that in all four cases the bubble cursor is rated more favourably than one or more of the three other input techniques, see Table 1.

Participants' preference rankings of the input techniques show a unanimous preference for the bubble cursor. For the three other input techniques the rankings are more mixed. The box cursor receives a median rank of 2, the point cursor a median rank of 3, and the pie cursor a median rank of 3.5. The differences in preference rankings are significant, Friedman test $\chi^2(3, N = 8) = 16.35$, $p < .001$.

5.7. Comments about the input techniques

During the experiment participants made comments about the input techniques. We want to mention the

following, which point toward further reasons for participants' performance and questionnaire responses.

For the *point cursor* several participants remarked that the clutching involved in long cursor movements was awkward and that some planning of the finger's trajectory on the touchpad was required to avoid excessive clutching. This indirectly suggests that with the other input techniques participants appreciated the reduction in the distance they had to move the cursor to select distant targets.

For the *box and pie cursors* one participant explicitly appreciated the box cursor's large activation area, which made it easy to home in on targets: "You can throw it off and it will catch something; like a fishing net." However, during the final selection of targets all participants were slowed down. For the pie cursor one participant remarked: "I often had to look at the touchpad to tap with sufficient precision." All participants experienced incidents where they lost sense of the position of their finger relative to the touchpad and had to glance down at the touchpad to regain orientation.

For the *bubble cursor* participants made remarks such as "this one is rather cool" and commented that the bubble cursor required minimal mental effort, thus corroborating the performance and questionnaire data.

6. Discussion of experiment one

The bubble cursor was faster than the three other input techniques, had lower error rates, scored better on subjective satisfaction questions, and was preferred by participants. This supports Grossman and Balakrishnan's (2005) previous evaluation of the bubble cursor and extends it to input techniques operated with a touchpad and to comparisons with cell cursors. In contrast, the performance of the two cell cursors provides little support for cursor activation areas containing multiple objects. The box cursor helped participants select objects from farther away, but for both cell cursors it took a long time to select the target from the set of objects in the cursor activation area. The box cursor achieved lower error rates than the point cursor for small targets, and participants tended to prefer the box cursor over the point cursor. Conversely, the pie cursor had higher error rates than the point cursor.

Based on the experiment we see two main limitations in the design of the box and pie cursors. First, the box and pie cursors have no default selection. Thus, any selection requires that users determine and tap the touchpad region that corresponds to the cursor cell containing the target. This adds an additional element to object selection, and this element consumes more time than participants save by selecting objects from farther away. In contrast, one of the large benefits of the bubble cursor is that the closest object is always immediately selectable. By introducing a default selection in the box and pie cursors it may be possible to achieve bubble-cursor performance when the default object is selected and even better performance when users, in certain cases, select an object other than the default.

Second, the box and the pie cursors mix relative positioning (during movement) with absolute positioning (during selection). During movement, users drag a finger along the surface of the touchpad to move the cursor in the desired direction and for the desired distance. During selection, users tap in a region of the touchpad to select the corresponding cursor cell. The mix of relative and absolute positioning may be mentally taxing and in some cases confusing because the switch from relative to absolute positioning may entail a reversal in the direction of movement in motor space (e.g., after moving rightward to approach a target the user may have to tap the left-hand side of the touchpad to select the target, even though the initial rightward movement did not overshoot). Thus, it may be possible to improve performance by eliminating this switch in favour of using relative positioning only.

7. Experiment two

The purpose of the second experiment is to evaluate the performance of the box and pie cursors after introducing a default selection and using relative positioning only, as suggested by experiment one. As the mixing of relative and absolute positioning was closely tied to the touchpad, the second experiment also involves a change of input device from touchpad to mouse.

7.1. Participants

Eight participants (two female, six male) took part in the experiment. Participants' age ranged from 25 to 37 years with an average of 33 years. All participants had normal or corrected-to-normal vision and operated the mouse with their right hand. None of the participants had participated in experiment one.

7.2. Apparatus

The experiment was conducted on a 1.8 GHz laptop with an external mouse. The display resolution was set to 1024 × 768 on a 15" screen. For all input techniques mouse acceleration was deactivated, and the C:D gain was set to the middle value in the control panel of Windows XP.

7.3. Input techniques

In contrast to experiment one, the four input techniques were operated with a mouse. The point and bubble cursors were unchanged from the first experiment. The box and pie cursors were, however, modified to use mouse gestures (rather than touchpad taps) for the final selection and to include a default selection, as described below.

The new variant of the *box cursor* is operated by moving the mouse until the target is within the cursor activation area and then making a mouse gesture to select the target. The gesture consists of moving the mouse in the direction from the centre of the box cursor toward the target, while

holding down the left mouse button. In Fig. 8 the user can select the object in the left-hand cell of the cursor activation area by making a leftward gesture. Thus, the directions of the gestures are mapped to the cells of the box cursor. Further, the closest object is designated as the default selection. The default object is visually indicated by a thicker border of the cursor cell containing it (see Fig. 8), and it is selected by a mouse click, rather than a gesture. That is, selection of the default object is equivalent to the bubble cursor, and further objects farther away can be selected by gestures.

With the new variant of the *pie cursor* movement is also accomplished by moving the mouse and selection by a mouse gesture. The gesture consists of moving the mouse a short distance in the direction of the target, while holding down the left mouse button. The directions of the gestures are mapped to slices of the pie cursor. Further, a default selection can be made by clicking the mouse rather than making a gesture. For the pie cursor, the default selection is the first object encountered by extrapolating the mouse movement in its current direction, and it is visually indicated by a thicker arrow and a filled circle slightly larger than the object itself. Fig. 8 shows an example. Note that the default selection in the figure is the object to the left because the user is moving toward that object, not the closer object to the right. A special case arises when the pie cursor is within 10 pixels of an object; then that object is the default selection, independently of the direction of mouse movement.

7.4. Tasks

The tasks were the same as in experiment one.

7.5. Procedure

The procedure was similar to that of experiment one. Thus, the data collected again comprised 5760 trials. An experimental session with one participant lasted about 50 min.

7.6. Dependent measures

The dependent measures were the same as in experiment one.

8. Results of experiment two

8.1. Error rate

Fig. 9 summarizes the error rates after removal of 60 outlier trials with completion times more than three inter-quartile ranges above the upper quartile. We find a significant difference between input techniques, $F(3, 5) = 15.18, p < .01$. As in experiment one, the bubble cursor has a significantly lower error rate ($M = 1\%$, $SD = 7$) than the other input techniques ($M = 9\%$, $SD = 29$). There were no significant differences between the box, pie, and point cursors.

We also find an effect of size on error rates, $F(1, 7) = 10.06, p < .05$, and an interaction between input technique and size, $F(3, 5) = 18.12, p < .01$. The main reason for these effects is that the point cursor is more sensitive to target size than the other input techniques. With the point cursor error rates increase from 4% for

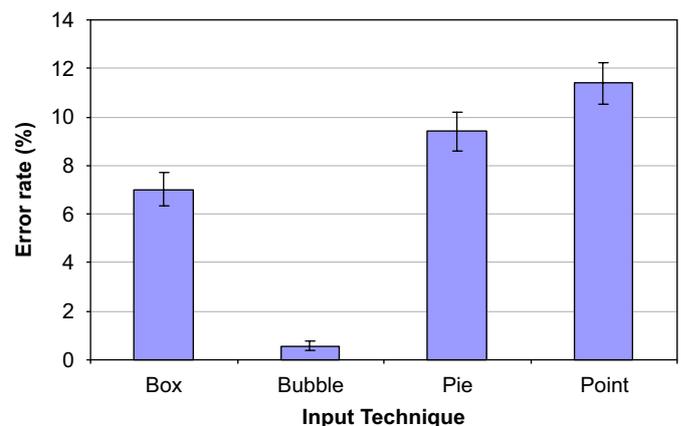


Fig. 9. Average error rates in experiment two ($N = 5700$); error bars show standard error of the mean.

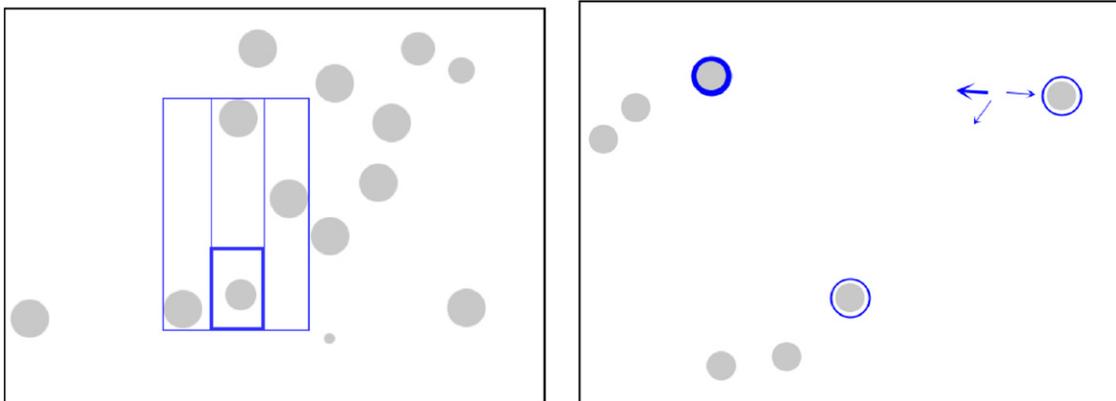


Fig. 8. The variants of the box cursor (left) and pie cursor (right) used in experiment two.

large targets to 19% for small targets; this makes it significantly less accurate for small targets than the box and pie cursors ($p < .05$). We find no effects of density or distance on error rates.

8.2. Trial completion time

Fig. 10 summarizes the trial completion times across input techniques for the 5297 non-error, non-outlier trials. We find a significant effect of input technique on trial completion time, $F(3, 5) = 46.58$, $p < .001$. Linear contrasts show that the bubble cursor ($M = 1035$ ms, $SD = 347$) is about 25% faster than the other input techniques. We find no differences among the box ($M = 1325$ ms, $SD = 489$), pie ($M = 1464$ ms, $SD = 555$), and point ($M = 1351$ ms, $SD = 438$) cursors.

We also find a significant effect of size on trial completion time, $F(1, 7) = 935.42$, $p < .001$, and an interaction between size and input technique, $F(3, 5) = 102.94$, $p < .001$. For all input techniques it takes longer to select small targets, but the point cursor is particularly sensitive to target size. Its trial completion time increases from 1075 ms ($SD = 236$) for large targets to 1679 ms ($SD = 395$) for small targets, making the box cursor significantly faster for small targets ($p < .05$).

As would be expected there is also an effect on trial completion time of distance $F(2, 6) = 57.44$, $p < .001$. As distance increases so does trial completion time. Further, distracter density influences trial completion time, $F(2, 6) = 112.20$, $p < .001$. As the number of distracters increases so does trial completion time. Contrary to experiment one, there is no interaction between input technique and distracter density.

8.3. Distance to target when selection occurs

As in experiment one, we find a significant difference between input techniques in how far from the target selection occurs, $F(3, 5) = 54.80$, $p < .001$. Linear contrasts show that selection with the pie cursor ($M = 78$ pixels, $SD = 70$) occurs from farther away than selections with the bubble ($M = 50$ pixels, $SD = 44$) and box cursors ($M = 48$ pixels, $SD = 51$), which in turn occur from farther away than selection with the point cursor ($M = 5$ pixels, $SD = 4$). Compared to experiment one, the box cursor is no longer the input technique for which selection occurs at the greatest distance from targets. It is also noteworthy that while the pie cursor enables selection from far away, it is not faster than the other input techniques.

8.4. Use of default selection

Default selection, gestures, and selection by positioning the cursor over the target were used in different ways across input techniques, see Table 2. For the box and pie cursors, the use of default selection is associated with lower error rates and trial completion times. Many

interpretations may be made of this association. One is that default selection is faster because it is simpler; another that participants do not turn to selection by gesture and by positioning the cursor over the target until other forms of selection have failed. Default selection with the box cursor ($M = 1264$ ms) is slower than selection with the bubble cursor ($M = 1035$ ms), in spite of the use of the same mechanism for determining the default selection (viz., the closest object). Using gestures in the pie and box cursors is about 30% slower than trial completion times with the point cursor.

8.5. Modelling trial completion times

We performed modelling similar to that reported for experiment one. Disregarding distracter density, Fitts's law models of the data yield r^2 values of .74 (box), .86 (bubble), .88 (pie) and .94 (point). The refined model, which uses combinations of index of difficulty and density, shows that inclusion of density in the model gives a significantly better fit for the pie and bubble cursors ($p < .05$), but not for the box cursor ($p > .5$). Using ID to signify the index of difficulty and DD to signify distracter density, the model for the pie cursor is $MT = 675 \text{ ms} + 156 \text{ ms} \times ID + 5 \text{ ms} \times DD$, and for the bubble cursor it is $MT = 323 \text{ ms} + 149 \text{ ms} \times ID + 3 \text{ ms} \times DD$. For comparison, the model for the box cursor is $MT = 565 \text{ ms} + 168 \text{ ms} \times ID$, and for the point cursor it is $MT = 222 \text{ ms} + 258 \text{ ms} \times ID$.

8.6. Subjective assessments and preference

Table 3 summarizes the answers to the questionnaires administered to participants. An overall multivariate analysis shows a significant difference between input techniques, Wilks's $\lambda = .23$, $F(3, 5) = 5.47$, $p < .05$, and linear contrasts show that the bubble cursor scores consistently better than the three other input techniques on six questions, see Table 3.

The preference ranking performed by participants after they had used all four input techniques shows a unanimous preference for the bubble cursor (as in experiment one); for the other input techniques the rankings are more mixed. The box cursor receives a median rank of 2; the point and pie cursors a median rank of 3.5. The differences in preference rankings are significant, Friedman test $\chi^2(3, N = 8) = 17.25$, $p < .001$.

8.7. Comments about the input techniques

Some participants commented that the *box cursor* would occasionally change during the selection process, probably due to an unintended mouse movement just before pressing the mouse button. Two participants said they were unsure about which object they were about to select when using gestures rather than default selection. It also seemed as if the visual indication of the default selection was not sufficiently prominent.

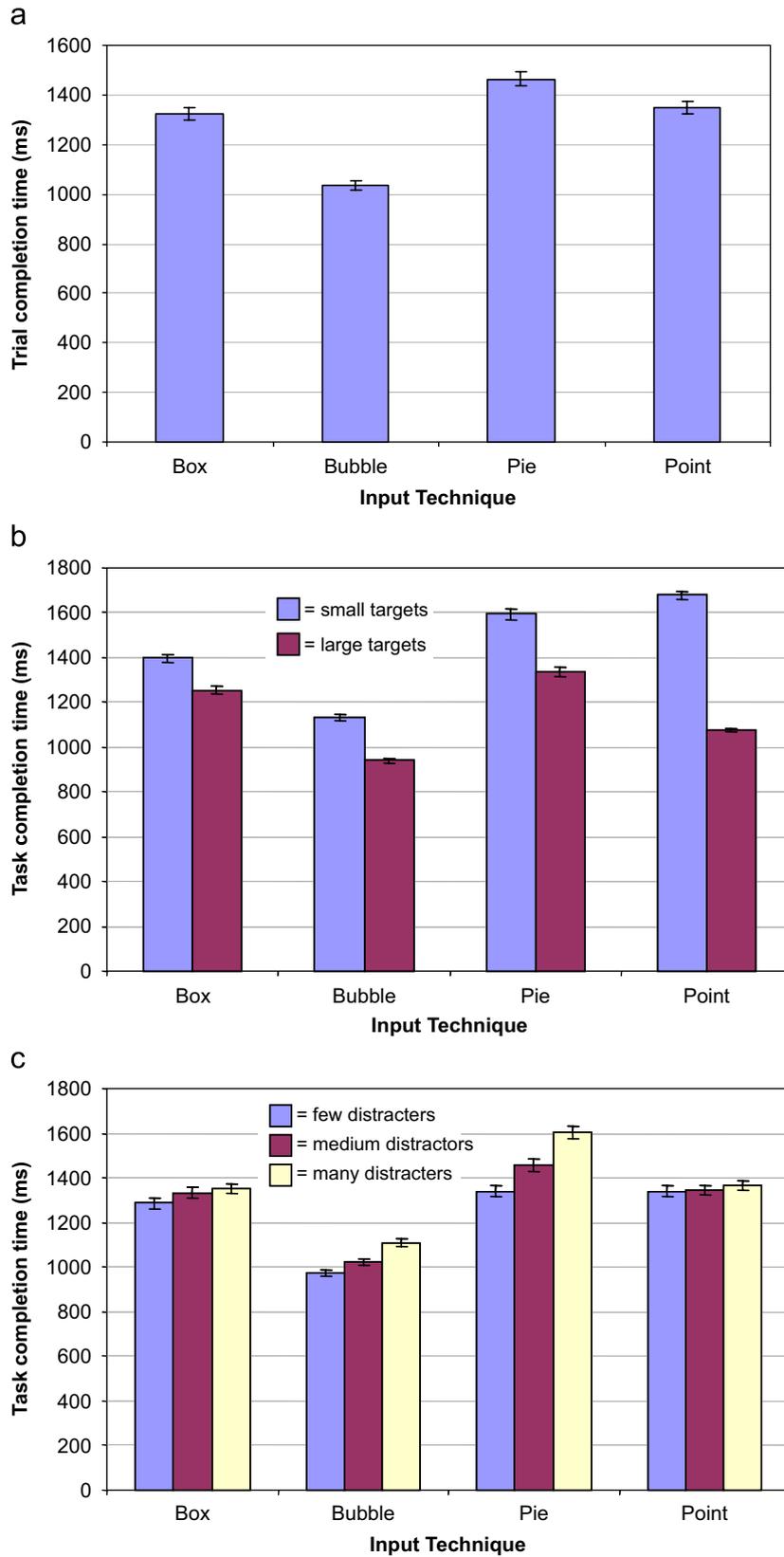


Fig. 10. Average trial completion times in experiment two for (a) input techniques, (b) input techniques given different target sizes, and (c) input techniques given different distracter densities; error bars show standard error of the mean ($N = 5297$).

Table 2
Participants' use of default selection and the corresponding trial completion times for non-outlier, error-free trials ($N = 5297$)

Input technique	Selection mode	Frequency (%)	Trial completion time (ms)		Error rate	
			<i>M</i>	<i>SD</i>	<i>M</i> (%)	<i>SD</i>
Box	Default	87	1264	443	6.1	24.0
	Gesture	13	1736	584	12.8	33.4
Bubble	Default	92	1042	352	.5	7.3
	Over the target	8	952	281	.8	9.3
Pie	Default	57	1331	465	4.0	19.5
	Gesture	22	1774	597	15.6	36.3
	Over the target	21	1505	598	16.1	36.8
Point	Over the target	100	1351	438	11.4	31.8

Error rates are given for non-outlier trials ($N = 5700$).

Table 3
Participants' subjective ratings of the input techniques in experiment two ($N = 8$); significant differences between input techniques are marked with asterisks

Question	Box cursor		Bubble cursor		Pie cursor		Point cursor	
	<i>M</i>	<i>SD</i>	<i>M</i>	<i>SD</i>	<i>M</i>	<i>SD</i>	<i>M</i>	<i>SD</i>
1. Overall, the cursor was * (1: very difficult to use–5: very easy)	3.75	.89	5.00	.00	2.63	1.30	2.88	1.13
2. General comfort * (1: very uncomfortable–5: very comfortable)	3.75	1.16	4.88	.35	3.13	1.13	2.88	1.13
3. Smoothness during operation was * (1: very rough–5: very smooth)	3.63	1.30	4.75	.46	3.25	1.16	3.25	1.28
4. Accurate pointing was * (1: easy–5: difficult)	2.13	1.25	1.38	.52	3.00	1.31	3.75	1.16
5. Force required for actuation was (1: appropriate–5: inappropriate)	2.25	1.16	1.50	.53	2.25	1.16	2.63	1.60
6. Mental effort required for operation was * (1: appropriate–5: inappropriate)	2.50	1.20	1.13	.35	3.13	1.25	2.50	1.20
7. Physical effort required for operation was * (1: appropriate–5: inappropriate)	1.87	.83	1.38	.52	1.87	.83	2.75	1.16
8. Finger fatigue (1: none–5: very high)	2.13	.83	1.50	.53	1.87	1.13	2.38	1.30
9. Wrist fatigue (1: none–5: very high)	1.87	.83	1.75	.71	2.25	1.04	2.50	1.20
10. Arm fatigue (1: none–5: very high)	2.00	1.20	1.63	.74	2.00	.93	2.13	1.55
11. Shoulder fatigue (1: none–5: very high)	2.13	1.36	1.63	.74	2.00	.93	2.13	1.55
12. Neck fatigue (1: none–5: very high)	1.75	1.16	1.50	.76	2.00	.93	1.87	1.36

* $p < .05$.

The *pie cursor* was considered hard to use when there were distracters between the current position and the target or when the cursor was close to a target. In the latter case, the default selection would often change even for small, unintended mouse movements. Two participants expressed enthusiasm about the potential of using direction for object selection; another commented that it was “weird”.

The *bubble cursor* was described as easy and intuitive. Participants also commented that selection of objects in densely populated areas of the screen was slightly difficult. In these situations the bubble cursor approaches the behaviour of a point cursor. The only other critical comment was that the bubble would occasionally become very large.

9. Discussion of experiment two

The second experiment yields results similar to the first experiment. The bubble cursor is faster and more accurate than the other input techniques, it is preferred over the other techniques, and it scores consistently higher on the

satisfaction questionnaire. The bubble cursor was 32% faster than the three other input techniques in experiment one; it is 25% faster in experiment two. Thus, the gap between the bubble cursor and the two cell cursors has not been narrowed much by the improvements introduced in experiment two. Relative to the point cursor, the box cursor is faster for small targets and both cell cursors are more accurate for small targets (in experiment one this was only the case for the box cursor).

Participants made frequent use of the default selection, particularly with the box cursor. For the box cursor, the default selection was the closest object. Thus, using the default selection meant using the box cursor in a manner similar to the bubble cursor. The increase in trial completion times relative to the bubble cursor is probably due to the visual overhead of the box cursor and the cognitive overhead of occasionally considering the possibility of making gestures to select non-default objects.

For the pie cursor, the default selection was the object in the current direction of mouse movement. This design choice did not lead to satisfactory performance, though

participants selected targets from farther away with the pie cursor than with the input techniques using the closest object as the default. In fact, participants had considerable difficulty controlling selection by direction, particularly for close-by objects. The algorithm for detecting direction averages the direction of movement over a series of mouse movements. When these are few and short the algorithm becomes error prone because a small imprecision in movement direction may change the default selection.

10. General discussion and conclusion

We have explored input techniques that dynamically adjust their cursor activation area. Whereas the bubble cursor has a cursor activation area containing exactly one object, the box and pie cursors have multiple objects in their cursor activation areas. In two experiments, the bubble cursor performed better than the pie and box cursors. For small targets the pie and box cursors were generally more accurate than a point cursor, and in one case also faster. Participants unanimously preferred the bubble cursor.

10.1. The bubble cursor

The bubble cursor performs well both when operated with a mouse and with a touchpad. This result replicates and extends Grossman and Balakrishnan's (2005) findings about the bubble cursor. Compared to their study, we base our evaluation on scenes with randomly placed distracters. As the number of distracters increases so does target selection time, but even for densely populated scenes the bubble cursor performs better than the other input techniques.

The main characteristic of the bubble cursor is the increase in the effective width of the closest object. This increase is optimal in the sense that it removes all empty space between objects and, thereby, eliminates errors consisting of clicking outside a target and not hitting anything. The removal of empty space is however restricted to the selection phase of pointing operations; for the approach phase the bubble cursor retains the same relationship between mouse movement and resulting cursor movement as the standard point cursor. As a result cursor movement is experienced as smooth, but this smoothness is achieved at the expense of suspending improvement over the point cursor until the pointing operation has progressed so far that there are no distracters between the cursor and the target. Object pointing abandons smoothness to reduce the amount of movement in motor space but has been found to be slower and less accurate than the bubble cursor (Grossman and Balakrishnan, 2005). The box and pie cursors maintain smoothness while approaching targets but the bubble cursor also performs better than box and pie cursors.

10.2. Cursor activation areas containing multiple objects

Overall the box and pie cursors showed similar performance to a point cursor, but they did better for small targets: In three of four cases this resulted in significantly better accuracy, in one case also in lower trial completion times.

The box and pie cursors were intended to be particularly advantageous in densely populated displays. Whereas the bubble cursor degrades to a point cursor when there is no space between objects (as is for example the case with cells in a spreadsheet) the box and pie cursors enable selection of close-by objects in these situations by a coarse-grained indication of the cursor cell containing the target. However, no performance improvement was found in the experiments; moreover, trial completion times increase with increasing numbers of distracters.

One explanation for the somewhat disappointing performance of the two cell cursors is that their cursor activation areas contain multiple objects. This way, target selection involves that users must choose among the objects in the cursor activation area. Models of skilled performance predict that this takes time. With the keystroke level model (KLM; Card et al., 1980), conventional point-and-click operations are modelled as a mental operator followed by a pointing movement and a click (MPK). Because the click is fully anticipated in the pointing movement there should not be a mental operator between the pointing movement and the click. For cell cursors the choice of the target from the cursor activation area may mean that the click is not fully anticipated in the pointing movement and that point-and-click operations therefore involve an additional mental operator (MPMK). The performance difference in experiment two between trials completed using the default selection and trials completed by selecting an object other than the default suggests that additional mental preparations are needed when selecting objects other than the default. In experiment one the duration of the selection phase for the box (1228 ms) and pie (1150 ms) cursors approaches KLM's value of 1350 ms for a mental operator.

Accepting that the final selection of the target will be slowed down, another explanation for the performance of the cell cursors may be that the initial part of the movement is not speeded up sufficiently. If cell cursors were to perform well compared to the bubble cursor, they should facilitate getting much quicker to the point where selection among objects in the cursor activation area takes over from movement of the cursor. With the box and pie cursors, participants frequently performed a ballistic movement toward the target. While this may be natural, often the box and pie cursors ended so close to the target that they were only used as bubble cursors; that is, to select the closest object. Maybe, performance gains are to be found in some other mechanism for the initial part of the movement; one that does not mimic normal mouse movement. It may also be possible to achieve improvements from a tighter

integration of the initial movement and the final selection, akin to the performance improvements MacKenzie and Oniszczak (1998) achieved by integrating movement and selection for normal touchpad pointing; in their case by utilizing finger pressure and tactile feedback. A final possibility may be to drastically increase C:D gain. This may help because precise pointing is provided by the selection mechanism of the cell cursors without requiring high movement precision in motor space.

In the second experiment we saw mixed results from using direction to indicate the default selection. While some participants were enthusiastic and selected targets from far away, it proved cumbersome to select close-by targets. One reason for this is that indications of direction become increasingly error prone as targets get closer. Another reason is that when objects are close to each other small changes in the direction of movement may result in inadvertently changing the default selection. Moyle and Cockburn (2005) find that when users indicate direction by flicking their mouse in the desired direction, as in our second experiment, the mean angular error is 3.6° . This suggests that direction can be indicated fairly precisely, a finding consistent with Guiard et al. (2004) who find no performance differences for object pointing across conditions in which users had a distance of either 10, 250, or 1330 pixels for indicating the direction toward the target. However, angular error appears to interact with input device (Moyle and Cockburn, 2005) and to increase as movement speed increases (Hwang et al., 2005).

In summary, while it may be possible to improve the performance of the box and pie cursors we doubt they will become as fast and accurate as the bubble cursor. The overhead involved in the final selection of the target from the set of objects in the cursor activation area appears larger than the improvements to be gained from selecting targets from farther away. A special case where box and pie cursors may still hold promise is in the selection of targets from groups of small and very closely spaced objects.

10.3. Modelling performance

In both experiments the bubble cursor performed better with large targets than with small targets, though the effective size of targets was determined by the random position of the distracters and thus similar for large and small targets. An effect of actual target size was also present for the box and pie cursors and in Grossman and Balakrishnan's (2005) study of the bubble cursor. This suggests that the visual impression of the difficulty of the pointing operation influences performance. If this explanation is correct it may imply a reduction in model accuracy when actual width is replaced by effective width in Fitts's law, which presupposes that they are equal. Further, input techniques that enlarge targets in motor space as compared to display space are often sensitive to the density of distracters and it is not obvious how distracters should be incorporated in performance models. For the box, bubble,

and pie cursors we have shown that performance models with a data point for each combination of distance, size, and density level explain more variance than do models based on only distance and size. This is unsurprising as density significantly affected trial completion times but also indicates a need for incorporating information about distracter density in Fitts's law. Balakrishnan (2004) finds that in general the effects of distracters are under-researched. Furthermore, the performance models for the box and pie cursors in experiment one, which did not involve default selection, have large intercepts, consistent with an interpretation that the final selection of the target from the objects in the cursor activation area requires mental preparations that are independent of target size, distance to target, and distracter density.

10.4. Effects of increased experience with the input techniques

In the experiments participants spent an average of 12 min familiarizing themselves with the input techniques and performed 180 trials with each of them, disallowing claims about proficient use of the input techniques. The number of trials is, however, similar to that of other studies of input techniques (e.g., Guiard et al., 2004; Bezerianos and Balakrishnan, 2005; Dai et al., 2005). To explore whether increased experience with the input techniques affects their relative performance we conducted an informal experiment in which one of the authors performed eight sessions of trials for a total of 1440 trials with each input technique. The informal experiment was identical to experiment one, except that the same participant completed all eight sessions.

Here we consider the last four sessions as indicators of the performance when a user has become experienced with the input techniques. Table 4 summarizes trial completion times and error rates for these sessions (because we consider skilled performance we did not remove outliers). As can be seen from the table, the box and pie cursors have much lower trial completion times compared to experiment one (see Fig. 6), 752 and 626 ms lower, respectively. Moreover, they also have much lower error rates (cf. Fig. 5); 8.9 percentage points lower for the box cursor and 14.6 percentage points lower for the pie cursor. While increased experience leads to substantial

Table 4
Performance after prolonged use of the input techniques ($N = 2880$)

Input technique	Trial completion time (ms)		Error rate	
	<i>M</i>	SD	<i>M</i> (%)	SD
Box	1797	473	2.1	14.3
Bubble	1436	371	1.5	12.3
Pie	2064	603	4.4	20.6
Point	2130	618	2.5	15.6

performance gains, the rank order of the input techniques' performance is similar to that of experiments one and two.

10.5. Future work

The bubble cursor warrants further study. What are the tasks for which a bubble cursor is appropriate? Can it handle or be extended to handle scenes with both small and very large objects (e.g., icons and application windows on a desktop), scenes with dynamic contents (e.g., cascading menus), objects that can be accessed at multiple levels of detail (e.g., characters, words, and paragraphs in text processing), and simultaneous selection of multiple objects (e.g., ranges of cells in a spreadsheet)? Can it be smoothly integrated with other cursors more appropriate for some of these tasks?

Cell cursors such as the box and pie cursors may be applicable in special cases. One such case could be selection of targets from groups of small and very closely spaced objects that otherwise require high movement precision during the final target acquisition. Another case may be on very large displays, where the ability to select from far away may outweigh the difficulty during the latter part of selection. Cell cursors appear to have a two-step selection process and in that regard to constitute an improvement over the multi-step selection processes of cross-keys and precision-handles for high-precision pointing (Albinsson and Zhai, 2003). In some use contexts it may be possible to bypass the first step and thereby reduce pointing to selection among objects already in the cursor activation area. One example could be displays with few objects, including modal dialog boxes for which the cursor could be automatically moved to the dialog box. Finally, due to the reduced requirements for precise movements cell cursors may hold promise in use situations without stable support for the input device and the user's arm and hand, including the work of many service technicians and other mobile users.

An increasing number of input techniques extrapolate the initial direction of cursor movements to speed up pointing operations. This is, for example, the basis for object pointing, the vacuum, and the second variant of the pie cursor. Also, it may be possible to improve the bubble cursor by replacing its circular cursor activation area with an elliptical area prolonged in the direction of movement. This warrants further study of how the initial cursor movement relates to the total movement.

Acknowledgements

We thank the participants in our experiments for their help.

References

Accot, J., Zhai, S., 1999. Performance evaluation of input devices in trajectory-based tasks: an application of the steering law. In:

- Proceedings of the CHI 99 Conference on Human Factors in Computing Systems. ACM Press, New York, pp. 466–472.
- Accot, J., Zhai, S., 2002. More than dotting the i's—foundations for crossing-based interfaces. In: Proceedings of the CHI 2002 Conference on Human Factors in Computing Systems. ACM Press, New York, pp. 73–80.
- Albinsson, P.-A., Zhai, S., 2003. High precision touch screen interaction. In: Proceedings of the CHI 2003 Conference on Human Factors in Computing Systems. ACM Press, New York, pp. 105–112.
- Balakrishnan, R., 2004. "Beating" Fitts' law: virtual enhancements for pointing facilitation. *International Journal of Human-Computer Studies* 61 (6), 857–874.
- Baudisch, P., Cutrell, E., Robbins, D., Czerwinski, M., Tandler, P., Bederson, B., Zierlinger, A., 2003. Drag-and-pop and drag-and-pick: techniques for accessing remote screen content on touch- and pen-operated systems. In: Proceedings of the INTERACT 03 Conference on Human-Computer Interaction. IOS Press, Amsterdam, pp. 57–64.
- Bezerianos, A., Balakrishnan, R., 2005. The vacuum: facilitating the manipulation of distant objects. In: Proceedings of the CHI 2005 Conference on Human Factors in Computing Systems. ACM Press, New York, pp. 361–370.
- Blanch, R., Guiard, Y., Beaudouin-Lafon, M., 2004. Semantic pointing: improving target acquisition with control-display ratio adaptation. In: Proceedings of the CHI 2004 Conference on Human Factors in Computing Systems. ACM Press, New York, pp. 519–526.
- Card, S.K., English, W.K., Burr, B.J., 1978. Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT. *Ergonomics* 21 (8), 601–613.
- Card, S.K., Moran, T.P., Newell, A., 1980. The keystroke-level model for user performance time with interactive systems. *Communications of the ACM* 23 (7), 396–410.
- Cockburn, A., Firth, A., 2003. Improving the acquisition of small targets. In: *People and Computers XVII: British Computer Society Conference on Human-Computer Interaction*. Springer, New York, pp. 181–196.
- Cohen, O., Meyer, S., Nilsen, E., 1993. Studying the movement of high-tech. rodents: pointing and dragging. In: Proceedings of the INTERCHI 93 Conference on Human Factors in Computing Systems: Adjunct Proceedings. ACM Press, New York, pp. 135–136.
- Dai, L., Goldman, R., Sears, A., Lozier, J., 2005. Speech-based cursor control using grids: modelling performance and comparisons with other solutions. *Behaviour & Information Technology* 24 (3), 219–230.
- Douglas, S.A., Kirkpatrick, A.E., MacKenzie, I.S., 1999. Testing pointing device performance and user assessment with the ISO 9241, part 9 standard. In: Proceedings of the CHI 99 Conference on Human Factors in Computing Systems. ACM Press, New York, pp. 215–222.
- Epps, B.W., 1986. Comparison of six cursor control devices based on Fitts' law models. In: Proceedings of the Human Factors Society 30th Annual Meeting. Human Factors Society, Santa Monica, CA, pp. 327–331.
- Fitts, P.M., 1954. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology* 47 (6), 381–391.
- Gold, C.M., 1994. Advantages of the voronoi spatial model. In: Frederiksen, P. (Ed.), *Proceedings of the Eurocarto XII Conference on Geo Related Databases*. Technical University of Denmark, Lyngby, DK, pp. 1–10.
- Grossman, T., Balakrishnan, R., 2005. The bubble cursor: enhancing target acquisition by dynamically resizing of the cursor's activation area. In: Proceedings of the CHI 2005 Conference on Human Factors in Computing Systems. ACM Press, New York, pp. 281–290.
- Guiard, Y., Beaudouin-Lafon, M., 2004. Fitts' law 50 years later: applications and contributions from Human-Computer interaction (special issue). *International Journal of Human-Computer Studies* 61 (6).
- Guiard, Y., Blanch, R., Beaudouin-Lafon, M., 2004. Object pointing: a complement to bitmap pointing in GUIs. In: *Proceedings of Graphics Interface 2004*. Canadian Human-Computer Communications Society, London, ON, pp. 9–16.

- Hertzum, M., Hornbæk, K., 2005. TouchGrid: touchpad pointing by recursively mapping taps to smaller display regions. *Behaviour & Information Technology* 24 (5), 337–346.
- Hwang, F., Keates, S., Langdon, P., Clarkson, J., 2005. A submovement analysis of cursor trajectories. *Behaviour & Information Technology* 24 (3), 205–217.
- Ingmarsson, M., Dinka, D., Zhai, S., 2004. TNT—a numeric keypad based text input method. In: *Proceedings of the CHI 2004 Conference on Human Factors in Computing Systems*. ACM Press, New York, pp. 639–646.
- ISO, 2000. *Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs)—Requirements for Non-Keyboard Input Devices* (Rep. No. ISO 9241-9). International Organization for Standardisation, Geneva, CH.
- Kabbash, P., Buxton, W., 1995. The “Prince” technique: Fitts’ law and selection using area cursors. In: *Proceedings of the CHI 95 Conference on Human Factors in Computing Systems*. ACM Press, New York, pp. 273–279.
- Keyson, D.V., 1997. Dynamic cursor gain and tactual feedback in the capture of cursor movements. *Ergonomics* 40 (12), 1287–1298.
- MacKenzie, I.S., 1992. Fitts’ law as a research and design tool in Human–Computer interaction. *Human–Computer Interaction* 7 (1), 91–139.
- MacKenzie, I.S., Oniszczak, A., 1998. A comparison of three selection techniques for touchpads. In: *Proceedings of the CHI 98 Conference on Human Factors in Computing Systems*. ACM Press, New York, pp. 336–343.
- MacKenzie, I.S., Sellen, S., Buxton, W., 1991. A comparison of input devices in elemental pointing and dragging tasks. In: *Proceedings of the CHI 91 Conference on Human Factors in Computing Systems*. ACM Press, New York, pp. 161–166.
- MacKenzie, I.S., Kauppinen, T., Silfverberg, M., 2001. Accuracy measures for evaluating computer pointing devices. In: *Proceedings of the CHI 2001 Conference on Human Factors in Computing Systems*. ACM Press, New York, pp. 9–16.
- McGuffin, M., Balakrishnan, R., 2002. Acquisition of expanding targets. In: *Proceedings of the CHI 2002 Conference on Human Factors in Computing Systems*. ACM Press, New York, pp. 57–64.
- Moyle, M., Cockburn, A., 2005. A flick in the right direction: a case study of gestural input. *Behaviour & Information Technology* 24 (4), 275–288.
- Robbins, D.C., Cutrell, E., Sarin, R., Horvitz, E., 2004. ZoneZoom: map navigation for smartphones with recursive view segmentation. In: *Proceedings of the AVI 2004 Working Conference on Advanced Visual Interfaces*. ACM Press, New York, pp. 231–234.
- Worden, A., Walker, N., Bharat, K., Hudson, S.E., 1997. Making computers easier for older adults to use: area cursors and sticky icons. In: *Proceedings of the CHI 97 Conference on Human Factors in Computing Systems*. ACM Press, New York, pp. 266–271.
- Zhai, S., Buxton, W., Milgram, P., 1994. The ‘silk cursor’: investigating transparency for 3D target acquisition. In: *Proceedings of the CHI 94 Conference on Human Factors in Computing Systems*. ACM Press, New York, pp. 459–464.
- Zhai, S., Conversy, S., Beaudouin-Lafon, M., Guiard, Y., 2003. Human on-line response to target expansion. In: *Proceedings of the CHI 2003 Conference on Human Factors in Computing Systems*. ACM Press, New York, pp. 177–184.