

# TractorBeam Selection Aids: Improving Target Acquisition for Pointing Input on Tabletop Displays

J. Karen Parker<sup>1,3</sup>, Regan L. Mandryk<sup>2</sup>, Michael N. Nunes<sup>1</sup>, and Kori M. Inkpen<sup>1</sup>

<sup>1</sup> Faculty of Comp. Sci., Dalhousie University, Halifax, NS, Canada

<sup>2</sup> School of Comp. Sci., Simon Fraser University, Burnaby, BC, Canada

<sup>3</sup> Dept. of Comp. Sci., University of British Columbia, Vancouver, BC, Canada  
parker@cs.ubc.ca, rlmndryk@cs.sfu.ca  
{nunes, inkpen}@cs.dal.ca

**Abstract.** This paper presents a comparison of several selection aids to improve pointing input on tabletop displays. Our previous research explored the TractorBeam—a hybrid point-touch interaction technique for tabletop displays. We found that while pointing input was preferred (over touch) by users of tabletop displays, it was slower for small distant targets. Drawing from previous work on improving target acquisition for desktop displays, we developed and tested three selection aids to improve pointing selection of small distant targets on *tabletop* displays: expanding the cursor, expanding the target, and snapping to the target. Our experiments revealed that all three aids resulted in faster selection times than no selection aid at all, with snapping to the target being the fastest. Additionally, participants liked snapping to the target better than the other selection aids and found it to be the most effective for selecting targets.

## 1 Introduction

Tabletop displays have emerged in the past 10 years as an area of interest in HCI research. As this research continues, it is important to develop effective interaction techniques for these types of displays. Although some researchers have developed specialized input devices and techniques for tabletop displays [9, 2, 11, 14], very few have conducted systematic evaluations on the effectiveness of these techniques. Determining appropriate input devices and interaction techniques is critical if tabletop displays are to become mainstream.

In our previous work, we developed the TractorBeam, an innovative technique which seamlessly combines remote pointing and touch – using a stylus – on tabletop displays [9]. Results from this work demonstrated that remote pointing was faster than touch input for large targets, was preferred over touch, and was also employed more often when users were given a choice. However, for small distant targets pointing was slower than touch. In remote pointing, small movements made with the hand are amplified on the screen. This amplification increases as distance to the target increases, so even though users must reach further to touch small distant targets than point to them, it is easier to make an accurate selection using touch.

Due to the amplification of small movements for distant targets, we felt that augmenting the technique with a selection aid might improve acquisition of small, distant targets. Past research into improving target acquisition has focused on desktop

displays. Researchers have explored methods such as expanding targets [7], area cursors [6], bubble cursors [4], object pointing [5], and semantic pointing [1]. Since the TractorBeam interaction technique for tabletop displays is significantly different than standard mouse-based interaction for desktop environments, we felt it was important to conduct a preliminary investigation of these previously proposed alternatives.

Drawing from this previous research, we developed and tested three selection aids to augment our TractorBeam technique, in order to improve acquisition of small distant targets on tabletop displays:

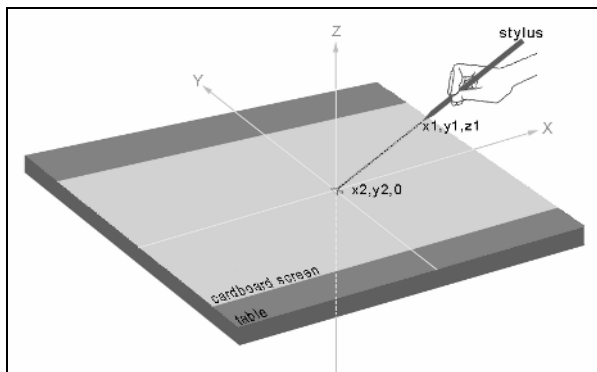
1. expanding the cursor (expand-cursor)
2. expanding the target (expand-target)
3. snapping to the target (snap-to-target)

All three selection aids, along with a fourth control condition, were evaluated for speed and accuracy, with snap-to-target emerging as the best option; it was fastest, and preferred by participants. The snap-augmented TractorBeam is a very good solution to the problems of reaching small distant targets encountered in our previous work.

## 2 Related Work

### 2.1 TractorBeam Technique

The TractorBeam interaction technique is a novel stylus-based input technique that combines close touch and distant pointing, allowing users to interact with nearby parts of the display more naturally with a stylus, and use the pointing functionality when they need to select an item that is beyond their reach. The technique works as follows: Using a stylus, the user points at the tabletop display. A cursor appears on the display to show the current trajectory of the stylus (Figure 1). The user moves the stylus around until the cursor is on the desired item. To select the item, the user clicks the button located on the top of the stylus.



**Fig. 1.** TractorBeam interaction technique

This allows for seamless interaction with all parts of the display. To interact with a close object, the user touches the stylus to the table, as one would normally use a stylus. To interact with a distant object, the user points the stylus towards their desired target, casting a virtual beam which positions the cursor where the user is pointing.

## 2.2 Other Tabletop Interaction Techniques

While previous tabletop research has used a wide variety of inputs, few researchers have specifically investigated interaction techniques for tabletop displays. Exceptions to this include Wu and Balakrishnan [14] who developed a suite of hand and finger gestures for multi-touch tabletop displays. Also, tangential to tabletop research, Rekimoto and Saitoh's hyperdragging (dragging an item off one display and onto another with a mouse) and pick-and-drop (picking an item up from one display and dropping it on another with a stylus) techniques allow users to move files between a tabletop and other devices, including distant displays such as large wall screens [10].

## 2.3 Improving Target Acquisition

Several researchers have proposed solutions to improve target acquisition time on traditional desktop computer monitors with mouse input. In attempting to improve target acquisition on tabletop displays, knowledge of these existing desktop techniques provides insight into possible solutions.

### Expanding the Target

Dynamically sized widgets which change size as a cursor approaches them (expanding targets), such as those used in the OS X operating system [8], are becoming more common in current user interfaces. McGuffin and Balakrishnan [7] investigated the effectiveness of expanding targets by comparing them to statically sized targets in a Fitts' task. The results from this work found that task performance was governed by the expanded target size, rather than the initial target size, even when they were already 90% of the way to the target before the expansion happened. This means that it is not necessary to expand a target until the cursor has traveled 90% of the distance to that target, since the same benefits will be achieved by expanding the target at that distance as at further distances.

Zhai et al. [15] further investigated expanding targets to determine whether McGuffin's results held when users did not know whether or not a target would expand. They ran trials in which targets would randomly shrink, expand, or remain unchanged, and found that target expansion improved pointing performance even when the user was not able to predict the expansion beforehand.

### Enlarging the Cursor

Kabbash and Buxton investigated the use of an "area cursor" in a Fitts' task and showed that, when using an area cursor to select a point, the action could be modeled with Fitts' law by making  $W$  (width of the target in Fitts' equation) the width of the

cursor, rather than the target [6]. The authors tested both a single point cursor moving between two large target areas, and a single large area cursor moving between two small target points. They found that the area cursor performed better than a single-point cursor in the task [6]. Worden et al. augmented area cursors with sticky icons in a study of basic selection tasks with older adult users [13] and found significant improvements in target selection times for small targets.

Although an area cursor could provide faster selection, it might be difficult for users to complete finer-grain actions. Grossman and Balakrishnan's bubble cursor solves this problem with its dynamic activation area, which only becomes large when it is close to a viable target [4]. Movement times for the bubble cursor were faster than both a standard point cursor and the object pointing technique [4].

### **Object and Semantic Pointing**

Guiard et al. introduced the idea of object pointing, where the cursor moves between valid targets and never travels in empty space between targets, as a means for improving target acquisition [5]. While object pointing outperforms regular pointing in Fitts' tasks, it may not be appropriate for interactions that require manipulations other than simple selection. For object pointing to work effectively, users would be required to enable it whenever they wanted to make selections, and disable it when they did not.

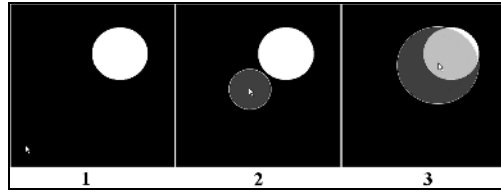
In semantic pointing, a related technique, targets "expand" in motor space (but not in visual space) according to their importance [1]. For example, as a user moves across a button they will move more slowly than when they move across a blank space because the button is expanded in motor space. Although this technique may be effective with a mouse or another relative input device, is not appropriate for direct input because direct input requires constant mapping of the cursor with the input device.

## **3 TractorBeam Selection Aids**

Our previous work revealed that acquisition of small, distant targets was difficult with the TractorBeam, our hybrid point-touch technique. Thus, we explored three possible selection aids to solve this problem: expanding the cursor (expand-cursor), expanding the target (expand-target), and snapping to the target (snap-to-target).

### **3.1 Expanding the Cursor**

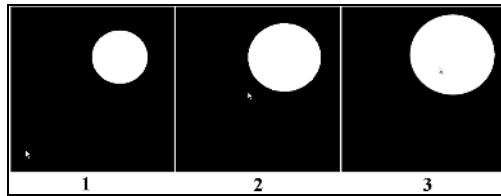
With this selection aid, users make selections using a "selection halo" area which surrounds the cursor, rather than having to use a single cursor point for selection. This is similar to the bubble cursor used by Grossman and Balakrishnan [4], which was shown to improve acquisition of targets. Whenever the cursor travels at least 90% of the distance to a target, a 30mm halo appears under the cursor and immediately expands to 60mm (Figure 2). The halo shrinks and disappears whenever the cursor moves outside of the 90% range. In order to make these changes appear more seamless, both the expansion and shrink are animated. In order to select a target, the selection halo must only overlap the target, not encompass it.



**Fig. 2.** Expand-cursor: (1) Cursor begins to approach target; (2) Cursor reaches 90% threshold and halo appears and begins expansion; (3) Halo continues expansion until it is full size

### 3.2 Expanding the Target

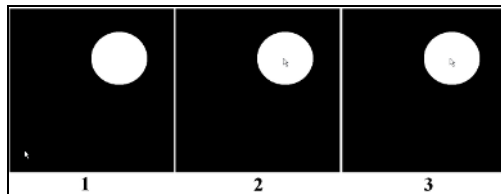
Similar to the expanding targets studied by McGuffin and Balakrishan [7], our expanding-target selection aid expands targets from their original size whenever the cursor is within 90% of the total distance traveled to the target (Figure 3). Targets shrink whenever the cursor moves outside of the 90% range. In order to make these changes appear seamless, both the expansion and shrink are animated. In our study, targets started at 30mm, 40mm, or 50mm, and expanded to a final size of 60mm.



**Fig. 3.** Expand-target (1) Cursor begins to approach target; (2) Cursor reaches 90% threshold and target begins expansion; (3) Target continues expansion until it has reached full size

### 3.3 Snapping to the Target

With this selection aid, the cursor “snaps” to the center of the target whenever it comes within 90% of the total distance to the target (Figure 4). It remains in this snapped position unless the “real” cursor position moves outside of this 90% range.



**Fig. 4.** Snap-to-target: (1) Cursor begins to approach target; (2) Cursor reaches 90% threshold and immediately snaps to centre of target; (3) Cursor remains snapped to target centre until pointer moves outside of 90% range

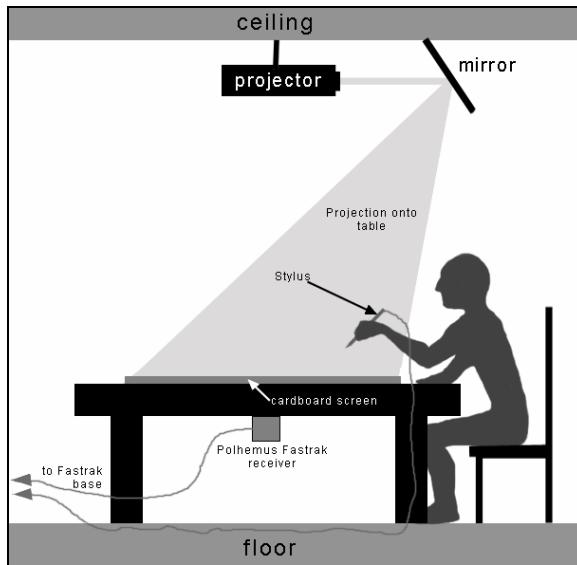
## 4 Experimental Design

### 4.1 Participants

Twenty-four participants, 18 male and 6 female, took part in our study. All participants were university students, staff, or faculty, and were right handed. None had participated in our previous TractorBeam user studies.

### 4.2 Hardware Setup

The hardware setup included a top-projected tabletop display consisting of a ceiling-mounted projector, mirror, desktop PC, wooden table, and white cardboard “screen” (Figure 5).



**Fig. 5.** TractorBeam hardware configuration

The PC was connected to the projector and its output was projected onto the mirror, which reflected the image onto the table. The cardboard screen was used to provide a clearer projection surface than the table alone.

Input for the tabletop display was received via a corded stylus and receiver attached to a Polhemus Fastrak® (a “six degrees of freedom” 3D tracking system). The Fastrak® receiver was secured to the centre of the underside of the table. Using information from the stylus and receiver, the Fastrak® base provided continuous information about the position of the stylus in 3D space to our software through a serial port connection on the PC. Our software then used this information to calculate the spot on the table to which the pen was pointing, and drew the cursor at that location. During our experiments, we experienced no noticeable lag or accuracy issues with the Pohemus.

### 4.3 Task

A multi-directional task (2D Fitts discrete task) was used to evaluate selection tasks in four conditions: control, expand-cursor, expand-target, and snap-to-target. A Java application was developed to implement the selection aids required for each of the four conditions on our tabletop display.

Participants used the TractorBeam interaction technique throughout the experiment, but used each of the four types of selection aids in four separate conditions. In all conditions, participants selected targets on the table by pointing to them or touching them with the stylus to position the cursor on the target, and clicking the stylus button to indicate the selection.

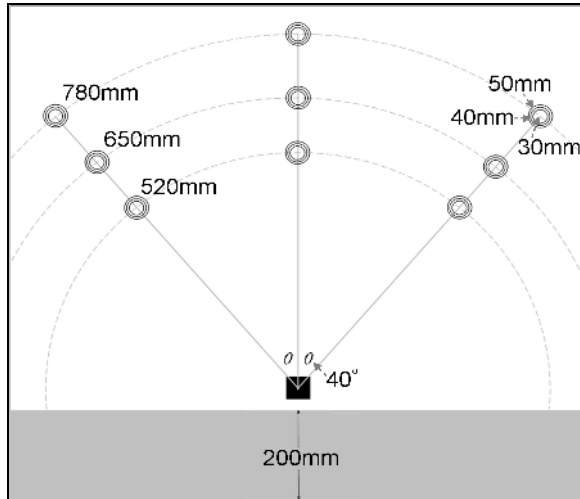
We had previously used dwell times to indicate target selection with the TractorBeam [9]. The stylus button click technique used in the present experiment introduces a small problem: the physical act of clicking the button caused the TractorBeam stylus to shift slightly, potentially displacing the cursor from the intended target. To compensate for this problem, our software tracked the length of time between the cursor exiting the target and the button being pressed. If the cursor was outside of the target when the button was pressed, but the time since it had exited the target was under a pre-determined threshold, we counted the click as a successful target selection. We tested several button-press threshold times in a small pilot study and found 100ms to be a suitable length of time to provide adequate compensation. Thus, if the button was pressed within 100ms after the cursor exited a target, it was counted as a successful target acquisition.

It has been shown that, with expanding targets, the expanded target width dictates the difficulty of the task even if the target does not expand until the cursor has already traveled 90% of the distance from the start position to the target [7]. As such, we designed all three of our selection aids to only take effect after the cursor had traveled 90% of the distance to the target. This design was sufficient for this preliminary investigation since only one target was visible at a time, and all targets were distant. Further investigation would obviously be required to examine appropriate thresholds when targets are grouped close together.

In each condition, participants were presented with a series of trials that required them to first select a home square (located in the bottom centre of the display area) and subsequently select a target circle (Figure 6). Target circles were presented with one of three widths (30mm, 40mm, 50mm), at one of three angles (40 degrees left, midline, 40 degrees right), and at one of three amplitudes (520mm, 650mm, 780mm).

Participants in our previous research were slower with pointing than touching for small, distant targets. Therefore, for this study we chose widths which ranged from slightly smaller than the “medium” sizes in our first study to exactly the “small” sizes used in our first study and amplitudes which ranged from just above the “medium” amplitude in our first study to exactly the “far” amplitude in our first study.

Each individual trial began when a user selected the home square, and ended when they selected the target circle. Selection was defined by a stylus button click in the target circle. Between the users’ selection of the home square and the appearance of the target circle, there was a random-length pause of between 500 and 1500 ms.



**Fig. 6.** 2D task setup. The black square is the starting point and the circles represent the targets.

Participants were asked to keep the cursor on the home square until the target appeared. Software logged when a target appeared, when a user moved off the home square, and when a user selected the target circle. As in our earlier studies, movement time was calculated as the difference between the time a user moved off the home square and the time they selected the target.

#### 4.4 Procedure

A within subjects design was utilized with each participant using all three selection aids and a control. To minimize order effects, condition order was counterbalanced.

After a background questionnaire, participants were asked to perform a series of trials using the experimental task software in each of the four conditions. Participants sat at the tabletop display and were asked to remain seated for the duration of the session. For every condition, each participant first completed a warm-up session which required them to select 10 random targets. They then completed exactly five trials of each unique combination of amplitude, width, and angle, for a total of 135 trials. The ordering of the trials was randomized for each participant. On average, participants took 12 minutes to complete each interaction technique (including answering the questionnaire), for a combined session total of approximately 48 minutes for all three interaction techniques.

Following each condition, users completed a post-task questionnaire to gather data on their comfort and perceived performance with the selection aid used. This questionnaire was based on the device assessment questionnaire from the ISO 9241, Part 9 standard [3], which outlines requirements for non-keyboard computer input devices. Once all four conditions were finished, users were given a final questionnaire asking them to rate the selection aids in terms of satisfaction and perceived effectiveness.



## 4.5 Hypotheses

Our previous research showed that users had difficulty using the TractorBeam to select small, distant targets. We expected that each of our three selection aids would perform better than the TractorBeam with no selection aid (control condition). We also expected that there would be significant differences between the selection aids.

## 4.6 Data Analyses

Computer logs were used to determine the dependent measures of movement time (MT) and error rate. We did not include reaction time in the movement time, due to the fact that the large tabletop display doesn't fit in a user's field of view and some time would likely be spent visually locating each target. Instead, MT data were calculated from when the cursor exited the home square until the user selected the target.

Errors occurred if the cursor was not on the target when the stylus button was clicked. We removed 1960 (14.7% of total trials) selection errors from the MT analysis. The high number of errors reflects the fact that all of the targets in our study were small and distant from the participants. If a wider range of target sizes and distances had been used, we would have expected a lower error rate.

Movement time data for the five repeated trials at each unique combination of target variables were averaged. Repeated Measures Analysis of Variance (ANOVAs) were performed on the mean MT and entry rate data. All main effects and interactions were tested at  $\alpha=.05$ . Questionnaires were analyzed using non-parametric statistical tests.

## 5 Results

Movement time and error data for all conditions are presented in Table 1. For the error rates, totals for each condition are given along with the percentage of total trials constituted by those errors.

**Table 1.** Mean MT in ms and error rate for each condition

<b>Condition</b>	<b>Movement Time</b> <i>Mean ms (SE)</i>	<b>Error Rate</b> <i>Total (%)</i>
Control	1544 (19.5)	586 (18.1%)
Expand-cursor	1326 (20.5)	313 (9.7%)
Expand-target	1370 (16.3)	429 (13.2%)
Snap-to-target	1060 (15.6)	582 (17.9%)

### 5.1 Hypothesis 1: Control Condition Would Be Slower Than the Other Conditions

ANOVAs were performed on the movement time data for the 4-condition design. As expected, there was a main effect for condition ( $F_{3,69}=14.7, p=.000, \eta^2=.39$ ). Pairwise

comparisons revealed our hypothesis was validated, with the control being significantly slower than expand-cursor ( $F_{1,23}=8.6$ ,  $p=.008$ ,  $\eta^2=.27$ ), expand-target ( $F_{1,23}=5.7$ ,  $p=.026$ ,  $\eta^2=.20$ ), and snap-to-target ( $F_{1,23}=51.5$ ,  $p=.000$ ,  $\eta^2=.69$ ).

## 5.2 Hypothesis 2: There Would Be a Difference Between the Three Selection Aids

ANOVAs were performed on the movement time data for the 3-(selection aid) condition design. There was a main effect for condition ( $F_{2,46}=9.7$ ,  $p=.000$ ,  $\eta^2=.30$ ), which validated our hypothesis. Pairwise comparisons revealed that snap-to-target was significantly faster than both expand-cursor ( $F_{1,23}=9.4$ ,  $p=.006$ ,  $\eta^2=.29$ ) and expand-target ( $F_{1,23}=20.9$ ,  $p=.000$ ,  $\eta^2=.48$ ). However, there was no significant difference between expand-cursor and expand-target ( $F_{1,21}=0.4$ ,  $p=.556$ ,  $\eta^2=.02$ ).

## 5.3 Error Rates

In order to investigate the number of errors in the various conditions, ANOVAs were performed on the error rate data for the 4-condition design. There was a main effect for condition ( $F_{3,69}=7.2$ ,  $p=.000$ ,  $\eta^2=.24$ ). Pairwise comparisons revealed that the control condition had significantly more errors than both expand-cursor ( $F_{1,23}=34.5$ ,  $p=.000$ ,  $\eta^2=.60$ ) and expand-target ( $F_{1,23}=6.2$ ,  $p=.021$ ,  $\eta^2=.21$ ), but that there was no significant difference with snap-to-target ( $F_{1,23}=.003$ ,  $p=.956$ ,  $\eta^2=.000$ ).

Mean error rates for each condition, separate by amplitude and width, are displayed in Figure 7. A value of zero would indicate no errors occurred for that target. The snap-to-target condition had significantly more errors than the expand-cursor ( $F_{1,23}=20.0$ ,  $p=.000$ ,  $\eta^2=.47$ ) condition, but not the expand-target condition ( $F_{1,23}=3.2$ ,  $p=.088$ ,  $\eta^2=.12$ ). There was no significant difference in errors between the expand-cursor and expand-target conditions ( $F_{1,23}=2.0$ ,  $p=.170$ ,  $\eta^2=.08$ ).

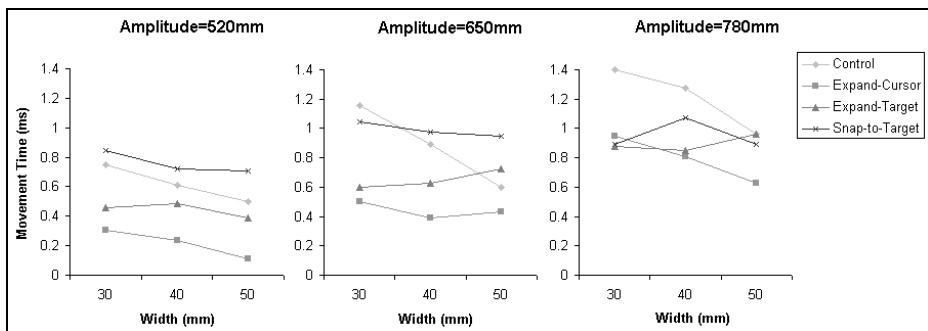


Fig. 7. Mean error rates for each condition, separated by amplitude and width

## 5.4 Inverse Efficiency

Although snap-to-target was found to be the fastest selection aid in our study, it also had significantly more errors than the slower expand-cursor aid. Thus, we wanted to

investigate the speed/accuracy tradeoffs of the various conditions. Townsend and Ashby [12] suggest combining movement time and error measures with the following equation for inverse efficiency (IE):  $IE = MT / (\text{Proportion of trials correct})$ . For example, a mean movement time of 2000 ms with 4 out of 5 trials successful would result in an inverse efficiency of 2500 ( $IE = 2000/0.8$ ). A lower IE score corresponds to a more efficient technique.

Inverse efficiencies were calculated using the collected MT and error data. Mean inverse efficiency for each of the conditions is displayed in Table 2.

**Table 2.** Mean inverse efficiency for each condition

Condition	Inverse Efficiency (IE)
Control	2411
Expand-cursor	1606
Expand-target	2003
Snap-to-target	1446

ANOVAs were performed on the IE data for the 4 condition design. There was a main effect for condition ( $F_{3,69}=5.1$ ,  $p=.003$ ,  $\eta^2=.18$ ). Pairwise comparisons revealed that snap-to-target had significantly lower inverse efficiency than the control condition ( $F_{1,23}=18.2$ ,  $p=.000$ ,  $\eta^2=.44$ ). However, while the mean IE for snap-to-target was also lower than that of the other selection aid conditions, it was not significantly different ( $F_{1,23}=4.0$ ,  $p=.055$ ,  $\eta^2=.15$  when compared to expand-target,  $F_{1,23}=.95$ ,  $p=.341$ ,  $\eta^2=.04$  compared to expand-cursor).

## 5.5 Questionnaire Responses

After each condition participants rated a number of factors related to effort, comfort, and effectiveness on a five-point scale. To determine differences between the conditions, results from these questionnaires were analyzed using a Friedman test. The means are summarized in Table 3.

There was a significant difference in perceived speed of the four conditions ( $\chi^2=18.7$ ,  $p=.000$ ). Wilcoxon matched-pairs tests revealed that participants perceived that both expanding-cursor ( $p=.003$ ) and snap-to-target ( $p=.002$ ) were significantly faster than the control condition, and there were no significant differences between other pairs.

There was also a significant difference between conditions in terms of perceived accuracy ( $\chi^2=16.1$ ,  $p=.001$ ). Wilcoxon matched-pairs tests revealed that they found snap-to-target ( $p=.006$ ) significantly more accurate than the control condition, but there were no significant differences between other pairs.

In terms of comfort, there was again a significant difference between conditions ( $\chi^2=10.2$ ,  $p=.017$ ), with matched-pairs tests showing snap-to-target as significantly more comfortable than the control ( $p=.008$ ) but no other significant differences between pairs. For ease of use, there was also a significant difference between conditions ( $\chi^2=14.5$ ,  $p=.002$ ), with matched-pairs tests revealing both snap-to-target

( $p=.003$ ) and expanding-cursor ( $p=.004$ ) significantly easier to use than the control, and no other significant differences between pairs.

At the end of the experiment we asked the participants to rate the four conditions according to how effective they were and how much they liked each technique. To determine differences between the interaction techniques, results from these questionnaires were also analyzed using a Friedman test. The means are summarized in Table 4. There was a significant difference between the conditions in terms of both effectiveness ( $\chi^2=22.96$ ,  $p=.000$ ) and enjoyability ( $\chi^2=22.94$ ,  $p=.000$ ).

Matched-pairs tests revealed that snap-to-target ( $p=.001$ ), expand-cursor ( $p=.000$ ), and expand-target ( $p=.001$ ) were all perceived by users to be significantly more effective than the control condition, but there were no significant differences between other pairings. Additionally, users enjoyed using snap-to-target ( $p=.001$ ), expand-cursor ( $p=.000$ ), and expand-target ( $p=.001$ ) significantly more than the control condition, with no significant differences between other pairings.

Participant feedback supported our quantitative finding that snap-to-target was as an effective and enjoyable selection aid:

"The snap helps a lot especially for small targets."

"I liked the fact that the snap-to-target stopped the cursor, as objects far away were harder to select because the cursor became more sensitive."

**Table 3.** Mean responses from condition questionnaires on a five-point scale where 1 is low and 5 is high. (\* denotes  $p<.05$ )

	Control Mean (SD)	Expand-Cursor Mean (SD)	Expand-Target Mean (SD)	Snap-to-Target Mean (SD)
Mental Effort	3.21 (.93)	2.96 (.81)	3.08 (.78)	2.58 (.97)
Physical Effort	3.92 (.65)	3.58 (.72)	3.71 (.69)	3.46 (.83)
Perceived Speed*	3.21 (.93)	3.96 (.81)	3.71 (.91)	4.25 (.90)
Perceived Accuracy*	2.96 (1.16)	3.58 (1.11)	3.79 (.88)	4.04 (1.12)
Wrist Fatigue	3.50 (.98)	3.17 (.92)	3.46 (.72)	3.25 (1.03)
Arm Fatigue	2.96 (1.09)	2.75 (.90)	3.13 (1.19)	2.75 (1.03)
Shoulder Fatigue	2.71 (1.09)	2.67 (.96)	2.50 (1.02)	2.37 (.82)
Neck Fatigue	2.33 (.96)	2.54 (.98)	2.21 (.88)	2.29 (.96)
Comfort*	2.96 (.81)	3.50 (.83)	3.42 (.88)	3.67 (.82)
Ease of Use*	3.04 (.96)	3.88 (1.04)	3.46 (1.02)	3.96 (.81)

**Table 4.** Mean responses from post-session questionnaires on a five-point scale where 1 is low and 5 is high. (\* denotes  $p<.05$ )

	Control Mean (SD)	Expand-Cursor Mean (SD)	Expand-Target Mean (SD)	Snap-to-Target Mean (SD)
Effective*	2.33 (1.01)	3.71 (.859)	3.63 (.875)	4.04 (1.122)
Enjoy*	2.33 (1.129)	3.75 (.847)	3.62 (1.09)	4.00 (1.103)

## 6 Discussion

We hypothesized that all three of our selection aids would improve upon the original TractorBeam interaction technique used in the control condition. This hypothesis was validated, as movement times for the control condition were significantly slower than all three selection aids. Additionally, we were able to confirm our hypothesis that there would be a difference between the three selection aids, with snap-to-target being significantly faster than the other two conditions.

There were also differences in the number of errors made by participants in the different conditions. There was no significant difference between the number of errors in the control and snap-to-target conditions. So, although the snap-to-target selection aid improved on movement times it had a higher error rate than the other selection aids. In particular, despite its slower movement time, the expand-cursor selection aid had significantly fewer errors than snap-to-target.

Combining the errors and movement times with an inverse efficiency calculation provided some insight into the speed/accuracy tradeoffs for our four conditions. While our fastest condition, snap-to-target, did have a significantly lower inverse efficiency than the control condition, there was no significant difference between it and the other two selection aids. This suggests that the number of errors that happen with snap-to-target may limit its efficiency to the point of it being on par with our other techniques.

Through our questionnaires, we found that users perceived snap-to-target to be significantly faster, more accurate, more comfortable, and easier to use than the control condition. They also perceived expanding-cursor as significantly faster and easier to use than the control. Additionally, all three selection aids were perceived to be more enjoyable and effective than the control condition.

Overall, snap-to-target was the only selection aid to be perceived as significantly more comfortable and accurate than the control condition. It also had significantly lower movement times than all other conditions, and was as good as the other two selection aids in terms of inverse efficiency. While there were more errors with snap-to-target than some of the other selection aids, future work on the TractorBeam could explore optimal snap thresholds for minimizing error while maximizing movement time, eventually improving on the condition's error rate.

## 7 Conclusions and Future Work

We explored several methods for improving selection of small, distant targets with the TractorBeam. Augmenting the TractorBeam with each of these selection aids increased our technique's effectiveness for selection of distant items. Additionally, despite no significant difference in inverse efficiency scores between the three selection aids, the positive user feedback from our third study gives snap-to-target an edge over the other solutions. Snap-to-target solves the main problem encountered with the TractorBeam technique in our first study, and further increases its viability as an interaction technique for large tabletop displays.

Although the snap-to-target selection aid had very positive results in our user study, this finding is only the first step. This study explored the effectiveness of many

possible selection aids for isolated, distant targets. Having now identified snap-to-target as a very effective technique, we plan to further examine this selection aid for less controlled, more ecologically valid tasks. In particular, we would like to test its effectiveness with groups of targets which are close together, as well as the general usability of the technique for close targets.

## Acknowledgements

Thanks to NSERC, Dalhousie University, and Mitsubishi Electric Research Labs for providing funding for this work. Thanks also to members of Dalhousie's EDGE Lab.

## References

1. Blanch, R., Guiard, Y., and Beaudouin-Lafon, M. (2004). Semantic Pointing: Improving target acquisition with control-display ratio adaptation. In *Proceedings of CHI 2004*. p. 519-526.
2. Deitz, P. and Leigh, D. (2000). DiamondTouch: A multi-user touch technology. In *Proceedings of UIST 2000*. p. 219-226.
3. Douglas, S.A., Kirkpatrick, A.E., and MacKenzie, I.S. (1999). Testing pointing device performance and user assessment with the ISO 9241, Part 9 standard. In *Proceedings of CHI '99*. p. 215-222.
4. Grossman, T. and Balakrishnan, R. (2005) The Bubble Cursor: Enhancing target acquisition by dynamic resizing of the cursor's activation area. In *Proceedings of CHI 2005*. p. 281-290.
5. Guiard, Y., Blanch, R., and Beaudouin-Lafon, M. (2004). Object pointing: a complement to bitmap pointing in GUIs. In *Proceedings of GI 2004*. p. 9-16.
6. Kabbash, P. and Buxton, W. (1995). The "Prince" Technique: Fitts' Law and selection using area cursors. In *Proceedings of CHI 1995*. p. 273-279.
7. McGuffin, M. and Balakrishnan, R. (2002). Acquisition of expanding targets. In *Proceedings of CHI 2002*. p. 57-64.
8. OS X, Apple Computer.
9. Parker, J.K., Mandryk, R., Inkpen, K.. (2005) TractorBeam: Seamless integration of local and remote pointing for tabletop displays. In *Proceedings of Graphics Interface 2005*. p. 33-40.
10. Rekimoto, J. and Saitoh, M. (1999). Augmented Surfaces: A spatially continuous work space for hybrid computing environments. In *Proceedings of CHI '99*. p. 378-385.
11. Shen, C., Lesh, N.B., Vernier, F., Forlines, C., and Frost, J. (2002). Sharing and building digital group histories. In *Proceedings of CSCW 2002*. p. 3.
12. Townsend, J.T. and Ashby, F.G., (1983). Stochastic modelling of elementary psychological processes. London: Cambridge University Press.
13. Worden, A., Walker, N., Bharat, K., Hudson, S. (1997). Making Computers Easier for Older Adults to Use: Area Cursors and Sticky Icons. In *Proceedings of CHI '97*. p. 266-271.
14. Wu, M. and Balakrishnan, R. (2003). Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. In *Proceedings of UIST 2003*. p. 193-202.
15. Zhai, S., Conversy, S., Beaudouin-Lafon, M., and Guiard, Y. (2003). Human on-line response to target expansion. In *Proceedings of CHI 2003*. p. 177-184.